

TUGAS AKHIR – 184833

**ANALISIS KINERJA PERAMALAN DAN KLASIFIKASI
PERMINTAAN PART OTOMOTIF DENGAN PENDEKATAN
*TIME-SERIES DAN DATA MINING***

DEFA IHSAN RAMADHAN

NRP. 02411640000032

DOSEN PEMBIMBING :

Prof. Ir. Budi Santosa, MS., Ph.D.

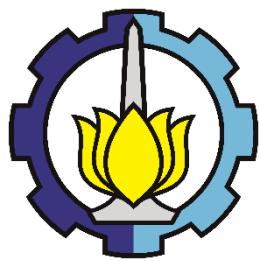
NIP. 196905121994021001

DOSEN KO-PEMBIMBING :

Putu Dana Karningsih, ST., M.Eng.Sc., Ph.D.

NIP. 197405081999032001

DEPARTEMEN TEKNIK SISTEM DAN INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI DAN REKAYASA SISTEM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2020



TUGAS AKHIR – 184833

**ANALISIS KINERJA PERAMALAN DAN KLASIFIKASI
PERMINTAAN PART OTOMOTIF DENGAN PENDEKATAN
*TIME-SERIES DAN DATA MINING***

DEFA IHSAN RAMADHAN

NRP. 02411640000032

DOSEN PEMBIMBING :

Prof. Ir. Budi Santosa, MS., Ph.D.

NIP. 196905121994021001

DOSEN KO-PEMBIMBING :

Putu Dana Karningsih, ST., M.Eng.Sc., Ph.D.

NIP. 197405081999032001

DEPARTEMEN TEKNIK SISTEM DAN INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI DAN REKAYASA SISTEM
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2020

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

ANALISIS KINERJA PERAMALAN DAN KLASIFIKASI PERMINTAAN PART OTOMOTIF DENGAN PENDEKATAN *TIME-SERIES DAN DATA MINING*

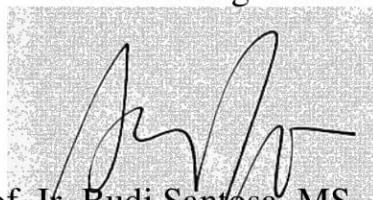
TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat Memeroleh Gelar Sarjana Teknik
pada Program Studi S-1 Departemen Teknik Sistem dan Industri
Fakultas Teknologi Industri dan Rekayasa Sistem
Institut Teknologi Sepuluh Nopember
Surabaya

Penulis:
DEFA IHSAN RAMADHAN
NRP. 02411640000032

Disetujui oleh
Dosen Pembimbing Tugas Akhir:

Pembimbing Utama



Prof. Ir. Budi Santosa, MS., Ph.D. Putu Dana Karningsih, ST., M.Eng.Sc., Ph.D.
NIP. 196905121994021001 NIP. 197405081999032001

Ko-Pembimbing



ABSTRAK

ANALISIS KINERJA PERAMALAN DAN KLASIFIKASI PERMINTAAN PART OTOMOTIF DENGAN PENDEKATAN TIME-SERIES DAN DATA MINING

Nama Mahasiswa : Defa Ihsan Ramadhan
NRP : 02411640000032
Pembimbing : Prof. Ir. Budi Santosa, MS., Ph.D.
Ko-pembimbing : Putu Dana Karningsih, ST., M.Eng.Sc., Ph.D.

ABSTRAK

Kontribusi *after sales service* yang besar terhadap profit dan pertumbuhan bisnis menyebabkan *spare part management* menjadi faktor yang penting untuk bersaing di pasar. Termasuk bagi PT. X, produsen mobil penumpang dengan *market share* terbesar secara global di tahun 2019. Berbagai upaya *spare part management* telah dilakukan PT. X untuk memenuhi kebutuhan *part* domestik maupun ekspor. Salah satu upayanya adalah melakukan perencanaan produksi bulanan (Getsudo), termasuk peramalan permintaan *spare part* setiap bulannya. Akan tetapi, metode peramalan *moving average*, yang saat ini digunakan untuk semua *spare part* di PT. X, kurang efektif pada beberapa pola permintaan *spare part* yang variasinya tinggi. Penyimpangan pada hasil peramalan berdampak pada *lead time back order* dan biaya *material handling* yang semakin besar. Berdasarkan hal tersebut maka PT. X memerlukan perbaikan terhadap sistem peramalan *spare part*-nya. Tugas akhir ini memiliki dua tujuan utama. Tujuan pertama adalah mengusulkan pengelompokan *spare part* berdasarkan pola permintaannya sebelum diramalkan. Tujuan kedua adalah menentukan metode peramalan yang paling sesuai untuk masing-masing kelompok *spare part* dengan cara membandingkan empat metode peramalan yaitu: Croston, Modifikasi Croston, SVR, dan ANN. Seluruh metode peramalan dibandingkan berdasarkan parameter *forecasting error* dan *robustness*. Hasil penelitian ini menunjukkan bahwa metode SVR dan ANN memiliki kinerja yang lebih unggul dari metode lainnya di tahap *training* maupun *testing*. Selain itu, ketika diimplementasikan untuk peramalan multi-periode, metode SVR dan ANN juga lebih unggul dan dapat memperbaiki kesalahan peramalan sebesar 18.8% dibandingkan dengan metode peramalan yang saat ini digunakan oleh PT. X.

Kata kunci: *Artificial Neural Network*, *Data Mining*, Metode Croston, Peramalan *Spare Part*, Peramalan *Time-Series*, *Support Vector Regression*.

ABSTRACT

ANALYSIS OF FORECASTING PERFORMANCE AND CLASSIFICATION OF AUTOMOTIVE PART DEMAND WITH TIME-SERIES AND DATA MINING APPROACHES

Author Name	: Defa Ihsan Ramadhan
NRP	: 02411640000032
Supervisor	: Prof. Ir. Budi Santosa, MS., Ph.D.
Co-supervisor	: Putu Dana Karningsih, ST., M.Eng.Sc., Ph.D.

ABSTRACT

The large contribution to profit and business growth of after sales service have made spare part management as an important factor to compete in the market. Especially in PT. X, a passenger car manufacturer with the largest market share globally in 2019, various part of service management efforts have been carried out to meet the domestic and export part demand. One of them is monthly production planning (Getsudo), in which including forecasting spare part demand every month. However, the moving average forecasting method, which is currently being used to forecast all spare part demand in PT. X, is proven less effective to high demand variations of spare part. Deviations in forecasting results in greater lead time backorders and material handling costs. Because of those problems, PT. X requires improvements to its spare part forecasting system. This final project has two main objectives. The first objective is to propose the classification system of spare part based on the demand pattern before being forecasted. The second objective is to determine the forecasting method that is most appropriate for each group of spare part by comparing the four forecasting methods, namely: Croston, Croston Modification, SVR, & ANN. All forecasting methods are compared based on forecasting error & robustness. The results of this study indicate that the SVR and ANN methods have performance that is superior to other methods in the training and testing data. In addition, when implemented for multi-period forecasting, the SVR and ANN methods are also superior and can improve forecasting errors by 18.8% compared to the existing forecasting method.

Keywords: Artificial Neural Network, Croston's Method, Data Mining, Spare Part Forecasting, Support Vector Regression, Time-Series Forecasting.

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur penulis haturkan kepada Allah SWT atas segala rahmat dan karunia-NYA sehingga penulis dapat menyelesaikan tugas akhir ini dengan lancar dan tepat pada waktunya. Sholawat dan salam penulis haturkan kepada Nabi Muhammad SAW yang senantiasa menjadi tauladan dan pengingat bagi penulis dalam menuntut ilmu.

Penulis menyadari bahwa penulisan tugas akhir ini tidak luput dari kesalahan dan kekurangan. Maka dari itu, dengan selesainya tugas akhir ini tidak terlepas dari kritik dan saran yang membangun dari berbagai pihak. Oleh karena itu, secara tulus penulis ingin mengucapkan terima kasih kepada:

1. Bapak Dedi Tarmidi dan Ibu Ayi Rohilah, orang tua penulis, yang senantiasa mendoakan dan memenuhi kebutuhan penulis baik material maupun moral.
2. Bapak Prof. Ir. Budi Santosa, MS., Ph.D., pembimbing tugas akhir penulis, yang telah meluangkan waktu dan pikiran untuk memberikan kritik dan saran yang membangun bagi penelitian.
3. Ibu Putu Dana Karningsih, ST., M.Eng.Sc., Ph.D., pembimbing tugas akhir penulis, yang telah memberikan peluang bagi penulis untuk bekerja-sama dengan PT. X dan meluangkan waktu serta pikiran untuk memberikan kritik dan saran bagi penelitian.
4. Bapak Hertyoso Nursasongko, Kepala Divisi *After Sales Service & Production* di PT. X, yang telah memberikan kesempatan dan fasilitas bagi penulis untuk menyelesaikan penelitian di PT. X.
5. Bapak Rifky Widipratomo, dari Divisi *After Sales Service & Production* di PT. X, yang telah menjadi mentor penulis selama menyelesaikan penelitian di PT. X.
6. Pihak PT. X lainnya yang telah membantu kelancaran penelitian penulis.
7. Dea Ahmad dan Fathia Mauludina, kakak penulis, yang senantiasa memberikan semangat dan membagi pengalamannya.

8. Teman-teman Asisten Laboratorium QMIPA Angkatan 2014-2017 yang senantiasa membagi ilmunya dan menjadi teman berdiskusi dan bertukar pikiran.
9. Teman-teman Asisten Laboratorium SISMAN Angkatan 2016 yang telah memberikan informasi kepada penulis untuk melakukan penelitian ini.
10. Teman-teman ADHIGANA, TI Angkatan 2016, yang senantiasa memberikan bantuan material dan moral selama kegiatan akademik dan non-akademik.

Surabaya, Juli 2020

Penulis

(Halaman ini sengaja dikosongkan)

DAFTAR ISI

DAFTAR ISI

LEMBAR PENGESAHAN	iii
ABSTRAK.....	v
ABSTRACT.....	vii
KATA PENGANTAR	ix
DAFTAR ISI.....	xiii
DAFTAR TABEL.....	xix
DAFTAR GAMBAR	xxi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	8
1.3 Tujuan Penelitian.....	8
1.4 Manfaat Penelitian.....	8
1.5 Ruang Lingkup Penelitian	9
1.5.1 Batasan	9
1.5.2 Asumsi	10
1.6 Sistematika Penulisan.....	10
BAB 2 TINJAUAN PUSTAKA	13
2.1 <i>Spare Part Management</i>	13
2.2 Klasifikasi Permintaan <i>Spare Part</i>	14
2.2.1 Pendekatan Boylan.....	15
2.2.2 Teknik Klaster.....	17
2.3 Peramalan <i>Spare Part</i>	22
2.3.1 Pendekatan Time-Series.....	23
2.3.2 Pendekatan Data Mining.....	27

2.4	Metode Croston dan Modifikasinya.....	28
2.5	Support Vector Machine (SVM).....	30
2.6	Artificial Neural Network (ANN).....	36
2.7	Mengukur Kinerja Peramalan (<i>Forecast Error</i> dan <i>Robustness</i>)	40
2.8	Validasi Peramalan	42
2.9	Posisi Penelitian.....	43
	BAB 3 METODOLOGI PENELITIAN	49
3.1	Evaluasi Sistem <i>Existing</i>	50
3.2	Pra-Pengolahan Data.....	51
3.3	Pengelompokkan Data	52
3.4	Peramalan.....	56
3.5	Validasi	59
3.6	Interpretasi dan Analisis	61
3.7	Kesimpulan dan Saran	62
	BAB 4 PRA-PENGOLAHAN DAN PENGELOMPOKKAN DATA	65
4.1	Seleksi dan Pembersihan Data Riwayat Permintaan <i>Spare Part</i>	65
4.2	Transformasi Data Riwayat Permintaan <i>Spare Part</i>	66
4.2.1	Normalisasi Skala	67
4.2.2	Konversi Ke-Bentuk Predictor-Target.....	67
4.3	Pengelompokkan Pola Permintaan <i>Spare Part</i>	67
4.3.1	Klasifikasi Boylan	68
4.3.2	K-Medoids dan Agglomerative Hierarchical Clustering	69
4.3.3	Pengelompokkan Berdasarkan Family Part.....	72
4.4	Eksperimentasi Data <i>Training</i> Peramalan	72
4.4.1	Trend dan Seasonal Adjustment	72
4.4.2	Transformasi Stasioner Box-Cox dan Differencing	74

4.4.3	Ringkasan Data Training untuk Peramalan	76
BAB 5 PERAMALAN <i>SPARE PART</i> DAN VALIDASI.....		79
5.1	Pengembangan dan Validasi Model Peramalan Croston, SBA, dan TSB	
	79	
5.1.1	Struktur Model	79
5.1.2	Validasi Model.....	80
5.2	Pengembangan dan Validasi Model Peramalan SVR	90
5.2.1	Struktur Model	91
5.2.2	Validasi Model.....	91
5.3	Pengembangan dan Validasi Model Peramalan ANN	96
5.3.1	Struktur Model	97
5.3.2	Validasi Model.....	98
5.4	Implementasi Peramalan Multi-Periode dan Perhitungan Interval Peramalan.....	102
5.4.1	Pemilihan Model Peramalan Terbaik.....	102
5.4.2	Implementasi Peramalan Multi-Periode dan Perhitungan Interval Peramalan Dengan Model Terpilih	110
BAB 6 INTERPRETASI DAN ANALISIS		123
6.1	Analisis Sistem Pengelompokan Pola Permintaan <i>Spare Part</i> di PT. X	
	123	
6.2	Analisis Perbandingan Kinerja Berbagai Model Peramalan terhadap Permintaan <i>Spare Part</i> di PT. X	124
6.3	Rencana Implementasi Sistem Peramalan dalam Operasional di PT. X	
	126	
BAB 7 KESIMPULAN DAN SARAN		131
7.1	Kesimpulan.....	131
7.2	Saran.....	132

REFERENSI DAN DAFTAR PUSTAKA.....	135
LAMPIRAN	145

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

DAFTAR TABEL

Tabel 2. 1 Nilai <i>Cut-Off</i> untuk CV ² dan ADI	17
Tabel 2. 2 Berbagai Alternatif Pengukuran Akurasi Peramalan.....	40
Tabel 2. 3 Penelitian-Penelitian Serupa Mengenai Peramalan Permintaan <i>Spare Part</i>	44
Tabel 3. 1 Bentuk Pengelompokkan <i>Spare Part</i> Pada Tugas Akhir Ini	53
Tabel 3. 2 Metode Untuk Menentukan Jumlah Klaster	55
Tabel 3. 3 <i>Hyperparameter</i> Pada Masing-Masing Metode Peramalan.....	56
Tabel 4. 1 Nilai <i>Cut-Off</i> untuk CV ² dan ADI (2).....	68
Tabel 5. 1 Struktur Model Peramalan Croston, SBA, dan TSB	80
Tabel 5. 2 Data <i>Training</i> dan <i>Testing</i> Untuk Model Croston, SBA, dan TSB	80
Tabel 5. 3 Struktur Model Peramalan SVR dengan 2288 kombinasi <i>hyperparameter</i>	91
Tabel 5. 4 Data <i>Training</i> dan <i>Testing</i> Untuk Model SVR	92
Tabel 5. 5 Struktur Model Peramalan ANN dengan 8748 kombinasi <i>hyperparameter</i> , <i>dropout</i> , dan regularisasi	97
Tabel 5. 6 Perbandingan <i>Training Forecasting Error</i> Semua Model Peramalan Pada Semua Skenario Peramalan.....	103
Tabel 5. 7 Perbandingan <i>Testing Forecasting Error</i> Semua Model Peramalan Pada Semua Skenario Peramalan.....	105
Tabel 5. 8 Perbandingan Apakah Terjadi <i>Overfitting</i> Pada Masing-Masing Model Peramalan di Semua Skenario.....	107
Tabel 5. 9 Rekapitulasi Hasil Uji Signifikansi Pengaruh Berbagai Skenario Peramalan Terhadap Semua Model Peramalan (1).....	109
Tabel 5. 10 Perbandingan <i>In-Sample Forecasting Error</i> Metode Usulan Terbaik (SVR dan ANN) Dengan Metode <i>Existing</i> (SMA).....	115

DAFTAR GAMBAR

DAFTAR GAMBAR

Gambar 1. 1 <i>Global Automotive Market Share berdasarkan Brand</i>	2
Gambar 1. 2 Proses Bisnis <i>After Sales Service</i> di PT. X	3
Gambar 1. 3 Peramalan <i>Existing</i> Yang Cenderung Melebihi Permintaan Aktual Ekspor <i>Spare Part</i> di PT. X (<i>Overforecast</i>).....	4
Gambar 1. 4 Kinerja Metode Peramalan SMA terhadap Total Permintaan Ekspor <i>Spare Part</i> di PT. X	6
Gambar 1. 5 10 Negara Dengan Volume Ekspor Tertinggi Dari PT. X Selama Tahun 2017 – 2019	7
Gambar 2. 1 Berbagai Pendekatan Untuk Klasifikasi <i>Spare Part</i>	15
Gambar 2. 2 Ilustrasi Parameter CV^2 dan ADI.....	17
Gambar 2. 3 (a) Data Awal, (b) Implementasi <i>Partitional Clustering</i> , (c) Implementasi <i>Hierarchical Clustering</i>	18
Gambar 2. 4 Contoh Dendogram Dari Klaster Hierarkis.....	20
Gambar 2. 5 Urutan Proses Peramalan <i>Time-Series</i>	22
Gambar 2. 6 <i>Knowledge Discovery in Database (KDD) Process</i>	27
Gambar 2. 7 Ilustrasi Pemetaan <i>Input Space</i> Terhadap <i>Output Space</i>	31
Gambar 2. 8 ϵ -insensitive loss function, Semua Titik Di Luar Area Berwarna Dikenai Penalti	32
Gambar 2. 9 Ilustrasi Kernelisasi Suatu <i>Input</i> Ke Dimensi Yang Lebih Tinggi ..	34
Gambar 2. 10 Arsitektur <i>Artifical Neural Network</i>	36
Gambar 2. 11 Model <i>Single Layer Perceptron</i>	37
Gambar 2. 12 Berbagai Alternatif Fungsi Aktivasi ANN.....	38
Gambar 2. 13 Ilustrasi <i>Overfitting</i>	43
Gambar 3. 1 <i>Flowchart</i> Metodologi Penelitian	49
Gambar 3. 2 Interpretasi Nilai MAPE	50
Gambar 3. 3 Ilustrasi <i>similarity in time points</i> pada data <i>time-series</i>	52
Gambar 3. 4 Ilustrasi <i>Grid Search</i>	57
Gambar 3. 5 Ilustrasi Metode <i>Second-Half Prediction</i>	59
Gambar 3. 6 Ilustrasi Metode <i>Day Forward-Chaining</i>	60
Gambar 4. 1 Seleksi Data Berdasarkan Segmentasi Kasus	65

Gambar 4. 2 Seleksi Data Berdasarkan Objektif Peramalan	66
Gambar 4. 3 Rekomendasi Jumlah Klaster ($k = 9$) Hasil Uji Silhouette untuk Teknik K-Medoids	70
Gambar 4. 4 Rekomendasi Jumlah Klaster ($k = 3$) Hasil Uji CalinskiHarabasz untuk Teknik K-Medoids.....	70
Gambar 4. 5 Jumlah Klaster Optimal ($k = 9$) yang Meminimalkan SSE atau SSW	71
Gambar 5. 1 Uji Signifikansi Pengaruh Skenario Berbagai <i>Input</i> Peramalan.....	82
Gambar 5. 2 Peramalan Multi Periode (Periode = $T+3$) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode <i>Existing</i> (SMA)	111
Gambar 5. 3 Peramalan Multi Periode (Periode = $T+3$) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode <i>Existing</i> (SMA)	112
Gambar 5. 4 Peramalan Multi Periode (Periode = $T+3$) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode <i>Existing</i> (SMA)	113
Gambar 5. 5 Peramalan Multi Periode (Periode = $T+3$) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode <i>Existing</i> (SMA)	114
Gambar 5. 6 Plot ACF Dari Residual Peramalan SVR Terhadap Permintaan Aktual Semua Klaster Dan Pola Permintaan Boylan	117
Gambar 5. 7 Plot ACF Dari Residual Peramalan ANN Terhadap Permintaan Aktual Semua Klaster Dan Pola Permintaan Boylan	119
Gambar 6. 1 Tampilan Depan Pada Aplikasi	126
Gambar 6. 2 Tampilan Utama Pada Aplikasi	127
Gambar 6. 3 Tampilan <i>Report</i> Pada Aplikasi.....	127
Lampiran 1 <i>Min-Max Normalization</i> Menggunakan RStudio.....	145
Lampiran 2 Konversi Ke-Bentuk <i>Predictor Target</i> Menggunakan RStudio	145
Lampiran 3 Klasifikasi Boylan Menggunakan RStudio (1)	146
Lampiran 4 Klasifikasi Boylan Menggunakan RStudio (2)	146
Lampiran 5 Penentuan Ukuran <i>Linkage</i> Terbaik (Ward) Menggunakan RStudio	147
Lampiran 6 Proses K-Medoids <i>Clustering</i> Seluruh 1682 Pola Permintaan <i>Part</i>	147
Lampiran 7 Proses Agglomerative <i>Clustering</i> Seluruh 1682 Pola Permintaan <i>Part</i>	147

Lampiran 8 Hasil K-Medoids <i>Clustering</i> Menggunakan RStudio	148
Lampiran 9 Pola Permintaan <i>Part</i> Yang Menjadi Pusat Masing-Masing Klaster	148
Lampiran 10 Hasil Uji Mann-Kendall Menggunakan RStudio	149
Lampiran 11 Dekomposisi Tren (Merah) Menggunakan RStudio	149
Lampiran 12 Transformasi Fourier Menggunakan RStudio	150
Lampiran 13 Penentuan <i>Seasonality</i> Menggunakan RStudio (1)	150
Lampiran 14 Penentuan <i>Seasonality</i> Menggunakan RStudio (2)	151
Lampiran 15 Transformasi Data Setelah <i>Trend Adjustment</i> (Kanan) Menggunakan RStudio	151
Lampiran 16 Transformasi Data Setelah <i>Seasonal Adjustment</i> (Kanan) Menggunakan RStudio.....	152
Lampiran 17 Mencari Lambda Optimal ($\lambda = 0.45$) dan Melakukan Transformasi Box-Cox pada RStudio	152
Lampiran 18 <i>Skewness</i> Transformasi Box-Cox Lebih Mendekati Nol.....	153
Lampiran 19 Transformasi Box-Cox (Kanan) mendekati Distribusi Normal	153
Lampiran 20 Penentuan Orde <i>Differencing</i> dengan Uji Augmented Dickey-Fuller Menggunakan RStudio.....	153
Lampiran 21 Orde <i>Seasonal Differencing</i> Bernilai Nol.....	154
Lampiran 22 Data Menjadi Stasioner Dalam Rataan (Kanan) Setelah <i>First Order Differencing</i>	154
Lampiran 23 <i>Time-Series</i> Sudah Stasioner Dalam Rataan Pada Uji Augmented Dickey-Fuller <i>Test</i>	154
Lampiran 24 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Berbagai Pola Permintaan Boylan	155
Lampiran 25 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Berbagai Pola Permintaan Boylan	156
Lampiran 26 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode Croston.....	156
Lampiran 27 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Berbagai Pola Permintaan Boylan.....	157

Lampiran 28 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Berbagai Pola Permintaan Boylan.....	158
Lampiran 29 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode SBA	159
Lampiran 30 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode TSB Terhadap Berbagai Pola Permintaan Boylan	160
Lampiran 31 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i>	161
Lampiran 32 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode TSB	162
Lampiran 33 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Berbagai <i>Family Part</i>	163
Lampiran 34 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Berbagai <i>Family Part</i>	164
Lampiran 35 Uji Signifikansi Pengaruh <i>Family Part</i> Terhadap Metode Croston	164
Lampiran 36 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Berbagai <i>Family Part</i>	165
Lampiran 37 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Berbagai <i>Family Part</i>	166
Lampiran 38 Uji Signifikansi Pengaruh <i>Family Part</i> Terhadap Metode SBA....	167
Lampiran 39 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode TSB Terhadap Berbagai <i>Family Part</i>	168
Lampiran 40 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode TSB Terhadap Berbagai <i>Family Part</i>	169
Lampiran 41 Uji Signifikansi Pengaruh <i>Family Part</i> Terhadap Metode TSB	170
Lampiran 42 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama .	171
Lampiran 43 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama.....	172

Lampiran 44 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode Croston	173
Lampiran 45 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama.	174
Lampiran 46 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama	175
Lampiran 47 Uji Asumsi Anova Normalitas Residual MSE <i>Training</i> Metode SBA Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama	175
Lampiran 48 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode SBA	176
Lampiran 49 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode TSB Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama.	177
Lampiran 50 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode TSB Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama	178
Lampiran 51 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode TSB.....	179
Lampiran 52 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode Croston Terhadap Klaster K-Medoids	180
Lampiran 53 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode Croston Terhadap Klaster K-Medoids	181
Lampiran 54 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode Croston.....	182
Lampiran 55 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SBA Terhadap Klaster K-Medoids	183
Lampiran 56 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode SBA Terhadap Klaster K-Medoids.....	184
Lampiran 57 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode SBA	185

Lampiran 58 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode TSB Terhadap Klaster K-Medoids	186
Lampiran 59 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode TSB Terhadap Klaster K-Medoids	187
Lampiran 60 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode TSB	188
Lampiran 61 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SVR Terhadap Berbagai Pola Permintaan Boylan	189
Lampiran 62 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode SVR Terhadap Berbagai Pola Permintaan Boylan	190
Lampiran 63 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode SVR	191
Lampiran 64 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SVR Terhadap Berbagai <i>Family Part</i>	192
Lampiran 65 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode SVR Terhadap Berbagai <i>Family Part</i>	193
Lampiran 66 Uji Signifikansi Pengaruh <i>Family Part</i> Terhadap Metode SVR....	194
Lampiran 67 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SVR Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama .	195
Lampiran 68 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode SVR Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama	196
Lampiran 69 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode SVR.....	197
Lampiran 70 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode SVR Terhadap Klaster K-Medoids	198
Lampiran 71 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode SVR Terhadap Klaster K-Medoids	199
Lampiran 72 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode SVR	200
Lampiran 73 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode ANN Terhadap Berbagai Pola Permintaan Boylan	201

Lampiran 74 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode ANN Terhadap Berbagai Pola Permintaan Boylan	202
Lampiran 75 Uji Asumsi Anova Normalitas Residual MSE <i>Training</i> Metode ANN Terhadap Berbagai Pola Permintaan Boylan	202
Lampiran 76 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode ANN	203
Lampiran 77 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode ANN Terhadap Berbagai <i>Family Part</i>	204
Lampiran 78 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode ANN Terhadap Berbagai <i>Family Part</i>	205
Lampiran 79 Uji Signifikansi Pengaruh <i>Family Part</i> Terhadap Metode ANN ..	206
Lampiran 80 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode ANN Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama.	207
Lampiran 81 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode ANN Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama ..	208
Lampiran 82 Uji Asumsi Anova Normalitas Residual MSE <i>Testing</i> Metode ANN Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama ..	208
Lampiran 83 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode ANN Di Tahap <i>Training</i>	209
Lampiran 84 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode ANN Di Tahap <i>Testing</i>	209
Lampiran 85 Boxplot MSE <i>Training</i> dan <i>Testing</i> Metode ANN Terhadap Klaster K-Medoids	210
Lampiran 86 Uji Asumsi Anova Homogenitas Varian MSE <i>Training</i> Metode ANN Terhadap Klaster K-Medoids.....	211
Lampiran 87 Uji Asumsi Anova Normalitas Residual MSE <i>Testing</i> Metode ANN Terhadap Klaster K-Medoids.....	211

Lampiran 88 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode ANN	212
Lampiran 89 <i>Data Flow Diagram</i> Skema Aplikasi Peramalan Implementasi Dari Usulan Perbaikan Pada Tugas Akhir Ini.....	213
Lampiran 90 Model Peramalan Croston.....	214
Lampiran 91 Model Peramalan SBA dan TSB	214
Lampiran 92 Model Peramalan SVR.....	215
Lampiran 93 Model Peramalan ANN (1)	216
Lampiran 94 Model Peramalan ANN (2)	216
Lampiran 95 Model Peramalan ANN (3)	217
Lampiran 96 Visualisasi Peramalan Berdasarkan Metode dan Kelompok (Grup) <i>Part</i>	217
Lampiran 97 Perhitungan Interval Peramalan	218
Lampiran 98 Permintaan Ekspor Bulanan <i>Spare Part</i> Ke Arab Saudi Selama 2018-2019	219

BAB 1

PENDAHULUAN

BAB 1

PENDAHULUAN

Pada bab ini akan dijelaskan mengenai pendahuluan dari tugas akhir ini, yang meliputi: latar belakang, rumusan masalah, tujuan, manfaat, ruang lingkup penelitian, dan sistematika penulisan.

1.1 Latar Belakang

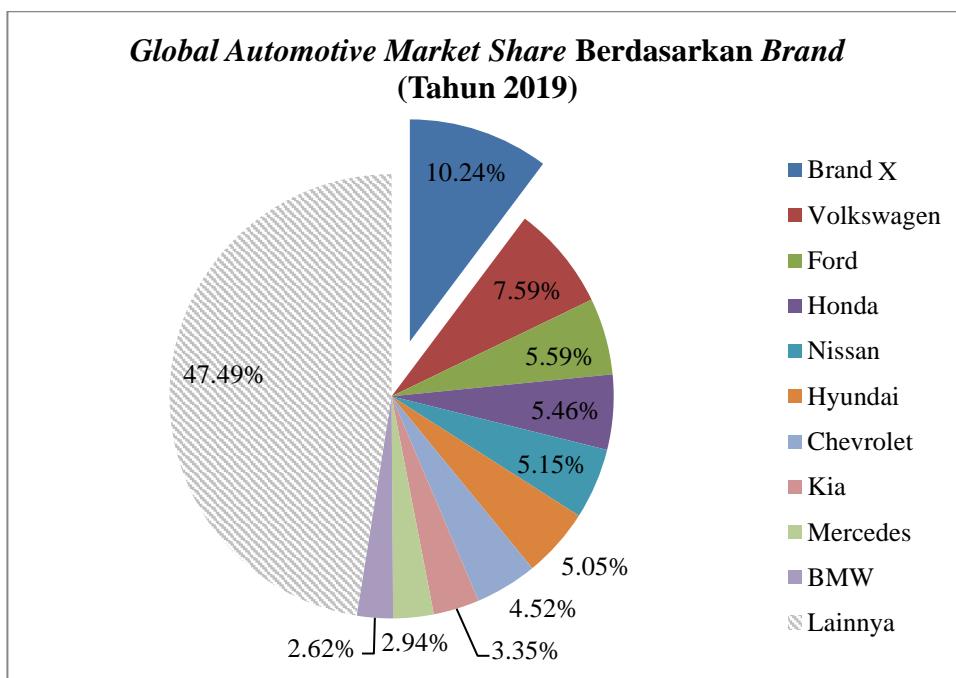
Lini bisnis yang paling penting dalam suatu industri otomotif selain dari penjualan produk utama adalah penjualan *spare part*. Hal tersebut ditunjukkan dengan kontribusi yang besar dari pelayanan *after sales*, termasuk dari penjualan *wear & tear part*, terhadap keseluruhan profit yang diperoleh industri otomotif (EY, 2018, McKinsey&Company, 2018). Selain itu, pasar *after sales service* diperkirakan akan terus berkembang selama satu dekade ke depan dengan proyeksi pertumbuhan *Compound Annual Growth Rate* (CAGR) sebesar 3% per tahun (McKinsey&Company, 2018). Oleh karena itu, persaingan antara produk otomotif tidak jarang diikuti juga dengan persaingan *after sales service*-nya.

Salah satu produk otomotif dengan penjualan yang berkembang pesat adalah mobil penumpang. Angka kepemilikan mobil penumpang yang terus meningkat adalah salah satu penyebab pertumbuhan permintaan *spare part* (Loukmidis & Luczak, 2006, McKinsey&Company, 2018). Hal tersebut karena ketersediaan *spare part* di pasar menjadi salah satu faktor kompetitif yang dapat mempengaruhi kepercayaan *customer* terhadap *brand* dan keputusan *customer* dalam membeli mobil (Klug, 2018). Oleh karena itu, agar dapat bertahan di pasar, produsen mobil perlu meningkatkan pelayanan *after sales*-nya dengan menjaga kualitas dan ketersediaan *spare part*.

Suatu mobil modern dapat tersusun lebih dari 30.000 *part* (Toyota Motor Corporation, 2019). Suatu waktu, *part* tersebut akan rusak dan *part* pengganti harus tersedia dalam jumlah dan kualitas yang tepat. Kerusakan suatu *part* sangat sulit diprediksi karena beragamnya faktor pemakaian dan akibat faktor yang tidak bisa dikontrol, seperti kecelakaan dan bencana (Pfohl, 2010). Banyaknya macam *part*,

tingginya ketidakpastian kebutuhan *part* pengganti, dan tuntutan ketersediaan di pasar, menyebabkan *spare part management* menjadi hal yang kompleks bagi produsen mobil.

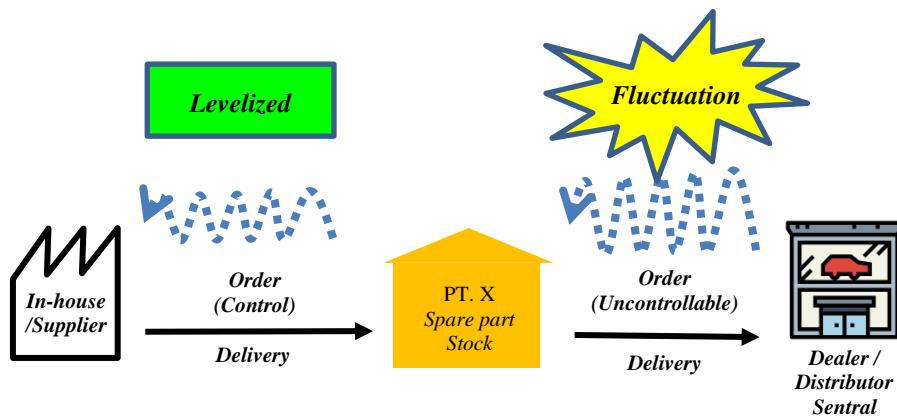
Berdasarkan **Gambar 1. 1**, salah satu produsen mobil dengan *market share* terbesar secara global di tahun 2019 adalah PT. X (Statista, 2020). Dalam mempertahankan *market share*, PT. X sangat memperhatikan kualitas pelayanan *after sales*-nya. Hal tersebut ditunjukkan dengan ketersediaan *dealer spare part* yang tersebar luas di 80 negara di seluruh dunia (PT. X, 2019). Selain itu, dalam mempertahankan kepercayaan *customer* terhadap *brand*, PT. X juga memberikan *warranty* ketersediaan *part* selama 15 tahun, baik untuk mobil jenis baru maupun lama yang dimiliki oleh *customer* (PT. X, 2019). Di Indonesia, pemberian jaminan ketersediaan *part* tersebut juga merupakan kewajiban produsen yang diatur dalam Pasal 25 Undang-Undang Perlindungan Konsumen No. 8 Tahun 1999. Oleh karena itu, *spare part management* menjadi suatu aktivitas yang penting di PT. X.



Gambar 1. 1 *Global Automotive Market Share berdasarkan Brand*
(Statista, 2020)

Fungsi *spare part management* di PT. X merupakan tanggung-jawab dari Divisi *After Sales Service & Production* (ASSPD). ASSPD melakukan aktivitas

pemenuhan *order after sales service* domestik maupun ekspor ke berbagai *dealer* di seluruh Asia-Pasifik. Pada model bisnis *after sales service* yang diterapkan sekarang (**Gambar 1. 2**), PT. X melalui ASSPD bertanggung-jawab untuk: melakukan penanganan *order spare part*, melakukan pemesanan *spare part* kepada *supplier*, melakukan manajemen persediaan *spare part*, dan melakukan *finishing* serta *packing spare part* yang akan dikirim secara langsung ke *dealer* di berbagai negara atau melalui Distributor Sentral Asia-Pasifik di Thailand (PT. X, 2019).



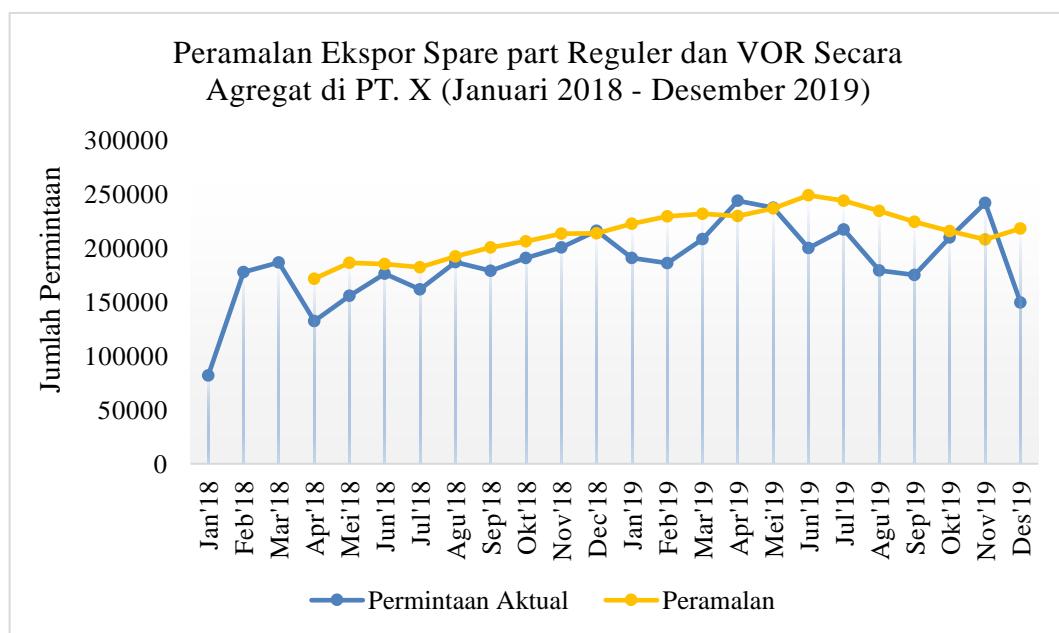
Gambar 1. 2 Proses Bisnis *After Sales Service* di PT. X
(PT. X, 2019)

Salah satu aktivitas penting yang menjadi bagian dari *spare part management* adalah peramalan permintaan (Jiang, et al., 2017). Peramalan permintaan berfungsi sebagai petunjuk bagi produsen untuk menjaga level ketersediaan *spare part* dan melakukan perencanaan produksi bersama dengan *supplier*. Semakin cepat dan akurat peramalan permintaan *spare part* yang dihasilkan, semakin efisien *lead time* pemenuhan *order* sehingga dapat menurunkan biaya-biaya yang berhubungan dengan penanganan *spare part*, seperti: *rework*, penyimpanan, dan transportasi (Klug, 2018).

Peramalan permintaan merupakan bagian dari aktivitas bulanan atau *Getsudo* di PT. X, yang berarti aktivitas yang dilakukan setiap bulan untuk memastikan kesiapan produksi di periode yang akan berjalan (PT. X, 2019). Pada ASSPD, peramalan permintaan *spare part* dilaksanakan oleh Departemen *Business Planning*. Pada departemen ini, peramalan digunakan untuk memperkirakan permintaan *spare part* bulanan sampai dengan 3 bulan ke depan ($T \sim T+3$), baik

untuk permintaan domestik atau *in-house* maupun ekspor, dan terkecuali untuk jenis permintaan tertentu.

Terdapat dua jenis permintaan yang ditangani oleh PT. X, yaitu *actual demand* dan *special* atau *created demand* (PT. X, 2019). *Special* atau *created demand* terdiri dari permintaan besar-besaran untuk armada perusahaan dan permintaan yang sewaktu-waktu diciptakan oleh *dealer* untuk keperluan promosi atau servis spesial. Jenis permintaan spesial tidak dapat dikontrol sehingga tidak dapat diramalkan. Sedangkan *actual demand* terdiri dari *regular demand*, yaitu permintaan *dealer* yang disuplai dari produksi reguler dan *vehicle-on-repair (VOR) demand*, yaitu permintaan prioritas yang harus segera dipenuhi karena terdapat *customer* yang kondisi kendaraannya rusak dan harus segera di servis namun *part* yang dibutuhkan tidak tersedia di *dealer* sehingga terjadi *delay*.



Gambar 1. 3 Peramalan *Existing* Yang Cenderung Melebihi Permintaan Aktual Ekspor *Spare Part* di PT. X (*Overforecast*)

Saat ini, fluktuasi permintaan *spare part* yang paling tinggi di PT. X berasal dari permintaan ekspor. Hal tersebut dibuktikan dengan hasil peramalan *regular* dan *VOR demand* yang deviasinya masih besar dari permintaan aktual untuk ekspor setiap bulannya (**Gambar 1. 3**). Sistem peramalan *spare part* yang belum akurat

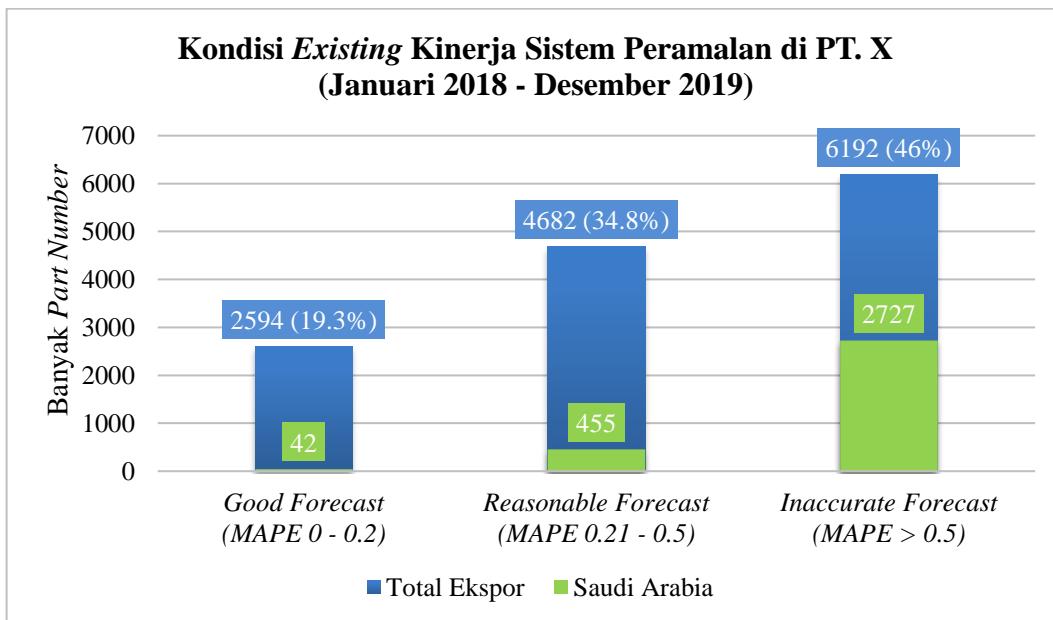
tersebut, ternyata memperparah problem lainnya yang sedang dihadapi oleh PT. X, seperti: tingkat *on-time delivery spare part* untuk ekspor yang rendah ketika terjadi *underforecast* karena tidak adanya *stock function* dan *inventory cost* yang semakin besar ketika terjadi *overforecast*, terutama pada *part* berdimensi besar, karena fasilitas gudang yang belum efisien (PT. X, 2019).

Sistem peramalan permintaan *spare part* yang tidak akurat disebabkan pemilihan metode peramalan yang kurang tepat dengan riwayat permintaan *spare part* yang dimiliki. Riwayat permintaan sepanjang periode tertentu akan membentuk suatu data *time-series*. Setiap data *time-series* memiliki pola dan komponen-komponen, yaitu: tren, siklus, musiman, dan *random*, yang berbeda-beda (Chopra & Meindl, 2016, Hyndman & Athanasopoulos, 2018). Selain itu, pola permintaan *spare part* seringkali lebih kompleks karena memiliki variasi yang besar dari periode ke periode bahkan pada beberapa periode bisa tidak ada permintaan atau bernilai nol (Altay & Litteral, 2011). Oleh karena itu, riwayat permintaan satu *spare part* dengan yang lainnya bisa sangat bervariasi sehingga satu metode peramalan belum tentu akurat apabila digunakan untuk semua pola permintaan *spare part* (Loukmidis & Luczak, 2006).

Tidak adanya sistem pengelompokan *spare part* berdasarkan pola permintaannya menjadi salah satu faktor penyebab belum ditemukannya metode peramalan yang tepat dan akurat dalam menangani permintaan berbagai *spare part* di PT. X. Oleh karena itu, selama ini metode analisis *time-series* sederhana, yaitu *simple moving average* dengan model MAX{MA(3),MA(6)}, berdasarkan nilai paling besar dari rata-rata tiga dan enam bulan ke belakang, digunakan untuk meramalkan permintaan bulanan semua *spare part* di PT. X. Hasilnya menunjukkan kinerja yang kurang baik, di mana peramalan terhadap sebagian besar *part* seringkali diestimasi terlalu tinggi dari permintaan aktualnya.

Penyimpangan hasil peramalan tersebut disebabkan oleh parameter rata-rata pada SMA yang sensitif terhadap *outlier* (Nijss & Klausen, 2013). Maka dari itu, metode SMA tidak cocok untuk sebagian besar pola permintaan *spare part* yang memiliki variasi besar sepanjang *time-series*. Akibatnya, kinerja peramalannya juga kurang memuaskan (**Gambar 1. 4**) setelah diukur dengan rata-rata persentase

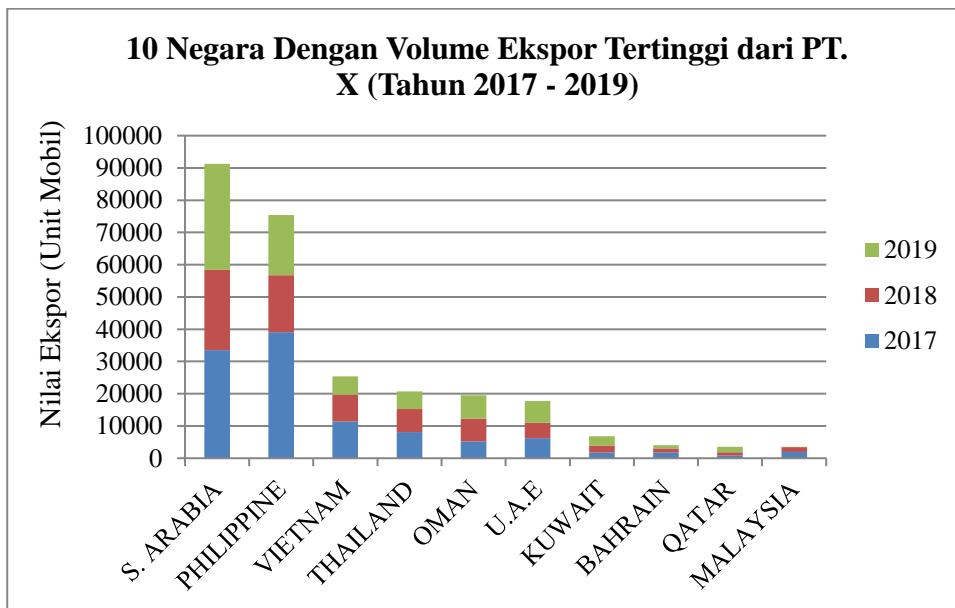
penyimpangan absolut (MAPE) dan berdasarkan kriteria “*Innacurate Forecast*” menurut (Lewis, 1982).



Gambar 1. 4 Kinerja Metode Peramalan SMA terhadap Total Permintaan Ekspor Spare Part di PT. X

Permintaan *spare part* yang kompleks, sebenarnya bisa diatasi dengan metode peramalan selain analisis *time-series*, seperti *data mining* dan metode modifikasi analisis *time-series* konvensional. Peramalan permintaan *spare part* dengan metode *data-mining* telah menunjukkan kesuksesan dan kinerja yang baik, seperti metode *Support Vector Machine (SVM)* dalam Hua & Zhang (2006), Jiang, et al. (2017), Kaya & Turkyilmaz (2018) dan *Artificial Neural Network (ANN)* dalam Şahin, et al. (2013) dan Henkelmann (2018). Metode modifikasi *time-series* seperti metode Croston pada Croston (1972) dan modifikasinya yaitu Syntetos-Boylan *Approximation (SBA)* pada Syntetos & Boylan (2005) dan Teunter-Syntetos-Babai Model (TSB) pada Teunter, et al. (2011) juga menunjukkan akurasi yang baik dalam meramalkan permintaan *spare part* dan mengungguli metode MA serta metode *time-series Single Exponential Smoothing (SES)*. Dari alternatif-alternatif metode tersebut, selanjutnya dapat dilakukan analisis perbandingan untuk

menentukan metode peramalan yang terbaik dalam menangani permintaan berbagai *spare part* di PT. X.



Gambar 1. 5 10 Negara Dengan Volume Ekspor Tertinggi Dari PT. X Selama Tahun 2017 – 2019

Segmen permintaan *spare part* dengan tujuan ekspor ke Arab Saudi selanjutnya dipilih sebagai pilot penelitian dalam tugas akhir ini. Segmen tersebut representatif dan menjadi prioritas karena volume ekspor produk PT. X ke Arab Saudi adalah yang tertinggi dibandingkan dengan negara lainnya (**Gambar 1. 5**) dan mayoritas *market share*nya, yaitu sebesar 27.7%, juga dikuasai oleh PT. X (Statista, 2020). Selain itu, *order spare part* dari Arab Saudi adalah *order* yang dipenuhi secara langsung oleh PT. X atau tidak melalui distributor sentral sehingga analisis dalam tugas akhir ini menjadi lebih spesifik karena karakteristik permintaan yang lebih homogen.

Secara garis besar, dalam tugas akhir ini pengelompokan berdasarkan Boylan, et al. (2005) dan pendekatan *data mining*, yaitu teknik klaster, akan diimplementasikan untuk mengelompokkan pola permintaan seluruh *part* dari segmen ekspor ke Arab Saudi selama Januari 2018 – Desember 2019. Kemudian pada masing-masing kelompok, akan dilakukan peramalan dengan metode Croston,

SBA, TSB, SVM, dan ANN. Pada akhirnya, kinerja semua metode akan dibandingkan dan divalidasi sehingga diperoleh metode peramalan yang lebih akurat terhadap permintaan aktual dan *robust* terhadap berbagai perubahan data permintaan untuk masing-masing kelompok *spare part*.

1.2 Rumusan Masalah

Berdasarkan latar belakang pada tugas akhir ini, maka rumusan masalah yang akan dibahas adalah bagaimana klasifikasi pola permintaan *spare part* dan teknik peramalan yang lebih akurat terhadap permintaan aktual dan *robust* terhadap berbagai perubahan data permintaan sehingga dapat memperbaiki sistem perencanaan stok *spare part* di PT. X.

1.3 Tujuan Penelitian

Tujuan yang akan dicapai oleh tugas akhir ini di antaranya:

1. Merancang sistem pengelompokan pola permintaan *spare part* yang dapat digunakan secara berkelanjutan pada operasional kegiatan *procurement spare part* di PT. X.
2. Membandingkan kinerja berbagai metode peramalan dan menentukan metode peramalan yang lebih baik untuk masing-masing klasifikasi pola permintaan *spare part*.
3. Membuat sistem peramalan yang lebih akurat terhadap permintaan aktual dan *robust* terhadap berbagai perubahan data permintaan untuk PT. X.

1.4 Manfaat Penelitian

Manfaat yang akan diperoleh dari hasil tugas akhir ini di antaranya:

1. Menjadi pilot penelitian dalam rencana perbaikan sistem perencanaan stok *after sales service* di PT. X.
2. Menjadi solusi untuk mengurangi risiko terjadinya *overforecast* atau *underforecast* dalam pengadaan stok *spare part* di PT. X dengan sistem peramalan yang lebih akurat terhadap permintaan aktual dan *robust* terhadap berbagai perubahan data permintaan.

3. Menjadi referensi bagi pengetahuan dan pengembangan riset selanjutnya dalam bidang peramalan permintaan *spare part* dan *data mining*.

1.5 Ruang Lingkup Penelitian

Ruang lingkup terdiri dari batasan dan asumsi yang digunakan dalam tugas akhir ini.

1.5.1 Batasan

Batasan yang digunakan dalam tugas akhir ini di antaranya:

1. Penelitian ini dilakukan pada PT. X yang bergerak di industri otomotif sebagai produsen mobil penumpang dan penyedia *original spare part* merek X.
2. Data riwayat permintaan *spare part* yang digunakan untuk peramalan berperiode bulanan dari Januari 2018 sampai Desember 2019 dan dibatasi pada permintaan reguler dan VOR untuk kebutuhan ekspor ke Arab Saudi.
3. Variabel-variabel eksogen yang mempengaruhi permintaan *spare part* tidak dipertimbangkan karena keterbatasan akses sehingga peramalan hanya dilakukan berdasarkan variabel tunggal (univariat), yaitu variabel permintaan *spare part* saja.
4. Pendekatan *time-series* yang dibandingkan adalah: pengelompokkan *part* dengan klasifikasi Boylan dan metode peramalan Croston serta modifikasinya.
5. Pendekatan *data mining* yang dibandingkan adalah: pengelompokkan *part* dengan teknik klaster aglomeratif dan K-Medoids serta metode peramalan *Support Vector Regression* (SVR) dan *Artificial Neural Network* (ANN).
6. Peramalan permintaan pada tugas akhir ini bertujuan untuk meramalkan permintaan mutlak dari *dealer* bukan untuk menentukan kebutuhan stok *spare part* di PT. X.

1.5.2 Asumsi

Pada tugas akhir ini diasumsikan tidak ada perubahan metode peramalan *existing* di PT. X.

1.6 Sistematika Penulisan

Penulisan dalam tugas akhir ini disusun dalam enam bab yang secara garis besar dideskripsikan sebagai berikut.

BAB 1 PENDAHULUAN

Bab ini menjelaskan tentang: latar belakang dilakukannya tugas akhir ini, rumusan masalah yang akan diselesaikan, tujuan yang akan dicapai, manfaat yang diperoleh dari hasil tugas akhir ini, ruang lingkup tugas akhir, dan penjelasan mengenai sistematika penulisan.

BAB 2 TINJAUAN PUSTAKA

Bab ini menjelaskan tentang beberapa landasan teori yang relevan dengan topik tugas akhir ini, seperti: *spare part management*, klasifikasi pola permintaan *spare part* yang dibedakan menjadi pendekatan Boylan dan teknik klaster, peramalan *spare part* berdasarkan *time-series* menggunakan metode Croston dan modifikasinya, peramalan *spare part* dengan pendekatan *data mining* menggunakan SVM dan ANN, pengukuran kinerja peramalan, dan validasi peramalan. Selain itu, di akhir bab juga dijelaskan posisi tugas akhir ini terhadap penelitian-penelitian serupa yang telah dilakukan sebelumnya.

BAB 3 METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metodologi yang dilaksanakan dalam tugas akhir ini yang dipaparkan dalam bentuk *flowchart* beserta penjelasannya. Secara garis besar, metodologi tersebut terdiri dari tujuh tahap yaitu: analisis kondisi *existing*, pra-pengolahan data, pengelompokkan data, peramalan, validasi, interpretasi dan analisis, dan penarikan kesimpulan dan saran.

BAB 4 PRA-PENGOLAHAN DAN PENGELOMPOKKAN DATA

Bab ini menjelaskan tentang proses pra-pengolahan dan pengelompokkan data riwayat permintaan *spare part* di PT. X. Pra-pengolahan data terdiri dari proses seleksi data riwayat permintaan *spare part* sesuai dengan segmentasi permasalahan, pembersihan data (data *cleaning*) riwayat permintaan yang volumenya tidak

signifikan, dan transformasi data yang digunakan sebagai *input* pengelompokan dan peramalan. Data yang sudah siap kemudian dikelompokkan dengan pendekatan Boylan dan teknik klaster, yaitu K-Medoids *Clustering* dan *Agglomerative Hierarchical Clustering*. Pada akhirnya akan diterapkan berbagai eksperimentasi pada data hasil pengelompokan sebelum digunakan sebagai *input* peramalan di bab selanjutnya.

BAB 5 PERAMALAN SPARE PART DAN VALIDASI

Bab ini menjelaskan tentang proses pengembangan model peramalan Croston, SBA, TSB, SVR dan ANN, dan uji pengaruh berbagai skenario pengelompokan *part* dan eksperimentasi *input* terhadap kinerja peramalan. Pada bab ini, model peramalan terbaik akan dipilih berdasarkan parameter *forecasting error* dan *robustness* di tahap *training* dan *testing*. Pada akhirnya, Metode usulan terbaik akan diimplementasikan untuk peramalan multi-periode dan dibandingkan kinerjanya dengan metode peramalan *existing*.

BAB 6 INTERPRETASI DAN ANALISIS

Bab ini menjelaskan tentang interpretasi hasil dari berbagai skenario pengelompokan dan peramalan *spare part* yang telah dilakukan. Masing-masing skenario tersebut kemudian dianalisis kinerjanya dan dibandingkan satu sama lain sehingga diperoleh skenario pengelompokan dan peramalan terbaik sebagai rekomendasi perbaikan sistem peramalan di PT. X.

BAB 7 KESIMPULAN DAN SARAN

Bab ini menjelaskan tentang kesimpulan-kesimpulan yang diperoleh dari penelitian yang sudah dilakukan untuk menjawab tujuan pada tugas akhir ini. Selain itu, usulan perbaikan untuk PT. X dan rekomendasi pengembangan untuk penelitian selanjutnya juga akan disampaikan pada saran.

BAB 2

TINJAUAN PUSTAKA

BAB 2

TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan mengenai landasan teori, pustaka, dan penelitian-penelitian terdahulu yang relevan dengan topik pada tugas akhir ini.

2.1 *Spare Part Management*

Spare Part disebut juga sebagai produk sekunder, yaitu produk yang dapat diperbaiki atau diganti untuk mengembalikan fungsi dari produk utama (Schröter, 2006). Terdapat dua macam *spare part*, yaitu *repairable part* dan *consumable part* (Driessen, et al., 2014). *Repairable part* adalah *part* yang dapat berfungsi kembali setelah diperbaiki sehingga tidak perlu dilakukan pengadaan *part* pengganti. Oleh karena itu, mayoritas permintaan *spare part* berasal dari *consumable part*.

Pada mobil penumpang, *consumable part* termasuk, namun tidak terbatas pada: *oil filter*, *air filter*, *wiper*, *bearing*, *gasket*, dan fluida. *Part* tersebut perlu diganti secara berkala karena mengalami keausan atau habis selama penggunaan mobil dari waktu ke waktu. Oleh karena itu, harus selalu tersedia ketika dibutuhkan. Ketersediaan *part* bisa dipasok dari berbagai sumber. Menurut Klug (2018), berdasarkan sumbernya *part* bisa dibedakan menjadi:

- ***Original Part***

Original part adalah *part* yang dipasok dari produsen produk utama yang diproduksi sendiri atau melalui *supplier*-nya (OEM). Ketersediaan *part original* pada umumnya dijamin oleh produsen dan menjadi bagian dari fasilitas bagi konsumen.

- ***Foreign Part***

Foreign part adalah *part* yang diproduksi oleh manufaktur di luar pemilik merek produk utama dan OEM.

- ***Used Part***

Used part adalah *part* yang sudah pernah terpakai kemudian diperbarui atau didaur-ulang.

Segala aktivitas manajemen untuk memastikan ketersediaan *part* pada waktu, kualitas, dan kuantitas yang tepat disebut dengan *spare part logistics* atau *spare part management* (Biedermann, 2008). Menurut Schuh & Stich (2013), dalam perspektif manufaktur, *spare part management* bertujuan untuk mencapai stok *spare part* yang optimal secara ekonomis. Namun pada kenyataannya, untuk mengatasi tingginya ketidakpastian permintaan *spare part*, sebagian besar produsen cenderung melebihi stok untuk berjaga-jaga (*security stock*) sehingga berdampak pada biaya *capital* yang tinggi (Schröter, 2006). Oleh karena itu, estimasi permintaan *spare part* yang akurat diperlukan untuk mengurangi kebutuhan *security stock*.

Pada umumnya di industri otomotif, *spare part management* adalah bagian dari *after sales service*. Menurut Klug (2018), *after sales service* merupakan salah satu alat *marketing* yang terdiri dari semua aktivitas untuk mempertahankan kesetiaan *customer* terhadap *brand* setelah pembelian suatu produk. *Spare Part management* yang optimal akan memberikan kepuasan kepada *customer* karena tidak adanya *delay* dalam pemenuhan *order*. *Customer* yang puas berpotensi merekomendasikan *brand* kepada calon *customer* baru. Oleh karena itu, Klug (2018) menyimpulkan bahwa *spare part management* menjadi faktor kompetitif yang penting di industri otomotif.

2.2 Klasifikasi Permintaan *Spare Part*

Berdasarkan Loukmidis & Luczak (2006), tidak semua teknik prediksi bisa diterapkan pada semua pola permintaan. Oleh karena itu, sebelum memilih teknik prediksi yang tepat, diperlukan adanya pemisahan masing-masing *spare part* sesuai dengan pola permintaannya (Klug, 2018). Pengklasifikasian *spare part* mengikuti pola permintaan merupakan salah satu pendekatan dari pengklasifikasian *spare part* menurut (Loukmidis, 2014).

Classification Attribute	Reference Value	Spare Part Categories
Relevance	Business Impact, Criticality Price, Revenue, Demand, Volume, Risk (functional, distributional) etc.	ABC (high, medium, low)
Lifecycle	Primary Product Lifecycle Demand per time period	Initial Phase, Normal Phase, Final Phase
Demand Characteristics	Demand Frequency Demand Regularity Demand Patterns Irregularity, Seasonality and Trend Attributes	Slow and Fast moving parts XYZ (steady, volatile, intermittent) smooth, irregular, slow moving, mildly intermittent and highly intermittent Smooth, erratic, intermittent, lumpy 6 Demand Distribution Shapes Stationary, sporadic, seasonal, trending

Gambar 2. 1 Berbagai Pendekatan Untuk Klasifikasi Spare Part

(Loukmidis, 2014)

Salah satu pendekatan yang paling populer adalah pendekatan (Boylan, et al., 2005). Dalam pendekatan tersebut, Boylan, et al, membagi *spare part* berdasarkan empat karakteristik permintaan, yaitu: *smooth*, *intermittent*, *erratic*, dan *lumpy*. Selain pendekatan tersebut, tugas membagi data ke dalam kelompok-kelompok yang lebih kecil di mana anggota di dalamnya memiliki karakteristik yang mirip bisa juga dilakukan dengan teknik klaster (Santosa & Umam, 2018). Teknik klaster merupakan salah satu teknik yang banyak dipakai dalam *data mining*. Teknik klaster pada umumnya digunakan untuk mengelompokkan data tanpa memerlukan label yang telah ditentukan sebelumnya (*pre-determined label*). Pendekatan Boylan dan teknik klaster akan dijelaskan lebih lanjut pada paragraf setelah ini.

2.2.1 Pendekatan Boylan

Pendekatan Boylan, et al. (2005) pada dasarnya merupakan klasifikasi karakteristik permintaan yang menggabungkan pendekatan *variance partition* dari Williams (1984), pendekatan Eaves & Kingsman (2004), dan *intermittent demand estimation procedure* dari Croston (1972). Seluruh pendekatan tersebut bertujuan untuk mengidentifikasi perbedaan pada pola permintaan yang sama-sama terlihat

sporadis. Hasilnya, pendekatan Boylan membedakan pola permintaan tersebut menjadi empat, yaitu: *smooth*, *intermittent*, *erratic*, dan *lumpy*.

- ***Smooth***

Ketika pola permintaan memiliki variabilitas dan periode antara dua permintaan bukan nol berturut-turut yang kecil, maka permintaan tersebut merupakan *smooth demand*.

- ***Intermittent***

Ketika pola permintaan memiliki persentase nilai nol yang besar dan nilai nol tersebut tersebar secara random, maka permintaan tersebut merupakan *intermittent demand*.

- ***Erratic***

Ketika pola permintaan memiliki variabilitas yang tinggi tetapi periode antara dua permintaan bukan nol berturut-turutnya kecil, maka permintaan tersebut merupakan *erratic demand*.

- ***Lumpy***

Lumpy demand terjadi ketika pola permintaan memiliki variabilitas dan periode antara dua permintaan bukan nol berturut-turut yang besar. *Lumpy demand* merupakan pola permintaan yang paling sulit diprediksi.

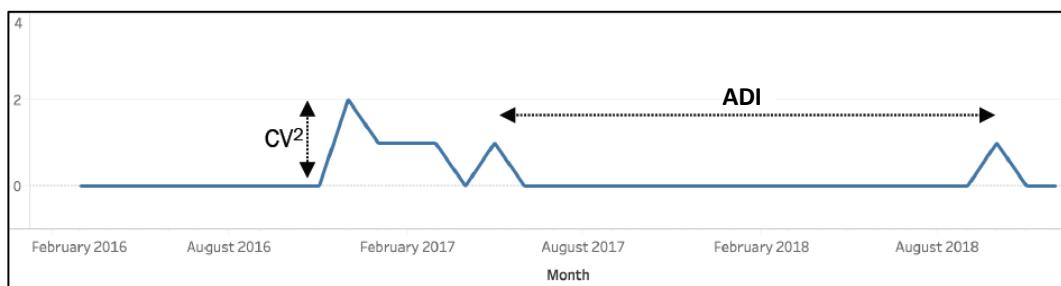
Skema pengelompokan Boylan menggunakan dua parameter, yaitu *Squared Coefficient of Variation (CV²)* dan *Average Inter-Demand Interval (ADI)*, yang diturunkan dari ketiga pendekatan sebelumnya.

$$CV^2 = \left(\frac{\sqrt{\frac{\sum_{i=1}^N (\varepsilon_i - \bar{\varepsilon})^2}{N}}}{\bar{\varepsilon}} \right)^2 \quad (2.1)$$

$$\text{Di mana } \bar{\varepsilon} = \frac{\sum_{i=1}^N \varepsilon_i}{N}$$

$$ADI = \frac{\sum_{i=1}^n p_i}{n} \quad (2.2)$$

Parameter N adalah jumlah periode permintaan yang diobservasi, sedangkan n adalah jumlah periode permintaan yang bukan nol. Kemudian parameter ε_i adalah nilai permintaan pada periode i dan $\bar{\varepsilon}$ adalah nilai rata-rata permintaan yang diobservasi, dan p_i adalah jumlah periode antara dua permintaan bukan nol berturut-turut. Dalam penelitiannya, Chawla & Miceli (2019) memberikan gambaran mengenai dua parameter tersebut.



Gambar 2. 2 Ilustrasi Parameter CV^2 dan ADI

(Chawla & Miceli, 2019)

Boylan, et al. (2005) menyarankan ($CV^2 = 0.49$ dan $ADI = 1.32$) sebagai nilai *cut-off* dalam mengklasifikasikan pola permintaan sebagai *smooth*, *intermittent*, *erratic*, atau *lumpy*.

Tabel 2. 1 Nilai *Cut-Off* untuk CV^2 dan ADI

	$ADI \leq 1.32$	$ADI > 1.32$
$CV^2 > 0.49$	<i>Erratic</i>	<i>Lumpy</i>
$CV^2 \leq 0.49$	<i>Smooth</i>	<i>Intermittent</i>

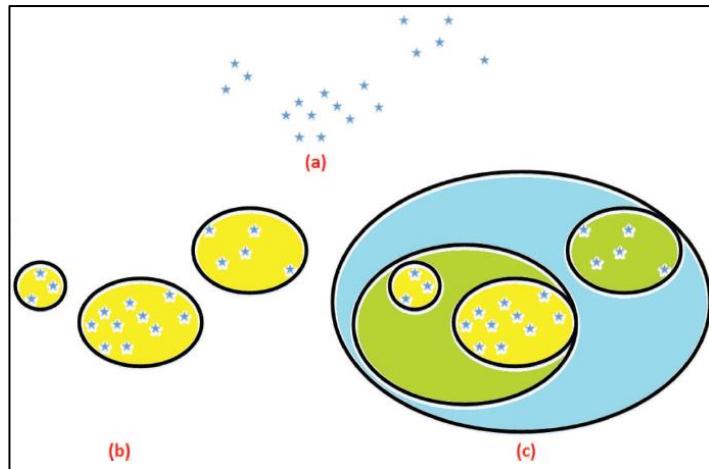
(Boylan, et al., 2005)

2.2.2 Teknik Klaster

Tujuan utama dari analisis klaster adalah untuk menempatkan objek yang saling mirip (jaraknya saling dekat) ke dalam satu klaster atau kelompok dan membuat jarak antara klaster yang saling beda sejauh mungkin (Santosa & Umam, 2018). Analisis klaster merupakan teknik *unsupervised learning* yang paling umum digunakan dalam *data mining*. Menurut Santosa & Umam (2018), yang termasuk pendekatan *unsupervised learning* adaah metode-metode yang tidak membutuhkan

contoh dalam fase *learning* atau dengan kata lain tidak memerlukan *pre-determined label* dari data yang diobservasi. Berdasarkan strateginya, secara garis besar teknik klaster dapat dikelompokkan ke dalam empat kategori, yaitu: (1) metode *partitioning*, (2) metode hierarkis, (3) *density-based method*, dan (4) *grid-based method*. Metode *partitioning* dan hierarkis merupakan metode yang paling umum digunakan (Roelofsen, 2018).

Metode *partitioning* membagi data ke dalam sejumlah klaster yang banyaknya ditentukan di awal sebagai K , kemudian merelokasi data-data tersebut secara iteratif sehingga masing-masing data berada tepat di salah satu klaster atau tidak saling *overlap* (*hard clustering*). Berbeda dengan metode *partitioning*, metode hierarkis menghasilkan set klaster sebanyak N dari set data berukuran N yang saling berhubungan dalam bentuk pohon klaster. Perbedaan dari kedua metode tersebut ditunjukkan dalam (Kutbay, 2018).



Gambar 2. 3 (a) Data Awal, (b) Implementasi *Partitional Clustering*, (c) Implementasi *Hierarchical Clustering*
(Kutbay, 2018)

Menurut Sardá-Espinosa (2018), metode *partitioning* termasuk ke dalam problem optimasi kombinatorial. Dalam memperoleh solusi global, seluruh kombinasi klaster yang mungkin harus dicoba sehingga memerlukan waktu komputasi yang sangat lama. Oleh karena itu, metode-metode heuristik dikembangkan sebagai alternatif, contohnya: k-means, k-median, dan K-Medoids.

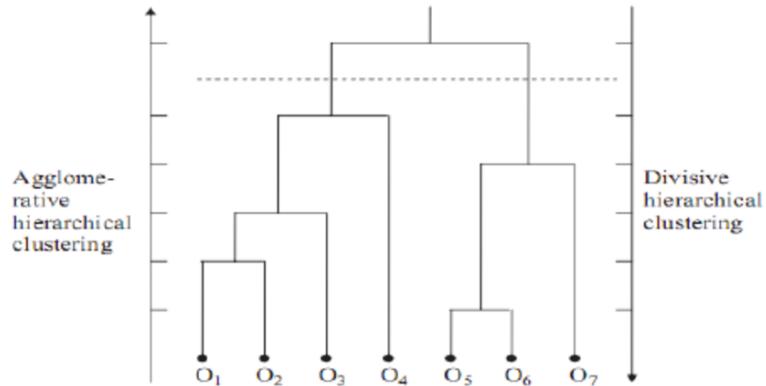
Akibatnya, solusi yang dihasilkan adalah solusi lokal (Santosa & Umam, 2018). Metode heuristik tersebut dikembangkan berdasarkan perbedaan *cluster center* yang digunakan. Dari ketiga *cluster center* tersebut, median dan medoids adalah parameter yang paling *robust* terhadap *outliers* dan *noises* (Santosa & Umam, 2018). Berdasarkan Santosa & Umam (2018), medoids adalah nilai yang diambil langsung dari set data, akan tetapi yang dapat memberikan nilai rata-rata perbedaan (*dissimilarity*) paling kecil.

Jarak intra-klaster adalah salah satu ukuran ketidakmiripan (*dissimilarity*) yang digunakan untuk menggabungkan objek-objek menjadi satu klaster. Dalam ukuran tersebut, nilai jarak antara data terhadap pusat klasternya atau *centroid*-nya, yang dilambangkan dengan $d(x_{ij}, C_{lj})$, diharapkan seminimal mungkin (Santosa & Umam, 2018). Beberapa ukuran jarak yang sering digunakan diantaranya: (1) jarak Euclidean (L₂-norm), (2) jarak Manhattan atau Cityblock (L₁-norm), (3) jarak Minkowski, dan (4) jarak Chebyshev atau Supremum (L_∞-norm). Dalam menentukan jumlah K di awal, terdapat beberapa alternatif pendekatan yang bisa dipilih, seperti: *Rule of Thumb*, *Elbow Method*, pendekatan menggunakan *Information Criterion*, parameter *Silhouette*, *Cross-Validation*, Calinski-Harabasz *Criterion*, dan *Gap Statistics*. Pendekatan-pendekatan tersebut diterangkan dengan cukup jelas dalam Kodinariya & Makwana (2013) dan Roelofsen (2018).

Berdasarkan Santosa & Umam (2018), terdapat dua pendekatan metode klaster hierarkis, yaitu pendekatan Aglomeratif dan divisif. Pendekatan Aglomeratif adalah pendekatan yang bersifat *bottom-up* karena klaster-klaster yang berukuran kecil, yang berisi objek-objek yang individual, diaglomerasikan menjadi klaster yang lebih besar. Sedangkan pendekatan divisif bersifat *top-down*, di mana pengklasteran dimulai dengan satu klaster besar yang berisi semua objek, kemudian klaster ini dibagi menjadi beberapa sub-klaster yang lebih kecil. Pendekatan divisif cenderung tidak praktis dan memerlukan waktu dan biaya komputasi yang lebih banyak daripada pendekatan Aglomeratif.

Metode hierarkis mengelompokkan objek-objek ke dalam pohon klaster, yang disebut dendogram. Dendogram menampilkan hubungan antara klaster, di mana pada akar pohon (atas) mewakili satu klaster, yang berisi semua objek, dan

pada daun pohon (bawah) terdapat N klaster, yang berisi satu objek. Dengan memotong pohon pada level tertentu yang diinginkan, akan diperoleh pengelompokan objek-objek ke dalam sejumlah K klaster yang diinginkan.



Gambar 2. 4 Contoh Dendogram Dari Klaster Hierarkis
(Sembiring, et al., 2010)

Ukuran yang memisahkan antara klaster pada dendogram adalah jarak antara klaster yang disebut sebagai *linkage*. Dari perhitungan *linkage* tersebut, matriks kemiripan (*similarity*) akan terbentuk dan klaster-klaster yang paling berdekatan akan digabungkan menjadi satu klaster yang lebih besar. Pada Santosa & Umam (2018) dijelaskan beberapa macam *linkage* yang dapat digunakan, yaitu:

1. Jarak maksimum antara objek dalam klaster (*complete linkage*)

Dengan cara ini jarak antara dua klaster didefinisikan sebagai:

$$d(A, B) = \max_{x \in A, y \in B} \{S_{xy}\} \quad (2.3)$$

di mana S_{xy} adalah jarak dua objek x dan y masing-masing dari klaster A dan B.

2. Jarak minimum antara objek dari setiap klaster (*single linkage*)

Dengan cara ini jarak antara dua klaster didefinisikan sebagai:

$$d(A, B) = \min_{x \in A, y \in B} \{S_{xy}\} \quad (2.4)$$

3. Rata-rata jarak antara objek dari setiap klaster (*average linkage*)

$$d(A, B) = \frac{1}{n_A n_B} \sum_{x \in A} \sum_{y \in B} s(x, y) \quad (2.5)$$

di mana n_A dan n_B masing-masing adalah banyaknya data dalam set A dan B

4. *Centroid linkage*

Dengan cara ini, jarak antara dua klaster didefinisikan sebagai:

$$d(A, B) = s(\bar{x}, \bar{y}) \quad (2.6)$$

di mana $\bar{x} = \frac{1}{n_A} \sum_{x \in A} x$ dan $\bar{y} = \frac{1}{n_B} \sum_{y \in B} y$

5. *Ward linkage*

$$d(A, B) = \frac{n_A n_B s^2_{AB}}{n_A + n_B} \quad (2.7)$$

di mana s^2_{AB} adalah jarak antara klaster A dan B menggunakan *centroid linkage*.

Akurasi dari suatu *linkage* dapat dihitung dengan koefisien korelasi *cophenet* (c) sebagai berikut.

$$c = \frac{\sum_{i < j} (Y_{ij} - y)(Z_{ij} - z)}{\sqrt{\sum_{i < j} (Y_{ij} - y)^2 \sum_{i < j} (Z_{ij} - z)^2}} \quad (2.8)$$

Koefisien tersebut mendefinisikan nilai korelasi linear antara jarak *linkage* (Z_{ij}) antara klaster, yang diperoleh dari pohon klaster, dan jarak asli antara objek (Y_{ij}), atau ukuran *dissimilarity*, yang digunakan untuk membangun pohon klaster. y dan z

y berturut-turut adalah rata-rata dari kedua ukuran jarak tersebut. Ukuran *linkage* yang paling baik adalah yang memiliki korelasi yang paling kuat dengan ukuran *dissimilarity* yang digunakan.

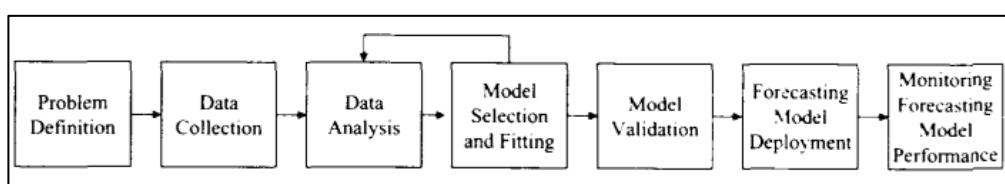
Menurut Kutbay (2018), kebaikan solusi klaster yang diperoleh dapat diukur dengan menggunakan *criterion function*. Salah satu ukuran *criterion function* yang paling sering digunakan adalah *sum of squared error* (SSE).

$$SSE = \sum_{i=1}^k \sum_{x \in D_i} \|x - C_i\|^2 \quad (2.9)$$

Dalam suatu klaster D_i , vektor C_i adalah nilai *centroid* yang paling baik untuk mewakili data-data dalam klaster tersebut, yaitu yang meminimalkan nilai dari SSE (Santosa & Umam, 2018). Oleh karena itu, secara umum, semakin kecil nilai SSE, maka semakin baik solusi klasternya.

2.3 Peramalan Spare Part

Peramalan merupakan teknik statistik yang paling sering digunakan dalam bisnis. Berdasarkan Hyndman & Athanasopoulos (2018), teknik peramalan adalah teknik untuk memprediksi kejadian di masa depan, dengan seakurat mungkin menggunakan seluruh data dan pengetahuan mengenai kejadian tersebut. Selain itu, peramalan juga mampu mengukur ketidakpastian secara kuantitatif sehingga bermanfaat dalam pengambilan keputusan bisnis. Secara umum proses peramalan *time-series* dapat dilihat pada **Gambar 2. 5**.



Gambar 2. 5 Urutan Proses Peramalan *Time-Series*
(Montgomery, et al., 2008)

Menurut Jiang, et al. (2017), peramalan permintaan merupakan aktivitas yang penting dalam *spare part management*. Berdasarkan Hyndman &

Athanassopoulos (2018), rentang peramalan dapat dibedakan menjadi: *short-term*, *medium-term*, dan *long-term/strategic forecasting*. Peramalan permintaan *spare part* termasuk ke dalam *short-term forecasting*. Banyak literatur menganggap permintaan *spare part* sebagai kasus khusus dalam studi peramalan. Hua & Zhang (2006) menganggap hal tersebut karena karakteristik khusus yang dimiliki oleh *spare part*, yaitu:

1. *Spare Part* memiliki permintaan yang sporadis. Hal tersebut ditunjukkan dengan nilai permintaan yang sangat fluktuatif (*erraticness*) dan atau tidak kontinyu (*intermittence*).
2. Data riwayat permintaan *spare part* cenderung sedikit atau terbatas akibat kemunculan permintaan yang tidak kontinyu.
3. Permintaan *spare part* seringkali relevan bahkan dependen terhadap beberapa variabel lainnya, seperti data *maintenance*.

Oleh karena itu, banyak riset telah mengembangkan metode peramalan untuk permintaan *spare part*. Beberapa dari metode tersebut berdasarkan pada pendekatan *analisis time-series* dan *data mining* (Lee & Kim, 2018). Kedua pendekatan tersebut akan dijelaskan lebih lanjut pada paragraf setelah ini.

2.3.1 Pendekatan Time-Series

Kebanyakan dari kasus peramalan melibatkan penggunaan data *time-series*. Berdasarkan Montgomery, et al. (2008), data *time-series* adalah serangkaian nilai dari suatu variabel yang diukur secara berurutan mengikuti variabel waktu tertentu. Data *time-series* di mana hanya ada satu variabel yang diukur disebut juga dengan *univariate time-series*, sedangkan apabila lebih dari satu disebut dengan *multivariate time-series*. Dalam Hyndman & Athanassopoulos (2018) dijelaskan bahwa, data *time-series* terdiri dari komponen tren, musiman, dan siklik.

Tren adalah kenaikan atau penurunan yang konsisten dalam jangka panjang pada suatu *time-series*, tren bisa berbentuk linear atau non-linear. Musiman atau *seasonal* adalah fluktuasi pada suatu *time-series* yang berulang dengan periode yang tetap, fluktuasi tersebut disebabkan oleh faktor-faktor yang terjadi dalam jangka waktu yang pendek, seringkali satu tahun. Siklik adalah fluktuasi yang terjadi secara tidak tetap dalam jangka waktu yang lebih lama dari *seasonal*,

seringkali dua tahun atau lebih. Pada Adhikari & Agrawal (2013) dijelaskan bahwa dalam suatu data *time-series* juga terdapat komponen *irregular* atau *random*. *Random* dalam *time-series* adalah fluktuasi yang tidak teratur akibat faktor-faktor yang tidak bisa dikontrol. Berdasarkan keempat komponen tersebut, secara garis besar suatu nilai *time-series* dapat didefinisikan dengan model aditif atau multiplikatif sebagai berikut.

$$\text{Model Aditif} \quad : Y(t) = T(t) + S(t) + C(t) + I(t) \quad (2.10)$$

$$\text{Model Multiplikatif} : Y(t) = T(t) \times S(t) \times C(t) \times I(t) \quad (2.11)$$

Dalam model tersebut $Y(t)$ adalah suatu nilai dalam *time-series*, $T(t)$ adalah tren, $S(t)$ adalah *seasonal*, $C(t)$ adalah siklik, dan $I(t)$ adalah komponen *irregular*. Pada model aditif diasumsikan bahwa keempat komponen tersebut saling independen, sedangkan pada model multiplikatif, keempat komponen tersebut dapat saling mempengaruhi satu sama lain.

Berdasarkan Montgomery, et al. (2008), komponen-komponen tersebut juga bisa dipisahkan dari data *time-series*, dengan melakukan *adjustment*, untuk menyederhanakan peramalan sesuai dengan kebutuhan. Selain *adjustment*, peramalan *time-series* juga bisa disederhanakan dengan melakukan transformasi stasioner. Terdapat dua macam transformasi stasioner, yaitu transformasi stasioner varian dan rataan.

Menurut Hyndman & Athanasopoulos (2018), transformasi varian berguna ketika data menunjukkan varian yang tidak konstan atau berubah-ubah sepanjang *time-series*. Beberapa metode untuk transformasi varian dalam *time-series* yaitu (Hyndman & Athanasopoulos, 2018):

1. Logarithms Transformation

Pada metode ini data y_1, \dots, y_T ditransformasi menjadi w_1, \dots, w_T dengan fungsi:

$$w_t = \log_b(y_t) \quad (2.12)$$

Kelebihan dari transformasi logaritma adalah bentuk datanya yang mudah diinterpretasikan karena perubahan pada nilai logaritma adalah perubahan relatif (berupa persentase) dari data aslinya. Selain itu, transformasi ini juga membatasi data tetap positif pada skala aslinya.

2. Power Transformation

Transformasi ini adalah alternatif dari transformasi logaritma. Pada metode ini data y_1, \dots, y_T ditransformasi menjadi w_1, \dots, w_T dengan fungsi:

$$w_t = y_t^p \quad (2.13)$$

3. Box-Cox Transformation

Transformasi Box-Cox adalah transformasi yang menggabungkan kedua transformasi sebelumnya. Pada metode ini, data y_1, \dots, y_T ditransformasi menjadi w_1, \dots, w_T bergantung pada parameter lambda (λ) dan didefinisikan sebagai berikut.

$$w_t = \begin{cases} \ln(y_t), & \lambda = 0 \\ \frac{y_t^\lambda - 1}{\lambda}, & \lambda \neq 0 \end{cases} \quad (2.14)$$

Data transformasi w_1, \dots, w_T juga bisa dikembalikan menjadi y_1, \dots, y_T dengan membalik fungsi Box-Cox menjadi:

$$y_t = \begin{cases} \exp(w_t), & \lambda = 0 \\ (\lambda w_t + 1)^{1/\lambda}, & \lambda \neq 0 \end{cases} \quad (2.15)$$

Sedangkan transformasi rataan bisa dilakukan dengan *differencing*, yaitu menghitung perubahan di antara observasi pada *time-series*. Data hasil *differencing* memiliki rataan yang stabil. Terdapat dua macam *differencing* yaitu *order differencing* dan *seasonal differencing* (Hyndman & Athanasopoulos, 2018):

1. First-Order dan Second-Order Differencing

Differencing orde pertama adalah sebagai berikut di bawah ini. Di mana y adalah observasi pada periode t dan $t-1$.

$$y_t^{(1)} = y_t - y_{t-1} \quad (2.16)$$

Pada beberapa kasus, *differencing* orde pertama tidak memberikan data yang stasioner dalam rataan sehingga perlu dilakukan *differencing* orde kedua.

$$y_t^{(2)} = y_t - 2y_{t-1} + y_{t-2} \quad (2.17)$$

2. Seasonal Differencing

Pada *time-series* dengan *seasonality* yang kuat, seringkali *seasonal differencing* perlu dilakukan terlebih dahulu sebelum *order differencing*. *Seasonal differencing* adalah perubahan suatu observasi pada periode tertentu dengan observasi di periode yang sama di tahun sebelumnya. *Seasonal differencing* orde pertama adalah sebagai berikut, di mana m untuk data bulanan adalah (m = 12).

$$y_t^{(1)} = y_t - y_{t-m} \quad (2.18)$$

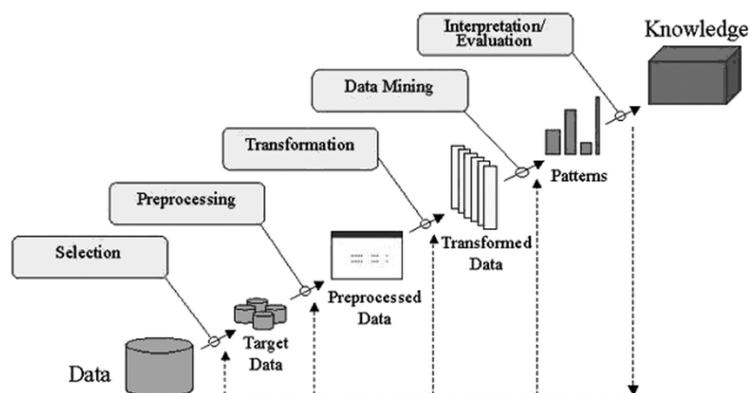
Menurut Hamilton (1994), peramalan dengan pendekatan *time-series* memiliki asumsi dasar bahwa pola permintaan di periode sebelumnya akan berulang di periode selanjutnya. Akibatnya, hal tersebut seringkali menjadi kelamahan dari analisis *time-series*. Beberapa metode peramalan *time-series* diantaranya adalah *Moving Average* (MA), *Autoregressive* (AR), dan *Simple Exponential Smoothing* (SES). Metode-metode tersebut tidak cocok terhadap karakteristik permintaan *spare part*. Hal ini dibuktikan oleh Croston (1972) di mana metode SES seringkali mengestimasi terlalu tinggi pada permintaan *spare part* yang bersifat *lumpy* karena riwayat permintaan terakhir diberikan bobot yang paling besar.

Oleh karena itu, Croston mengembangkan metode baru dari modifikasi SES yang dikenal dengan metode Croston. Kemudian Syntetos & Boylan (2005) membuktikan bahwa metode Croston ternyata bias terhadap beberapa karakteristik permintaan *spare part*. Maka dari itu, keduanya memodifikasi metode Croston dan mengusulkan metode yang dikenal dengan *Syntetos Boylan Approximation* (SBA). Selain SBA, dikenal juga model TSB yang diciptakan oleh Teunter, Syntetos, dan Babai sebagai usulan perubahan dari metode Croston untuk menghindari peramalan

yang cenderung konstan pada permintaan yang mengandung nol berperiode panjang (karena metode Croston hanya memperbarui peramalan ketika terjadi permintaan) (Teunter, et al., 2011). Metode Croston dan modifikasinya akan dijelaskan pada sub-bab berikutnya.

2.3.2 Pendekatan Data Mining

Berdasarkan Santosa & Umam (2018), *data mining* adalah kegiatan mengekstrak informasi atau pengetahuan penting dari suatu set data berukuran besar dengan menggunakan teknik tertentu. *Data mining* juga merupakan gabungan dari berbagai disiplin ilmu seperti statistik dan *machine learning*. Secara umum, proses-proses yang dilibatkan dapat dilihat pada **Gambar 2. 6**.



Gambar 2. 6 *Knowledge Discovery in Database (KDD) Process*
(Piatesky-Shapiro, et al., 1996)

Menurut Piatesky-Shapiro, et al. (1996), ciri-ciri problem yang dapat diselesaikan dengan data mining adalah:

1. Memerlukan keputusan yang bersifat *knowledge-based*.
2. Mempunyai lingkungan yang berubah-ubah.
3. Metode yang ada sekarang belum optimal.
4. Tersedia data yang bisa diakses, cukup, dan relevan.
5. Memberikan keuntungan yang tinggi jika keputusan yang diambil tepat.

Tugas-tugas yang bisa dilakukan oleh *data mining* diantaranya: (1) klasterisasi, (2) prediksi, dan (3) asosiasi. Peramalan merupakan salah satu

penerapan dari model prediksi dalam *data mining*. Berdasarkan Santosa & Umam (2018), model prediksi termasuk *supervised learning*, yaitu metode pembelajaran yang memerlukan contoh solusi atau *target* dari sejumlah variabel yang diamati (*predictor*), sehingga data *time-series* perlu disesuaikan mengikuti bentuk *predictor* dan *target*. Berdasarkan outputnya, model prediksi bisa dibedakan menjadi model klasifikasi dan estimasi. Output klasifikasi adalah nilai diskrit, sedangkan output estimasi adalah nilai kontinyu (Santosa & Umam, 2018). Peramalan permintaan termasuk ke dalam problem estimasi.

Beberapa riset telah menunjukkan keunggulan *data mining* dalam melakukan peramalan dibandingkan dengan metode lainnya. Menurut Kaya & Turkyilmaz (2018), keunggulan tersebut disebabkan sebagian besar metode *data-mining* mampu mengatasi non-linearitas dan *noise* dalam data. Selain itu, *data mining* memiliki kemampuan *learning*, sehingga mampu memperbaiki keakuratan prediksinya secara terus-menerus. Beberapa metode *data mining* yang sering digunakan dalam melakukan peramalan permintaan, yaitu: regresi, *support vector machine* (SVM), dan *artificial neural network* (ANN). Penjelasan mengenai SVM dan ANN akan disampaikan pada sub-bab berikutnya.

2.4 Metode Croston dan Modifikasinya

Croston (1972) menjadi pelopor dalam mengembangkan metode peramalan yang akurat dan mampu menangani kemunculan permintaan yang jarang-jarang (*intermittence*) pada suatu *time-series*. Croston mengusulkan prosedur peramalan yang tidak hanya memperbarui nilai permintaan saja, tetapi juga memperbarui nilai interval antara dua permintaan bukan nol yang berurutan secara terpisah. Nilai permintaan dan interval tersebut hanya diperbarui ketika ada permintaan pada periode yang diamati. Akhirnya, peramalan permintaan untuk periode selanjutnya dihitung dengan rasio antara nilai estimasi permintaan hasil SES dan nilai estimasi interval permintaan pada periode selanjutnya. Secara umum, rumus perhitungan metode Croston adalah sebagai berikut.

$$Y_t = \begin{cases} Y_{t-1}, & X_{t-1} = 0 \\ \alpha \cdot X_{t-1} + (1 - \alpha) \cdot Y_{t-1}, & X_{t-1} > 0 \end{cases} \quad (2.19)$$

$$q_t = \begin{cases} q_{t-1} + 1, & X_{t-1} = 0 \\ 1, & X_{t-1} > 0 \end{cases} \quad (2.20)$$

$$p_t = \begin{cases} p_{t-1}, & X_{t-1} = 0 \\ \alpha \cdot q_{t-1} + (1 - \alpha) \cdot p_{t-1}, & X_{t-1} > 0 \end{cases} \quad (2.21)$$

$$F_t = \frac{Y_t}{p_t} \quad (2.22)$$

Keterangan :

- X_t = permintaan aktual pada periode t
- α = parameter *smoothing*, bernilai nol sampai satu
- Y_t = estimasi permintaan pada periode t dari hasil *smoothing*
- q_t = interval aktual antara dua permintaan bukan nol berurutan pada periode t
- p_t = estimasi interval pada periode t
- F_t = peramalan permintaan pada periode selanjutnya

Kemudian Syntetos & Boylan (2005) dalam penelitiannya menemukan bahwa metode Croston ternyata bias terhadap beberapa nilai α . Keduanya menyatakan bahwa dalam beberapa kasus, *expected value* dari rumus Croston tidak setara dengan rasio antara nilai rata-rata aktual permintaan (μ) dan rata-rata interval aktual permintaan (p), dan mengusulkan perbaikan sebagai berikut.

$$\text{Croston} : E\left(\frac{Y_t}{p_t}\right) \approx \frac{\mu}{p} + \frac{\alpha}{2-\alpha} \cdot \mu \cdot \frac{p-1}{p^2} \quad (2.23)$$

$$\text{Syntetos dan Boylan} : E\left((1 - \frac{\alpha}{2}) \frac{Y_t}{p_t}\right) \approx \frac{\mu}{p} \quad (2.24)$$

Oleh karena itu, Syntetos & Boylan (2005) mengusulkan perubahan pada peramalan Croston menjadi:

$$F_t = (1 - \frac{\alpha}{2}) \frac{Y_t}{p_t} \quad (2.25)$$

Usulan tersebut dikenal sebagai *Syntetos Boylan Approximation* (SBA). Hasilnya menunjukkan kesimpulan bahwa, metode Croston akan menghasilkan bias yang

kecil pada riwayat permintaan dengan nilai nol yang sedikit, sedangkan pada riwayat permintaan dengan nilai nol yang lebih banyak, SBA menghasilkan bias yang lebih kecil.

Teunter, Syntetos, dan Babai juga melakukan perubahan pada metode Croston untuk menyesuaikan kebutuhan dalam penelitiannya, yang dikenal dengan model TSB (Teunter, et al., 2011). Ide dari model TSB adalah mengizinkan metode Croston untuk memperbarui estimasi interval secara periodik meskipun tidak ada permintaan selama periode tertentu. Hal tersebut berfungsi untuk menghindari peramalan yang cenderung konstan pada permintaan yang mengandung nol berperiode panjang.

$$\text{Croston \& SBA : } p_t = p_{t-1}, \quad X_{t-1} = 0 \quad (2.26)$$

$$\begin{aligned} \text{TSB : } & p_t = (1 - \beta) \cdot p_{t-1}, \quad X_{t-1} = \\ & 0 \end{aligned} \quad (2.27)$$

Pada model TSB terdapat dua parameter *smoothing*, yaitu α dan β . Parameter β berfungsi untuk memperbarui estimasi interval secara eksponensial. Oleh karena itu, meskipun tidak ada permintaan, model TSB tetap memperbarui estimasi interval dengan menguranginya secara eksponensial. Secara lengkap modifikasi metode Croston pada model TSB adalah sebagai berikut.

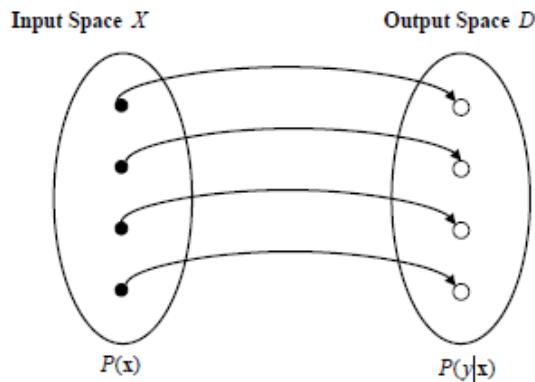
$$p_t = \begin{cases} (1 - \beta) \cdot p_{t-1}, & X_{t-1} = 0 \\ \beta + (1 - \beta) \cdot p_{t-1}, & X_{t-1} > 0 \end{cases} \quad (2.28)$$

$$F_t = Y_t \cdot p_t \quad (2.29)$$

2.5 Support Vector Machine (SVM)

Metode SVM, pada mulanya dikembangkan oleh Vapnik, et al. (1997) untuk menyelesaikan problem-problem klasifikasi. Metode SVM dikembangkan berdasarkan pada prinsip *Statistical Learning Theory* dan *Empirical Risk Minimization* yang juga telah mengilhami banyak metode lainnya, seperti regresi linear dan *neural network* (Vapnik, 2000). Tujuan utama dari kedua prinsip tersebut

adalah untuk mencari fungsi estimasi dari $f:X \rightarrow D$ yang paling tepat dalam memetakan titik-titik dari *input space* X pada *output space* D .



Gambar 2. 7 Ilustrasi Pemetaan *Input Space* Terhadap *Output Space*
(Adhikari & Agrawal, 2013)

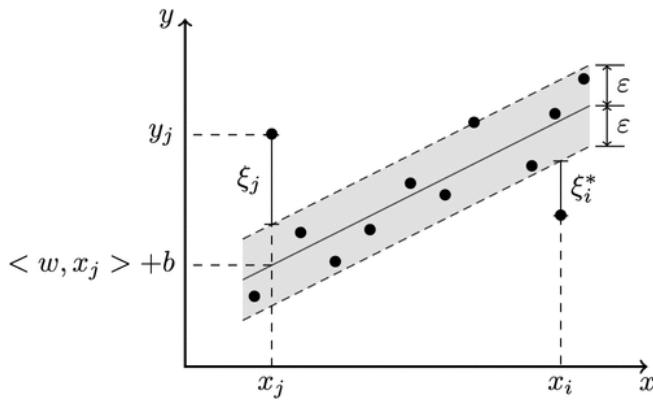
Fungsi yang dipilih adalah fungsi yang mampu meminimalkan nilai *empirical risk* dan didefinisikan sebagai $R_{emp}(f)$. Dalam SVM, $L(y_i, f(x_i, w))$ disebut juga *loss function*.

$$R_{emp}(f) = \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i, w)) \quad (2.30)$$

Selain itu, dalam proses pembelajarannya, metode SVM juga melibatkan penyelesaian problem *linear quadratic programming* dan pemetaan *input* pada ruang dimensi yang lebih tinggi menggunakan fungsi kernel. Oleh karena itu, metode SVM seringkali memberikan solusi yang global, akan tetapi memerlukan waktu komputasi yang lama ketika data yang digunakan berukuran besar.

Banyak penelitian telah mengembangkan kemampuan SVM untuk melakukan tugas prediksi dengan output bilangan yang kontinyu, seperti untuk meramalkan permintaan. Berdasarkan Schölkopf & Smola (2001), SVM bisa digeneralisasi untuk melakukan regresi. Ide dasar dari SVM untuk regresi atau SVR ini adalah untuk menemukan suatu fungsi regresi $f(x)$ yang mempunyai penyimpangan dari target aktual y_i maksimal sebesar ε untuk semua data x_i ,

sehingga fungsi regresi yang ideal tercapai ketika ε bernilai 0. Selain itu, metode SVR juga berdasarkan pada teori *Structural Risk Minimization*, sehingga berlaku suatu *loss function* yang disebut dengan ε -insensitive loss function. Akibatnya setiap titik yang berada di luar *hyperplane*, disebut juga dengan *support vectors*, akan dikenai penalti. Oleh karena itu, pada SVR terjadi *trade-off* untuk meminimalisasi nilai ε atau menghindari penalti.



Gambar 2. 8 ε -insensitive loss function, Semua Titik Di Luar Area Berwarna Dikenai Penalti
 (Santosa & Umam, 2018)

Dimisalkan suatu fungsi $f(x)$ tersebut adalah:

$$f(x) = w^T \phi(x) + b \quad (2.31)$$

Di mana $\phi(x)$ menunjukkan titik-titik pada ruang dimensi lain hasil pemetaan x dari *input space*. Koefisien w dan konstanta b pada fungsi tersebut bisa diestimasi dengan meminimalkan fungsi objektif pada persamaan sebagai berikut.

$$\min \frac{1}{2} \|w\|^2 + C \frac{1}{m} \sum_{i=1}^m L_\varepsilon(y_i, f(x_i)) \quad (2.32)$$

Subject to:

$$y_i - \phi(x_i) - b \leq \varepsilon \quad (2.33)$$

$$\phi(x_i) - y_i + b \leq \varepsilon, i = 1, 2, \dots, m \quad (2.34)$$

$L_\varepsilon(y_i, f(x_i))$ menyatakan fungsi penalti untuk ε -insensitive loss function. Berdasarkan ide dasar SVR sebelumnya, maka penalti tersebut hanya akan diberlakukan jika penyimpangan $f(x_i)$ lebih dari atau sama dengan ε .

$$\begin{aligned} L_\varepsilon(y_i, f(x_i)) \\ = \begin{cases} 0, & |y_i - f(x_i)| < \varepsilon \\ |y_i - f(x_i)| - \varepsilon, & |y_i - f(x_i)| \geq \varepsilon \end{cases} \end{aligned} \quad (2.35)$$

Menurut Santosa & Umam (2018), terdapat dua faktor yang harus diperhatikan pada fungsi objektif di atas, yaitu: *function capacity* dan *empirical error*. Meminimalkan $\|w\|^2$ akan membuat suatu fungsi setipis (*flat*) mungkin, sehingga bisa mengontrol kapasitas fungsi (*function capacity*), akan tetapi memperbesar kemungkinan terjadinya *empirical error* yang diukur dengan ε -insensitive loss function. Menurut Vapnik (2000), untuk memperoleh suatu fungsi regresi yang baik, diperlukan generalisasi pada $\|w\|^2$ sehingga Persamaan 2.10 berubah menjadi:

$$\min \frac{1}{2} \|w\|^2 \quad (2.36)$$

Subject to:

$$y_i - w^T \phi(x_i) - b \leq \varepsilon, i = 1, 2, \dots, m \quad (2.37)$$

$$w^T \phi(x_i) - y_i + b \leq \varepsilon, i = 1, 2, \dots, m \quad (2.38)$$

Namun pada Persamaan 2.12 masih diasumsikan bahwa, fungsi f dapat mengaproksimasi semua titik (x_i, y_i) dengan presisi ε . Oleh karena itu, hanya berlaku pada kasus yang *feasible*, di mana semua pasangan (x_i, y_i) masih di dalam *hyperplane*. Dalam kasus yang *infeasible*, di mana kemungkinan ditemui titik-titik yang keluar dari rentang $f \pm \varepsilon$, maka perlu ditambahkan variabel *slack* ξ, ξ^* untuk meminimalkan nilai penalti C sebagai berikut (Vapnik, 2000):

$$\min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m (\xi_i + \xi_i^*) \quad (2.39)$$

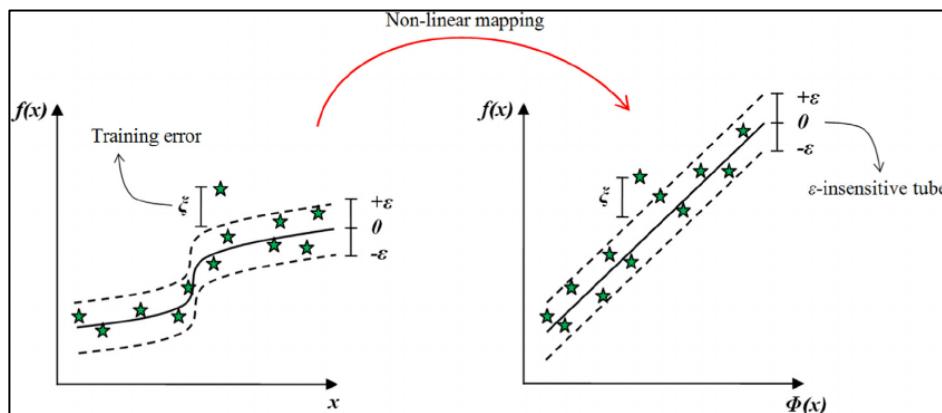
Subject to:

$$y_i - w^T \phi(x_i) - b - \xi_i \leq \varepsilon, i = 1, 2, \dots, m \quad (2.40)$$

$$w^T \phi(x_i) - y_i + b - \xi_i^* \leq \varepsilon, i = 1, 2, \dots, m \quad (2.41)$$

$$\xi_i, \xi_i^* \geq 0 \quad (2.42)$$

Sebagaimana diterangkan sebelumnya bahwa penyelesaian suatu problem dalam SVM adalah ekuivalen dengan penyelesaian suatu problem *linear quadratic programming*. Namun pada beberapa fungsi non-linear, penyelesaian SVR tetap bisa digunakan dengan cara menyelesaikan permasalahan dual dari fungsi tersebut sehingga titik-titik pada *input space* x_i bisa dipetakan menjadi titik-titik $\phi(x_i)$ pada ruang dimensi yang lebih tinggi menggunakan fungsi kernel. Dengan demikian, fungsi tersebut bisa diselesaikan secara linear (Schölkopf & Smola, 2001).



Gambar 2. 9 Ilustrasi Kernelisasi Suatu *Input* Ke Dimensi Yang Lebih Tinggi

(Mahdevari, et al., 2014)

Terdapat beberapa alternatif fungsi kernel yang bisa digunakan, di antaranya (Adhikari & Agrawal, 2013):

- **Kernel Linear**

$$K(x, y) = x^T y \quad (2.43)$$

- **Kernel Polinomial**

$$K(x, y) = (1 + x^T x_i)^p \quad (2.44)$$

- **Gaussian Radial Basis Function (RBF) Kernel**

$$K(x, y) = \exp\left(\frac{-\|x - x_i\|^2}{2\sigma^2}\right) \quad (2.45)$$

atau $K(x, y) = \exp(-\gamma\|x - x_i\|^2)$ (2.46)

di mana $\gamma > 0$

- **Kernel Sigmoid**

$$K(x, y) = \tanh(\alpha x^T x_i + \beta) \quad (2.47)$$

Menurut Vapnik (2000), suatu fungsi kernel yang baik untuk SVR harus memenuhi syarat kontinyu dan *positive definite* berdasarkan Teori Mercer. Kemudian, bentuk dual dari persamaan optimasi pada SVR adalah sebagai berikut.

$$\begin{aligned} \max - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(x_i, x_j) \\ - \sum_{i=1}^m y_i (\alpha_i - \alpha_i^*) - \varepsilon \sum_{i=1}^m (\alpha_i - \alpha_i^*) \end{aligned} \quad (2.48)$$

Subject to:

$$\sum_{i=1}^m (\alpha_i - \alpha_i^*) = 0 \quad (2.49)$$

$$0 \leq \alpha_i \leq C, i = 1, 2, \dots, m \quad (2.50)$$

$$0 \leq \alpha_i^* \leq C, i = 1, 2, \dots, m \quad (2.51)$$

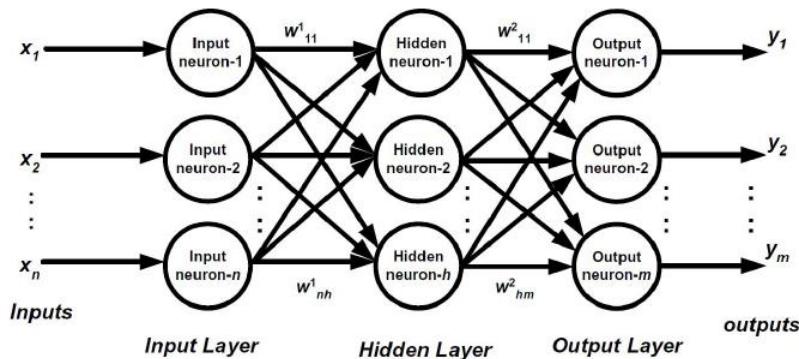
Dengan menggunakan *Lagrange Multiplier*, fungsi regresi secara eksplisit dapat dirumuskan dari kondisi optimalitas bentuk dual persamaan SVR, menjadi:

$$f(x) = \sum_{i=1}^m (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad (2.52)$$

Di mana $K(x_i, x)$, yang didefinisikan sebagai $K(x_i, x) = \langle \phi(x_i), \phi(x) \rangle$, adalah *dot-product* dari fungsi kernel K antara data *training* x_i dan data *testing* x .

2.6 Artificial Neural Network (ANN)

ANN atau metode jaringan tiruan syaraf merupakan salah satu model *non-parametric supervised learning* yang terinspirasi dari studi terhadap sistem saraf dalam otak manusia. Mitchell (1997) menjelaskan bahwa ANN dapat dideskripsikan sebagai model yang tersusun dari bagian-bagian yang disebut *layers* dan unit-unit yang saling terhubung yang disebut neurons. Setiap koneksi yang menghubungkan antara neurons memiliki variabel bobot yang dilambangkan dengan w (**Gambar 2. 10**).

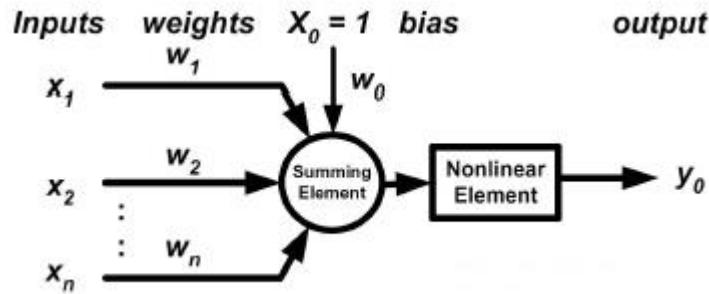


Gambar 2. 10 Arsitektur *Artifical Neural Network*

(Palit & Popovic, 2005)

Mekanisme dasar dari ANN dimulai dengan memproses semua *input* pada *input layer* menuju ke *output layer* dengan melewatkannya pada sejumlah bagian-bagian yang tersembunyi (*hidden layers*). Pada masing-masing bagian (*layers*) tersebut, setiap neurons memproses *input* menjadi *output* menggunakan suatu fungsi aktivasi dan mentransfernya ke neuron di bagian lain secara terus-menerus sampai mencapai *output layer*. Parameter-parameter dalam model ANN, contohnya bobot w , diestimasi melalui proses pembelajaran.

Banyaknya jumlah *layer* pada model ANN sering disebut juga sebagai *depth of the network* dan banyaknya jumlah neurons pada setiap *layer* disebut *width of the network*. Jumlah kedua struktur jaringan tersebut dan bagaimana keduanya berhubungan menimbulkan banyak variasi pada model ANN. Salah satu variasi yang paling dasar dan populer adalah model Perceptron (**Gambar 2. 11**), yang berasal dari paper (Widrow & Hoff, 1960).



Gambar 2. 11 Model *Single Layer* Perceptron

(Widrow & Hoff, 1960)

Nilai bobot (*weights*) dan bias pada model Perceptron merepresentasikan nilai batas yang digunakan untuk menghitung nilai *output* dari fungsi penjumlahan:

$$y_0 = f\left(\sum_{i=1}^n w_i x_i + w_0\right) \quad (2.53)$$

Nilai *output* tersebut kemudian diubah menjadi elemen nonlinear dengan persamaan $f(v = w^T x + w_0)$ yang disebut juga dengan fungsi aktivasi. Hal tersebut berarti, Perceptron akan aktif jika dan hanya jika $v \geq 0$.

$$f(v) = \begin{cases} 0, & v < 0 \\ 1, & v \geq 0 \end{cases} \quad (2.54)$$

Kemampuan Perceptron dalam mengenali nonlinearitas pada data, membuat ANN menjadi populer. Selain itu, ANN juga populer karena sifatnya yang universal dan *data-driven* sehingga bisa dikembangkan secara luas untuk menyelesaikan berbagai problem termasuk peramalan (Shahrabi, et al., 2009, Henkelmann, 2018). Akan tetapi, ANN memerlukan banyak sekali penyesuaian pada fitur-fiturnya agar memberikan hasil yang diinginkan. Penyesuaian tersebut diantaranya adalah: (1) penentuan jumlah *layers / depth*, (2) penentuan jumlah neurons / *width*, (3) pemilihan fungsi transfer, (4) pemilihan fungsi pembelajaran, (5) pemilihan

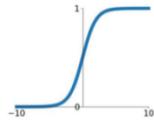
algoritma optimasi, (6) inisialisasi nilai bobot, dan (7) penentuan kriteria pemberhentian (Shahrabi, et al., 2009).

Beberapa fungsi aktivasi yang dapat dipilih yaitu termasuk, tetapi tidak terbatas pada (Nwankpa, et al., 2018): (1) Fungsi Sigmoid, (2) Fungsi Tangent Hiperbolik, (3) *Rectified Linear Units* (ReLUs), (4) Leaky ReLUs, (5) Fungsi Maxout, dan *Exponential Linear Units* (ELUs).

Activation Functions

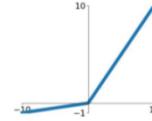
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



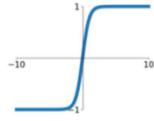
Leaky ReLU

$$\max(0.1x, x)$$



tanh

$$\tanh(x)$$

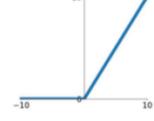


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

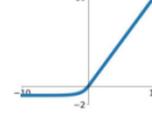
ReLU

$$\max(0, x)$$



ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Gambar 2. 12 Berbagai Alternatif Fungsi Aktivasi ANN

(Udofia, 2018)

Menurut Palit & Popovic (2005), Sigmoid dan ReLU merupakan fungsi aktivasi yang paling banyak digunakan pada model Perceptron. Sedangkan untuk fungsi pembelajaran, algoritma *Backpropagation* adalah yang paling dasar (Rumelhart, et al., 1986). Ide dari algoritma *Backpropagation* muncul disebabkan pada ANN, data *training* sebagai *input* diproses ke arah depan menuju *output layer* dan nilai *error* dihitung dari penyimpangan *output* ANN terhadap nilai yang sebenarnya. Oleh karena itu, untuk memperbaiki kinerja ANN, algoritma *Backpropagation* membantu untuk mengembalikan nilai *error* tersebut sampai ke *input layer* sehingga bisa menjadi referensi dalam menentukan nilai bobot *w* dan bias yang lebih baik.

Berbagai macam algoritma optimasi bisa dipilih untuk memperbarui nilai bobot dalam proses *Backpropagation*. Salah satu pendekatannya yang paling dasar dalam ANN adalah *Stochastic Gradient Descent* (SGD). SGD menerapkan

algoritma *gradient descent* secara iteratif, di mana pada setiap iterasi secara acak suatu sampel z_t dipilih dan dihitung *error*-nya terhadap w . Selanjutnya, nilai gradien dihitung berdasarkan turunan fungsi *error* z_t atau dilambangkan dengan $Q(z, w)$ sebagai berikut.

$$g_t = \nabla_w Q(z_t, w_t) \quad (2.55)$$

Pada setiap iterasi, nilai w diperbarui berdasarkan nilai gradien tersebut dan suatu parameter tingkat pembelajaran η yang mempengaruhi kecepatan konvergensi pada ANN.

$$w_{t+1} = w_t - \eta g_t \quad (2.56)$$

Nilai η harus ditentukan secara hati-hati, karena tingkat pembelajaran yang terlalu besar atau terlalu kecil menyebabkan nilai optimal sulit tercapai (Henkelmann, 2018). SGD populer karena kecepatannya dalam mencapai konvergen meskipun dengan biaya komputasi yang rendah.

Selain penyesuaian-penesuaian di atas, menurut Schmidhuber (2014), penentuan nilai bobot di awal pembelajaran juga sangat menentukan kemampuan ANN dalam mencapai optimal. Glorot & Bengio (2010) mengusulkan pendekatan inisialisasi bobot menggunakan distribusi uniform, di mana w_{in} dan w_{out} merupakan dimensi dari bobot pada *input* dan *output*.

$$U\left(-\sqrt{\frac{6}{w_{in} - w_{out}}}, \sqrt{\frac{6}{w_{in} - w_{out}}}\right) \quad (2.57)$$

Setelah semua penyesuaian dilakukan, maka proses pembelajaran pada ANN dapat dijalankan. Menurut Şahin, et al. (2013), proses pembelajaran pada ANN dapat dirangkum menjadi: *training stage*, *validation stage*, dan *testing stage*. Pada tahap *training*, ANN mempelajari hubungan antara *input* dan *output* dalam jaringan dan berusaha mengurangi *error* pada setiap iterasi. Tahap *training* berhenti ketika

kriteria pemberhentian terpenuhi, yang bisa jadi adalah jumlah maksimum iterasi (*epoch*) atau batas bawah nilai *error* yang sudah tercapai. Kemudian pada tahap *cross-validation*, nilai-nilai bobot terbaik disimpan untuk menguji kinerja model ANN terhadap data baru pada tahap *testing*.

2.7 Mengukur Kinerja Peramalan (*Forecast Error* dan *Robustness*)

Suatu problem peramalan dapat diselesaikan dengan berbagai alternatif pendekatan. Namun demikian, setiap pendekatan memiliki asumsi dan penyesuaian yang berbeda-beda dalam perancangan modelnya sehingga berdampak pada kinerja peramalan yang berbeda-beda. Kinerja dari suatu model peramalan biasanya diukur dengan suatu parameter yang disebut *performance metrics* (Cao & Tay, 2003). Menurut Montgomery, et al. (2008), *performance metrics* yang paling baik untuk mengukur kinerja peramalan adalah *forecasting error*. *Error* dalam peramalan didefinisikan sebagai nilai penyimpangan dari nilai estimasi \hat{y} terhadap nilai aktual peramalan y .

$$e_t(1) = y_t - \hat{y}_t(t-1) \quad (2.58)$$

Banyak pendekatan telah dikembangkan untuk mengukur *forecasting error*. Sebagian besar dari pendekatan tersebut dijelaskan dalam Adhikari & Agrawal (2013) dan dapat dirangkum pada **Tabel 2. 2** sebagai berikut.

Tabel 2. 2 Berbagai Alternatif Pengukuran Akurasi Peramalan

<i>Forecasting Error</i>	<i>Properties</i>					<i>Formula</i>
	I	II	III	IV	V	
1. Mean Forecast Error (MFE)	✓	✓		✓	✓	$MFE = \frac{1}{n} \sum_{t=1}^n e_t$

Tabel 2. 2 Berbagai Alternatif Pengukuran Akurasi Peramalan (Lanjutan)

Forecasting Error	Properties					Formula
	I	II	III	IV	V	
2. Mean Absolute Error (MAE)				✓	✓	$MAE = \frac{1}{n} \sum_{t=1}^n e_t $
3. Mean Percentage Error (MPE)	✓	✓			✓	$MPE = \frac{1}{n} \sum_{t=1}^n \left(\frac{e_t}{y_t} \right) \times 100$
4. Mean Absolute Percentage Error (MAPE)					✓	$MAPE = \frac{1}{n} \sum_{t=1}^n \left \frac{e_t}{y_t} \right \times 100$
5. Sum Squared Error (SSE)			✓	✓	✓	$SSE = \sum_{t=1}^n e_t^2$
6. Mean Squared Error (MSE)			✓	✓	✓	$MSE = \frac{1}{n} \sum_{t=1}^n e_t^2$
7. Signed Mean Squared Error (SMSE)	✓		✓	✓	✓	$SMSE = \frac{1}{n} \sum_{t=1}^n \left(\frac{e_t}{ e_t } \right) e_t^2$
8. Normalized Mean Squared Error (NMSE)			✓	✓	✓	$NMSE = \frac{MSE}{\sigma^2}$

Properties pada tabel di atas adalah: I) Menunjukkan arah positif atau negatif dari error, II) Error positif dan negatif saling menghilangkan, III) Menggambarkan nilai error yang ekstrem, IV) Dependen terhadap skala pengukuran, dan V) Dependen terhadap transformasi data.

Parameter-parameter *forecasting error* di atas juga digunakan untuk menilai akurasi model peramalan. Menurut Montgomery, et al. (2008), pendekatan terbaik untuk memilih model peramalan yang saling dibandingkan adalah dengan memilih model yang menghasilkan *mean squared error* (MSE) paling kecil. Namun perlu diperhatikan bahwa, memilih model hanya berdasarkan akurasinya terhadap data *training* seringkali menyebabkan *overfitting*. Hal tersebut ditunjukkan dengan penurunan pada akurasi model setelah diuji dengan data yang baru (*data testing*). Oleh karena itu, Montgomery, et al. (2008) menyarankan bahwa pengukuran akurasi pada model peramalan, sebaiknya dilakukan terhadap dua segmen data

yang berbeda, yaitu: data *training* dan *data testing*. Proses tersebut sering disebut juga dengan validasi peramalan yang akan dijelaskan pada sub-bab berikutnya.

Selain akurasi model peramalan, parameter lain yang perlu dipertimbangkan untuk meminimalisasi terjadinya *overfitting* adalah *forecasting robustness*. Menurut Xu & Mannor (2010) dan Chen (1997), *forecasting robustness* berhubungan dengan konsistensi model peramalan terhadap pengaruh proses *data generation*. Dalam kaitannya dengan peramalan menggunakan *data mining*, maka *forecasting robustness* seringkali didefinisikan sebagai *algorithmic stability*, yaitu stabilitas hasil prediksi dari suatu algoritma peramalan terhadap perubahan kecil pada *input* algoritma. Xu & Mannor (2010) menyatakan bahwa *robustness* bisa diidentifikasi dari hasil prediksi yang tidak jauh berbeda pada *data training* dan *data testing* sehingga mempertimbangkan *robustness* bisa meminimalisasi *overfitting*.

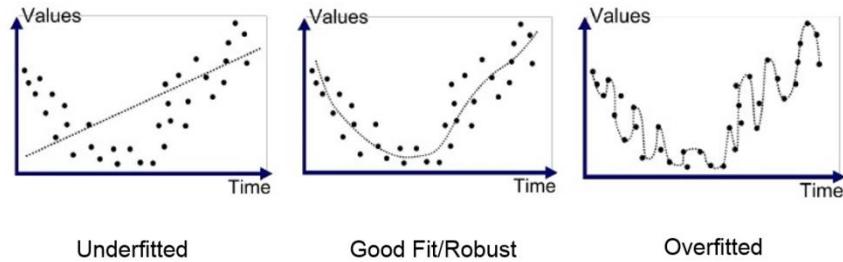
2.8 Validasi Peramalan

Menurut Montgomery, et al. (2008), proses validasi berfungsi sebagai petunjuk ketika memilih model peramalan yang saling dibandingkan. Pada tahap ini, setiap model akan dievaluasi kinerjanya dalam menangani sejumlah data asing, atau dengan kata lain adalah data yang tidak digunakan untuk membuat model peramalan. Pada evaluasi tersebut, setiap model akan dilihat apakah kinerjanya mengalami penurunan dibandingkan dengan sebelumnya, yaitu pada tahap pembelajaran. Model yang kinerja menurun dikatakan mengalami *overfitting*. Menurut Santosa & Umam (2018), *overfitting* terjadi karena nilai *error* dari suatu model prediksi memiliki unsur bias dan varian.

$$MSE = \text{varian} + \text{bias}^2 \quad (2.59)$$

Bias mewakili seberapa besar kesalahan prediksi, sedangkan varian mewakili variasi dari nilai output prediksi dan berhubungan dengan kompleksitas model. Seringkali *overfitting* terjadi ketika model prediksi dipaksa untuk memiliki bias yang kecil (kesalahan prediksi yang kecil) pada fase pembelajaran, akibatnya model yang dihasilkan memiliki varian yang tinggi (kompleksitas tinggi karena

duplikasi data minor yang berlebihan) sehingga memicu terjadinya *overfitting* (**Gambar 2. 13**).



Gambar 2. 13 Ilustrasi *Overfitting*

(Bhande, 2018)

Oleh karena itu, dapat disimpulkan bahwa bias (mewakili *error*) dan varian (mewakili *overfitting*) adalah suatu *trade-off* antara memilih model yang memiliki *error* yang kecil (tetapi rawan *overfitting*) atau model yang lebih general (tidak rawan *overfitting*) tetapi bisa jadi memiliki nilai *error* yang lebih besar. Model terbaik adalah model yang mempertimbangkan *trade-off* tersebut (Santosa & Umam, 2018).

2.9 Posisi Penelitian

Penelitian mengenai peramalan permintaan *spare part* dengan berbagai macam metode, baik *time-series* maupun *data mining*, sudah banyak dilakukan. Akan tetapi, hanya sebagian kecil di antaranya yang secara langsung membandingkan kinerja antara metode Croston, modifikasi Croston, SVM, dan ANN, sebagaimana tugas akhir ini. Ringkasan mengenai penelitian-penelitian tersebut beserta posisi tugas akhir ini terhadap penelitian-penelitian tersebut akan dijelaskan dalam **Tabel 2. 3** sebagai berikut.

Tabel 2. 3 Penelitian-Penelitian Serupa Mengenai Peramalan Permintaan *Spare Part*

NO.	JUDUL	TAHUN	AUTHOR	INPUT	METODE PERAMALAN				KESIMPULAN
					CR	MOD-CR	SVM	ANN	
1.	<i>Lumpy Demand Forecasting using Neural Networks</i>	2008	Gutierrez, et al.	Sejumlah 24 stock keeping units (SKU) masing-masing mengandung 967 permintaan harian yang bersifat <i>lumpy</i> .	✓	✓		✓	ANN dengan menggunakan <i>Multi Layer Perceptron</i> (MLP) mengungguli metode Croston, SBA, dan <i>Exponential Smoothing</i> , ketika diukur menggunakan MAPE.
2.	<i>Supply Chain Demand Forecasting: A Comparison of Machine Learning Techniques & Traditional Methods</i>	2009	Shahrabi, et al.	Data permintaan <i>supply chain</i> jangka panjang yang dimiliki oleh <i>supplier</i> komponen perusahaan mobil terbesar di Iran, namun tidak dijelaskan secara rinci.			✓	✓	ANN dengan <i>three-layer feedforward neural network</i> mengungguli metode SVM, MA, dan <i>Exponential Smoothing</i> , saat diukur dengan MAPE, baik terhadap data <i>training</i> maupun <i>testing</i> .

Tabel 2. 3 Penelitian-Penelitian Serupa Mengenai Peramalan Permintaan *Spare Part* (Lanjutan)

NO.	JUDUL	TAHUN	AUTHOR	INPUT	METODE PERAMALAN				KESIMPULAN
					CR	MOD-CR	SVM	ANN	
3.	<i>Forecasting Aviation Spare Part Demand Using Croston Based Methods & Artificial Neural Networks</i>	2013	Şahin, et al.	Data permintaan spare part dari perusahaan <i>Turkish Aircraft Maintenance Repair & Overhaul (MRO)</i> yang mayoritasnya bersifat sporadis. Pendekatan klasifikasi Boylan, et al (2005) digunakan untuk membagi data ke dalam beberapa jenis permintaan.	✓	✓		✓	ANN dengan menggunakan <i>Multi Layer Perceptron (MLP)</i> mengungguli metode Croston & SBA pada semua jenis data <i>intermittent, erratic, dan lumpy</i> , ketika diukur dengan GMAMAD atau <i>Geometric Mean of Mean Absolute Deviation</i> .

Tabel 2. 3 Penelitian-Penelitian Serupa Mengenai Peramalan Permintaan *Spare Part* (Lanjutan)

NO.	JUDUL	TAHUN	AUTHOR	INPUT	METODE PERAMALAN				KESIMPULAN
					CR	MOD-CR	SVM	ANN	
4.	<i>Demand Forecasting for Irregular Demands in Business Aircraft Spare Part Supply Chains by using Artificial Intelligence (AI)</i>	2017	Amirkolaii, et al.	Sejumlah permintaan 23.646 SKU dari perusahaan Dassault Aviation dibagi menggunakan pendekatan Boylan, et al (2005) sehingga diperoleh data sebesar 65% <i>intermittent</i> , 31% <i>lumpy</i> , 2% <i>erratic</i> , dan 2% <i>smooth</i> .	✓			✓	MLP pada sebagian besar skenario mengungguli metode Croston dan MA, ketika diukur menggunakan MSE.
5.	<i>Intermittent Demand Forecasting using Data Mining Techniques</i>	2018	Kaya & Turkyilmaz	Data buatan sebanyak 400 permintaan <i>intermittent</i> yang masing-masing memiliki 300 <i>datapoints</i> digunakan untuk peramalan.	✓		✓	✓	SVM mengungguli metode Croston, ANN, dan klasifikasi C5, ketika diukur dengan RMSE, MAD, dan MAPE, baik terhadap data <i>training</i> maupun data <i>testing</i> .

Tabel 2. 3 Penelitian-Penelitian Serupa Mengenai Peramalan Permintaan *Spare Part* (Lanjutan)

NO.	JUDUL	TAHUN	AUTHOR	INPUT	METODE PERAMALAN				KESIMPULAN
					CR	MOD-CR	SVM	ANN	
6.	<i>Forecasting Spare Part Demand on Automotice Industry using Artificial Neural Network (ANN)</i>	2018	Hakim & Dwantara	Data riwayat permintaan bulanan tujuh macam <i>spare part</i> mobil selama 48 periode digunakan untuk meramalkan permintaan 12 periode ke depan.	✓			✓	ANN mengungguli metode Croston dan SES, baik ketika diukur dengan menggunakan MSE maupun MAPE.
7.	Tugas Akhir ini: Analisis Perbandingan Kinerja Peramalan dengan Metode Croston, SBA, SVM, dan ANN, terhadap Klasifikasi Permintaan <i>Spare Part</i> di PT. X (Studi Kasus di Industri Otomotif)	2020	Ramadhan, Defa Ihsan	Riwayat permintaan <i>spare part</i> bulanan di PT. X selama Januari 2018 – Desember 2019 dikelompokkan berdasarkan pola permintaannya dengan pendekatan Boylan, et al (2005) dan teknik klaster, lalu digunakan untuk meramalkan permintaan 3 periode ke depan.	✓	✓	✓	✓	

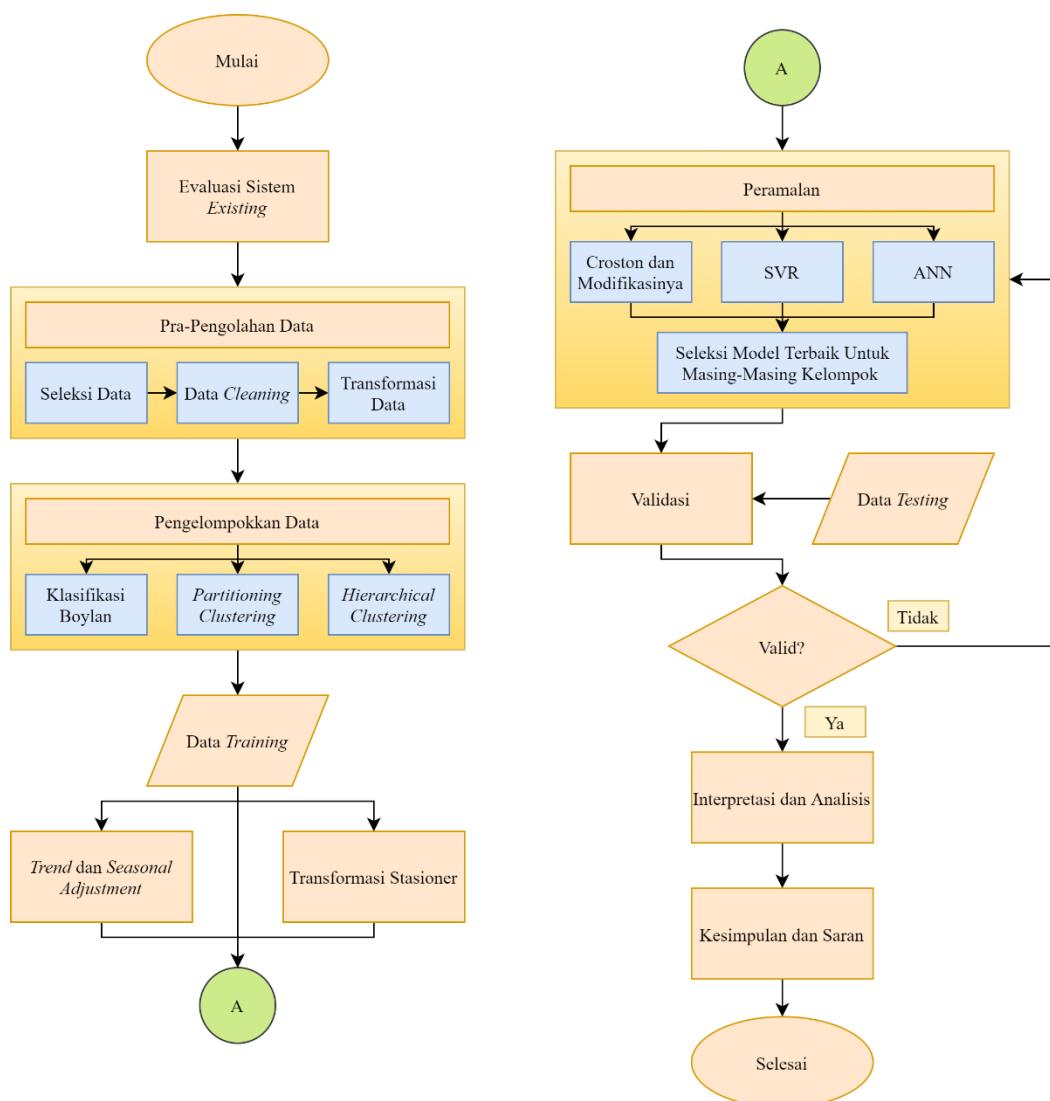
BAB 3

METODOLOGI PENELITIAN

BAB 3

METODOLOGI PENELITIAN

Pada bab ini akan dijelaskan mengenai metodologi yang dilakukan untuk menyelesaikan tugas akhir ini. Proses-proses tersebut terdiri dari tujuh tahap, yaitu: evaluasi sistem *existing*, pra-pengolahan data, pengolahan data yang terbagi menjadi pengelompokkan data dan peramalan, validasi, interpretasi dan analisis, serta penarikan kesimpulan dan saran, sebagaimana digambarkan pada **Gambar 3. 1** berikut.



Gambar 3. 1 *Flowchart* Metodologi Penelitian

3.1 Evaluasi Sistem Existing

Tahap ini dimulai dengan pengumpulan data berupa riwayat *order spare part export* yang terdiri dari beberapa fitur, yaitu:

1. *Periode Order*, berisi tanggal dan bulan ketika *order* terhadap suatu *part* masuk ke sistem.
2. *Tipe Order*, berisi kode yang membedakan tipe permintaan yang masuk, apakah spesial, reguler, atau VOR.
3. *Part No*, berisi nomor unik untuk setiap *part* yang terdiri dari 10 digit kombinasi angka dan huruf, di mana 5 digit pertama melambangkan *family part* dan 5 digit kedua melambangkan model mobil dan spesifikasi *part*.
4. *Part Name*, berisi keterangan nama *part* dari setiap nomor *part*.
5. *Supplier*, berisi keterangan kode dan nama pemasok *part*.
6. *Destination*, berisi keterangan dari negara mana *order* berasal.
7. *Dimensi*, berisi dimensi *part*, yang terdiri dari: *big, medium, dan small*.
8. *Car Model*, berisi keterangan dari model mobil apa suatu *part* berasal.
9. *Family*, berisi keterangan yang membedakan antara *part* penyusun *exterior, interior, engine, dan suspension* pada mobil.
10. *Order Qty*, berisi jumlah *order* suatu *part* pada suatu periode.

Selanjutnya, hasil peramalan *existing* menggunakan metode *simple moving average*, berdasarkan rata-rata *order* tiga dan enam bulan ke belakang, dihitung penyimpangannya terhadap permintaan aktual reguler dan VOR setiap bulannya dengan parameter MFE, MSE, MAE, dan MAPE untuk masing-masing *part*.

Interpretation of typical MAPE values	
MAPE	Interpretation
<10	Highly accurate forecasting
10-20	Good forecasting
20-50	Reasonable forecasting
>50	Inaccurate forecasting

Source: Lewis (1982, p. 40)

Gambar 3. 2 Interpretasi Nilai MAPE

(Lewis, 1982)

Dalam rangka pendalaman masalah, kemudian dilakukan studi literatur dan wawancara terhadap perwakilan staf, kepala departemen, dan *manager* dari divisi *After Sales Service & Production* PT. X. Akhirnya, diperoleh objektif berupa perbaikan (*improvement*) terhadap kinerja peramalan *spare part* yang kurang baik, yaitu yang memiliki MAPE di atas 20% (**Gambar 3.2**), sesuai dengan kriteria *good forecast* menurut (Lewis, 1982).

Kemudian, segmen permintaan *spare part* untuk ekspor ke Arab Saudi disepakati untuk dipilih menjadi representatif data dengan tujuan agar analisis dalam tugas akhir ini menjadi lebih spesifik karena karakteristik permintaan yang lebih homogen. Selain itu, *part* berdimensi *big* dan *medium* merupakan perhatian utama dalam tugas akhir ini, karena dampaknya yang lebih signifikan terhadap biaya penanganan dan penyimpanan *spare part* ketika terjadi *overforecast* atau *underforecast*.

3.2 Pra-Pengolahan Data

Pada tahap ini, data yang telah dikumpulkan kemudian diseleksi berdasarkan objektif dan segmentasi permasalahan. Kemudian dilakukan pembersihan data, di mana pada proses ini, riwayat permintaan *spare part*, yang hanya terjadi di bawah tiga bulan, dibuang dan tidak digunakan dalam peramalan untuk menghindari bias. Pada proses ini, nilai permintaan *spare part* yang *outlier* tidak dibuang karena akan menghilangkan karakteristik dari permintaan *spare part* itu sendiri.

Selanjutnya, set data yang akan digunakan sebagai *input* untuk beberapa metode seperti teknik klaster, SVR, dan ANN, perlu mengalami beberapa penyesuaian, seperti *scaling* dan konversi data ke dalam bentuk *predictor* dan *target*, agar kinerja metodenya lebih baik sebagaimana menurut (Santosa & Umam, 2018). *Scaling* berfungsi untuk menstandarkan seluruh data agar berada dalam rentang skala yang sama, seperti [0,1] atau [-1,1]. Contoh konversi suatu nilai X ke dalam skala [-1,1] dapat dilakukan dengan rumus:

$$\hat{X} = \frac{X - X_{min}}{X_{max} - X_{min}} * (1 - (-1)) + (-1) \quad (3.1)$$

Lalu untuk konversi variabel tunggal (*univariate*) menjadi bentuk *predictor* dan *target*, dapat dilakukan dengan *windowing*. Contoh *windowing* pada data *time-series*, yang dilambangkan dengan s , dengan *time window* $k = 3$ adalah:

dimisalkan $1,2,3,4,5,6,7,8,9 = s[1] \dots s[9]$, maka

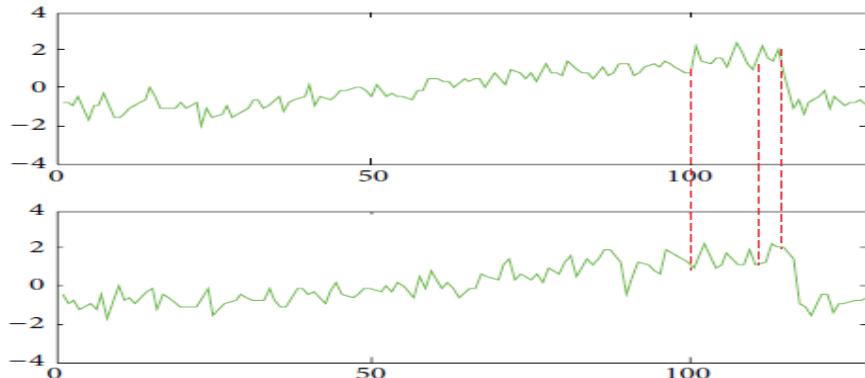
Predictor \rightarrow *Target*

$$[s[k-3], s[k-2], s[k-1]] \rightarrow s[k]$$

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \\ \vdots & \vdots & \vdots \\ 6 & 7 & 8 \end{bmatrix} \rightarrow \begin{bmatrix} 4 \\ 5 \\ 6 \\ \vdots \\ 9 \end{bmatrix}$$

3.3 Pengelompokkan Data

Pada tahap ini, sebelum diramalkan, set data riwayat permintaan *spare part* akan dikelompokkan terlebih dahulu berdasarkan klasifikasi *smooth*, *intermittent*, *erratic*, dan *lumpy* menggunakan pendekatan Boylan, et al. (2005) dan berdasarkan kemiripan polanya pada setiap titik waktu (*similarity in time points*) menggunakan teknik klaster (**Gambar 3. 3**).



Gambar 3. 3 Ilustrasi *similarity in time points* pada data *time-series*

(Aghabozorgi, et al., 2014)

Selain itu, *part* juga dipisah berdasarkan posisinya pada mobil, apakah termasuk: *exterior*, *interior*, *engine*, atau *suspension*. Pemisahan tersebut perlu dilakukan karena masing-masing posisi pada mobil memiliki sumber risiko kerusakan yang

berbeda-beda. Berikut ini adalah gambaran pengelompokkan *spare part* pada tugas akhir ini.

Tabel 3. 1 Bentuk Pengelompokkan *Spare Part* Pada Tugas Akhir Ini

Posisi Part	Kelompok 1	Kelompok ...	Kelompok K
<i>Exterior</i>	<i>Part A,B,C,D</i>
<i>Interior</i>	<i>Part E,F,G,H</i>
<i>Engine</i>	<i>Part I,J,K,L</i>
<i>Suspension</i>	<i>Part M,N,O,P</i>

Teknik pengelompokan yang digunakan adalah pendekatan Boylan, salah satu metode *Partitional Clustering*, yaitu K-Medoids, dan *Agglomerative Hierarchical Clustering*. Ketiga teknik tersebut akan dibandingkan pengaruhnya terhadap kinerja peramalan di tahap selanjutnya. Algoritma dari masing-masing teknik tersebut adalah sebagai berikut.

Algoritma 3. 1 Pendekatan Boylan

- 1: Untuk setiap *spare part* {1 ... N} lakukan :
 - 2: Hitung nilai *inter-demand interval* (p)
 - 3: Hitung jumlah periode (n) *non-zero demand*
 - 4: Hitung $ADI := \frac{p}{n}$
 - 5: Hitung standar deviasi *demand* (σ) dari seluruh periode
 - 6: Hitung rata-rata *demand* ($\bar{\epsilon}$) dari seluruh periode
 - 7: Hitung $CV^2 := \left(\frac{\sigma}{\bar{\epsilon}}\right)^2$
 - 8: Jika $ADI < 1.32$ dan $CV^2 < 0.49$ maka
 - 9: *Kelas := Smooth*
 - 10: Jika $ADI > 1.32$ dan $CV^2 < 0.49$ maka
 - 11: *Kelas := Intermittent*
 - 12: Jika $ADI < 1.32$ dan $CV^2 > 0.49$ maka
 - 13: *Kelas := Erratic*
 - 14: Jika $ADI > 1.32$ dan $CV^2 > 0.49$ maka
 - 15: *Kelas := Lumpy*
-

Algoritma 3. 2 K-Medoids Clustering / Partitioning Around Medoids (PAM)

- 1: Inisialisasi jumlah klaster K
 - 2: Pilih sebanyak K *data points* secara random sebagai *centroids*
 - 3: Hitung jarak dari setiap *data points* ke setiap *centroids*
 - 4: Alokasikan *data points* dengan *centroids* terdekat menjadi satu klaster
 - 5: Hitung *Sum of Squared Error* (SSE)
 - 6: Selama alokasi klaster masih berubah-ubah lakukan :
 - 7: Untuk setiap *centroids* (c) {1 ... K} lakukan :
 - 8: Untuk setiap *data points* non-*centroids* (o) {1 ... N – K} lakukan :
 - 9: Tukar c dengan o
 - 10: Alokasikan ulang *data points* dan hitung SSE
 - 11: Jika penukaran memperbaiki SSE maka
 - 12: *centroids* := *medoids* := o
 - 13: Ketika alokasi klaster tidak berubah, **stop**
 - 14: Simpan klaster, *centroids*, dan SSE
 - 15: Ulangi dengan jumlah K yang berbeda
 - 16: Evaluasi hasil klaster pada nilai K yang berbeda-beda
 - 17: Tentukan jumlah K yang terbaik
-

Algoritma 3. 3 Agglomerative Hierarchical Clustering

- 1: Untuk setiap *spare part* {1 ... N} lakukan :
 - 2: Hitung matriks jarak antara *data points*
 - 3: Inisialisasi semua N *data points* menjadi N klaster pada dendogram
 - 4: Selama masih ada lebih dari satu klaster lakukan :
 - 5: Hitung jarak *linkage* antara klaster
 - 6: Gabungkan dua *cluster* terdekat menjadi satu klaster
 - 7: Perbarui matriks jarak setelah penggabungan
 - 8: Ketika seluruh *data points* menjadi satu klaster, **stop**
 - 9: Simpan dendogram
 - 10: Ulangi dengan ukuran jarak dan *linkage* yang berbeda-beda
 - 11: Hitung *cophenet*
-

12: Evaluasi nilai *cophenet* masing-masing ukuran *linkage*

13: Tentukan ukuran *linkage* terbaik

Software RStudio dengan bahasa *R* 4.0.0 akan digunakan untuk menjalankan metode *K-Medoids Clustering* dan *Agglomerative Hierarchical Clustering*. Pada kedua metode tersebut juga dimplementasikan beberapa metode untuk menentukan jumlah klaster yang optimal, yaitu: Pendekatan Calinski-Harabasz, Metode *Elbow*, dan Metode *Silhouette* (**Tabel 3. 2**).

Tabel 3. 2 Metode Untuk Menentukan Jumlah Klaster

No.	Metode	Parameter	Keterangan
1.	Calinski-Harabasz	$CH(k) = \frac{N - k \cdot SSB}{k - 1 \cdot SSW}$ $SSB(k) = \sum_{l=1}^k \sum_{j=1}^n C_l (\bar{c}_{lj} - \bar{x}_j)^2$ $SSW(k) = \sum_{l=1}^k \sum_{i \in C_l} \sum_{j=1}^n (x_{ij} - \bar{c}_{lj})^2$	Jumlah klaster yang optimal adalah yang memberikan nilai <i>CH</i> paling tinggi
2.	<i>Elbow</i>	$W_k = \sum_{l=1}^k \frac{1}{2 C_l } \sum_{q,r \in C_l} d(q,r)$	Jumlah klaster yang optimal adalah yang berada pada posisi <i>Elbow</i> , yaitu penurunan nilai <i>W_k</i> yang signifikan
3.	<i>Silhouette</i>	$SWC = \frac{1}{N} \sum_{j=1}^N s(j)$ $s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$	Jumlah klaster yang optimal adalah yang memberikan nilai <i>SWC</i> paling tinggi

Calinski-Harabasz *Index* (*CH*) didefinisikan sebagai rasio antara jumlah jarak kuadrat antara klaster (*sum of squared between cluster*) dan jumlah jarak kuadrat antara objek dalam satu klaster (*sum of squared within cluster*) pada seluruh klaster yang berjumlah *k*. Metode *Elbow* didefinisikan dengan parameter *Within-Cluster Dispersion* (*W_k*), yaitu jumlah rata-rata jarak antara objek dalam satu klaster pada seluruh klaster. Terakhir, Metode *Silhouette* didefinisikan dengan parameter *Silhouette Width Criterion* (*SWC*), yaitu rata-rata nilai *Silhouette* (*s(i)*) dari setiap

objek i . Nilai *Silhouette* bisa dicari dengan menghitung rata-rata jarak objek i ke sesama objek dalam satu klaster ($a(i)$) dan rata-rata jarak objek i ke objek di klaster yang lain ($b(i)$).

3.4 Peramalan

Pada tahap ini dibagi dua tugas, yaitu pengembangan model dan peramalan. Metode peramalan yang dibandingkan dalam tugas akhir ini adalah metode: Croston, SBA, TSB, SVR, dan ANN. Masing-masing metode tersebut memiliki *hyperparameter*, yaitu parameter model yang tidak bisa diestimasi langsung dari data *training*, tetapi harus disesuaikan sendiri oleh pemodel. Penjelasan mengenai *hyperparameter* dari berbagai metode tersebut dan pengaruhnya terhadap kinerja peramalan dapat dilihat pada **Tabel 3. 3**.

Tabel 3. 3 *Hyperparameter* Pada Masing-Masing Metode Peramalan

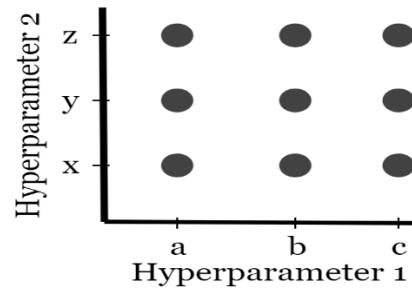
Performance Metrics	Croston, SBA, TSB		SVR			ANN			
	Large α	Large β	Large ϵ	Large C	Large γ	Large Neurons	Large η	Large Epoch	Large Threshold
Bias	Low	Low	High	Low	High	Low	High	Low	High
Varian	High	High	Low	High	Low	High	Low	High	Low

Sumber: (Patel, 2017, Radhakrishnan, 2017)

Pada tahap pengembangan model, dari masing-masing metode akan dikembangkan banyak model dengan berbagai kombinasi *hyperparameter* menggunakan metode *grid search*. Pada metode *grid search* (**Gambar 3. 4**), model akan dikembangkan sebanyak jumlah kombinasi nilai *hyperparameter* yang akan dibandingkan kinerjanya. Selain *hyperparameter*, pada tahap ini juga dilakukan beberapa eksperimen lain, seperti penggunaan data *training* setelah *trend* dan *seasonal adjustment* dan penggunaan data *training* setelah transformasi stasioner. Dari berbagai alternatif tersebut, pada akhirnya akan dipilih skenario terbaik untuk digunakan pada tahap peramalan.

Grid Search

```
Pseudocode
Hyperparameter_One = [a, b, c]
Hyperparameter_Two = [x, y, z]
```



Gambar 3. 4 Ilustrasi *Grid Search*
(Stalfort, 2019)

Algoritma dari metode Croston, SBA, TSB, SVR, dan ANN adalah sebagai berikut.

Algoritma 3. 4 Metode Croston, SBA, dan TSB

-
- 1: Inisialisasi nilai parameter *smoothing* $0 \leq \alpha \leq 1$ dan $0 \leq \beta \leq 1$
 - 2: Pada $t = 0$ lakukan :
 - 3: Inisialisasi nilai estimasi permintaan ($Y_{t=0}$) dari nilai permintaan aktual (X)
 - 4: Inisialisasi nilai estimasi interval ($p_{t=0}$) dari nilai interval aktual (q)
 - 5: Pada t selanjutnya lakukan :
 - 6: Hitung nilai estimasi permintaan (Y_t) dengan rumus Croston
 - 7: Hitung nilai estimasi interval (p_t) dengan rumus Croston dan TSB
 - 8: Hitung peramalan dengan rumus :
 - 9: $Croston := F_t := \frac{Y_t}{p_t}$ atau
 - 10: $SBA := F_t := (1 - \frac{\alpha}{2}) \frac{Y_t}{p_t}$ atau
 - 11: $TSB := F_t := Y_t \cdot p_t$
 - 12: Hitung penyimpangan peramalan dengan MAPE dan MSE
 - 13: Optimasi parameter α dan β sehingga nilai penyimpangan minimal
 - 14: Gunakan model terbaik untuk peramalan selanjutnya
-

Algoritma 3. 5 Metode Support Vector Regression (SVR)

- 1: Definisikan *predictor* dan *target* dari data *training*
 - 2: Inisialisasi nilai (ε), nilai penalti (C), dan nilai gamma (γ)
 - 3: Tentukan fungsi kernel
 - 4: Hitung matriks kernel (K)
 - 5: Tentukan fungsi tujuan problem kuadratik dari persamaan SVR
 - 6: Tentukan pembatas problem kuadratik dari persamaan SVR
 - 7: Selesaikan problem kuadratik SVR
 - 8: Simpan solusi nilai bias (b) dan *langrange multiplier* (a_i)
 - 9: Hitung peramalan menggunakan a_i dan b dengan persamaan regresi :
 - 10: $f(x) := \sum_{i=1}^m (\alpha_i - \alpha_i^*) K(x_i, x) + b$
 - 11: Hitung akurasi peramalan terhadap *target*
 - 12: Lakukan analisis *grid search*
 - 13: Simpan kombinasi *hyperparameter* dengan akurasi terbaik
 - 14: Gunakan model terbaik untuk peramalan selanjutnya
-

Algoritma 3. 6 Metode Artificial Neural Network (ANN): Multi-Layer Perceptron

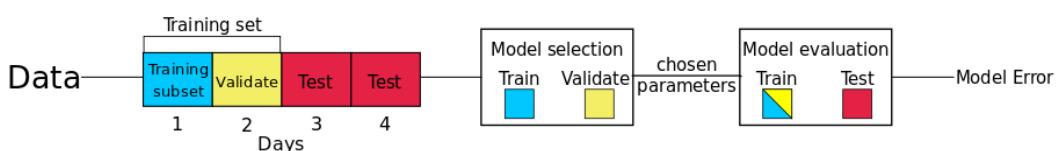
- 1: Definisikan *predictor* dan *target* dari data *training*
 - 2: Inisialisasi jumlah *neurons* pada model Perceptron
 - 3: Inisialisasi nilai *learning rate* (η)
 - 4: Inisialisasi nilai parameter pembelajaran bobot (w) dan bias (b)
 - 5: Tetapkan kriteria pemberhentian jumlah iterasi (*epoch*) dan nilai *threshold*
 - 6: Sampai dengan jumlah iterasi terpenuhi lakukan :
 - 7: Hitung parameter pembelajaran pada setiap *layer* sampai dengan *output layer*
 - 8: Tetapkan fungsi aktivasi dan hitung nilai *output*
 - 9: Hitung penyimpangan *output* terhadap *target* dengan SSE
 - 10: Jika SSE > *threshold* maka lakukan :
 - 11: Hitung nilai gradien dari fungsi aktivasi terhadap setiap parameter pembelajaran dengan *Backpropagation*
 - 12: Perbarui nilai parameter pembelajaran dengan *gradient descent*
 - 13: Hitung output dan SSE
 - 14: Jika SSE \leq *threshold* maka **stop**
-

-
- 15: Simpan nilai parameter pembelajaran
 - 16: Lakukan analisis *grid search*
 - 17: Simpan kombinasi *hyperparameter* dengan akurasi terbaik
 - 18: Gunakan model terbaik untuk peramalan selanjutnya
-

Pada tahap peramalan, ketiga metode peramalan tersebut akan diterapkan dan dibandingkan kinerjanya pada beberapa sampel *part* yang mewakili masing-masing kelompok pola permintaan yang sudah terbentuk. Pada tahap ini, parameter model yang paling baik di beberapa sampel *part* tersebut, yang diukur dengan parameter MAPE dan MSE, akan ditetapkan sebagai model peramalan yang mewakili masing-masing kelompok pola permintaan di fase *training*. Seluruh tahap pengembangan model dan peramalan pada tugas akhir ini dikerjakan menggunakan program RStudio dengan bahasa R 4.0.0.

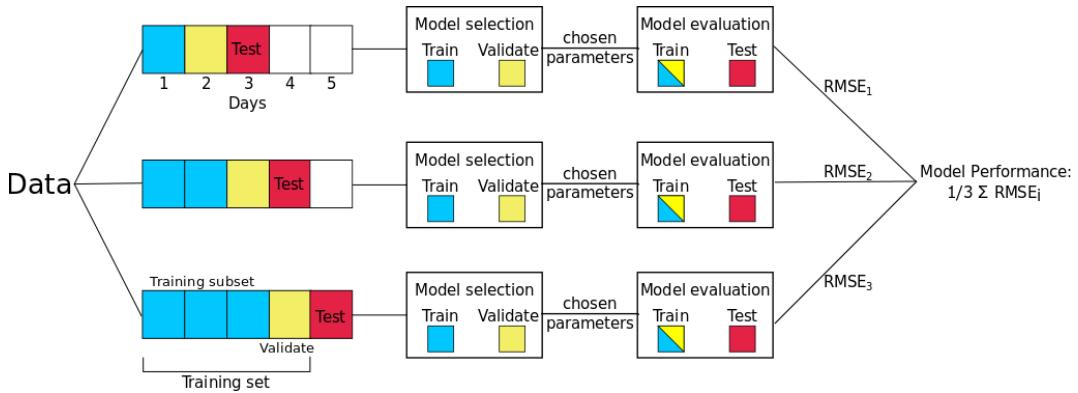
3.5 Validasi

Pada tahap validasi, seluruh model peramalan akan diuji konsistensi kinerjanya terhadap data *testing*. Data *testing* adalah data yang baru atau tidak pernah dilibatkan dalam fase *training*. Data *testing* bisa diperoleh dari luar atau dari dalam set data observasi. Contoh data *testing* dari luar observasi dalam tugas akhir ini adalah riwayat permintaan selain segmentasi pengamatan, seperti: permintaan *part* selain dari Arab Saudi dan permintaan *part* berdimensi *small*. Sedangkan data *testing* dari dalam observasi pada tugas akhir ini diperoleh dengan menerapkan *Time-Series Nested Cross-Validation* (Hyndman, 2016, Cochrane, 2018). Terdapat dua metode *Nested CV* yaitu *Second-Half Prediction* (**Gambar 3. 5**) dan *Day Forward-Chaining* (**Gambar 3. 6**).



Gambar 3. 5 Ilustrasi Metode *Second-Half Prediction*

(Cochrane, 2018)



Gambar 3. 6 Ilustrasi Metode *Day Forward-Chaining*

(Cochrane, 2018)

Setelah sejumlah pola permintaan *spare part* sebagai data *testing* diperoleh, maka masing-masing pola permintaan tersebut akan diukur kemiripannya dengan kelompok-kelompok pola permintaan *part* yang sudah terbentuk di fase *training*. Data *testing* yang sudah dikelompokkan tersebut kemudian diramalkan menggunakan model peramalan terbaik dari fase *training* sesuai kelompoknya masing-masing. Kinerja model peramalan di fase validasi ini juga diukur dengan parameter MAPE dan MSE. Model peramalan dianggap valid ketika kinerjanya baik secara konsisten atau memiliki nilai MAPE dan MSE saat validasi yang sama atau lebih kecil dari fase *training*.

Pada *spare part* yang memiliki beberapa permintaan aktual nol maka pengukuran kinerja peramalan menggunakan MAPE tidak bisa digunakan, karena permintaan aktual adalah denominator pada MAPE. Oleh karena itu terdapat beberapa alternatif agar tetap bisa mengukur relatif *error* peramalan yaitu:

1. Mengecualikan permintaan aktual nol dalam perhitungan MAPE. Sebagaimana yang dilakukan oleh sebagian *software* peramalan (Hoover, 2006).
2. Menggunakan *symmetric-MAPE* (SMAPE) yang didefinisikan sebagai berikut (Amstrong, 1985):

$$SMAPE = \frac{1}{n} \sum_{t=1}^n \frac{|A_t - F_t|}{(A_t + F_t)} \quad (3.2)$$

3. Menggunakan *Mean Absolute Scaled-Error* (MASE) yang didefinisikan sebagai berikut (Hyndman & Koehler, 2006):

$$MASE = \frac{1}{n} \sum_{t=1}^n \left(\frac{|A_t - F_t|}{\frac{1}{T-1} \sum_{t=2}^T |A_t - A_{t-1}|} \right) \quad (3.3)$$

Model yang sudah valid kemudian digunakan untuk meramalkan permintaan rata-rata dan intervalnya sebagai berikut (Hyndman & Athanasopoulos, 2018):

$$\hat{y}_{T+h|T} \pm 1.96 \cdot \hat{\sigma}_h \quad (3.4)$$

Di mana \hat{y}_t adalah nilai peramalan permintaan rata-rata dan $\hat{\sigma}_h$ adalah standar deviasi residual peramalan atau sama dengan standar deviasi *forecasting error*. Peramalan yang baik memiliki nilai residual yang tidak saling berkorelasi satu sama lain.

3.6 Interpretasi dan Analisis

Beberapa hal yang akan dianalisis dalam tugas akhir ini terdiri dari: interpretasi hasil dan perbandingan pengaruh perbedaan metode pengelompokan pola permintaan *spare part* (Boylan, K-Medoids, dan *Agglomerative Hierarchical Clustering*), interpretasi hasil dan perbandingan pengaruh perbedaan perlakuan data *training* (tanpa perlakuan, dengan *trend* dan *seasonal adjustment*, dan dengan transformasi stasioner), interpretasi hasil dan perbandingan pengaruh perbedaan parameter model peramalan (*smoothing* pada metode Croston dan modifikasinya, *hyperparameter* dan fungsi kernel pada SVR, serta *hyperparameter* dan fungsi aktivasi pada ANN), dan terakhir adalah interpretasi hasil validasi dan penentuan metode peramalan terbaik untuk masing-masing kelompok *spare part*.

3.7 Kesimpulan dan Saran

Pada tahap ini, penarikan kesimpulan dilakukan untuk menjawab tujuan penelitian dalam tugas akhir ini, yaitu mengenai metode apa yang paling baik untuk mengelompokkan dan meramalkan pola permintaan *spare part* di PT. X. Kemudian, saran juga akan diberikan sebagai rekomendasi perbaikan sistem peramalan permintaan *spare part* untuk PT. X dan sebagai referensi untuk pengembangan penelitian selanjutnya.

(Halaman ini sengaja dikosongkan)

BAB 4

PRA-PENGOLAHAN DAN PENGELOMPOKKAN DATA

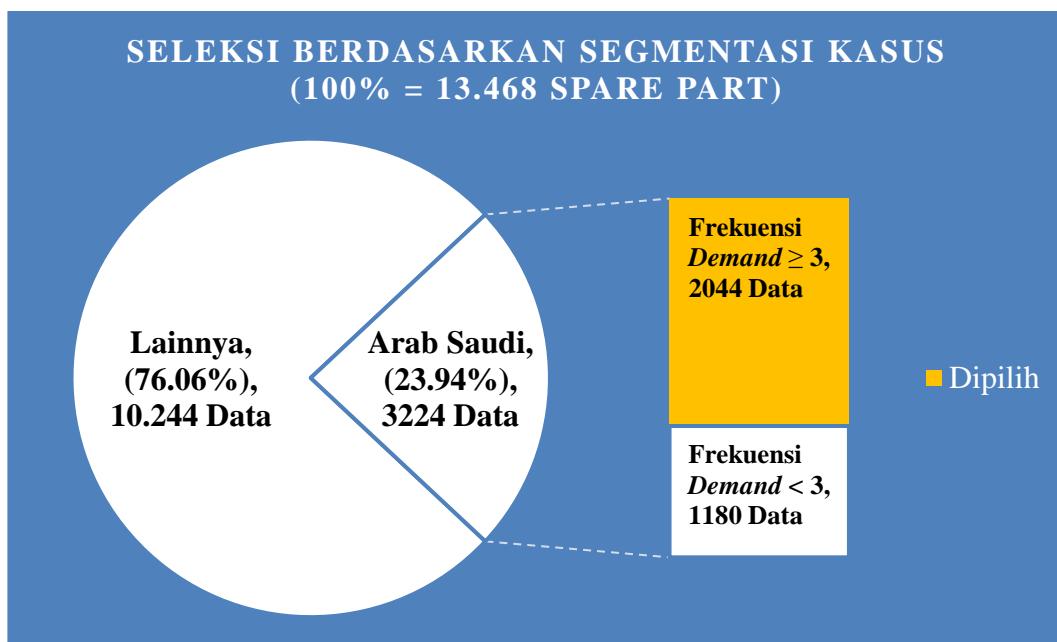
BAB 4

PRA-PENGOLAHAN DAN PENGELOMPOKKAN DATA

Pada bab ini akan dijelaskan mengenai proses pra-pengolahan data yang terdiri dari: seleksi data, pembersihan data, dan transformasi data. Kemudian, pengelompokkan data dengan metode klasifikasi Boylan dan teknik klaster. Terakhir, eksperimentasi terhadap data *training* yang terdiri dari *trend* dan *seasonal adjustment* serta transformasi stasioner.

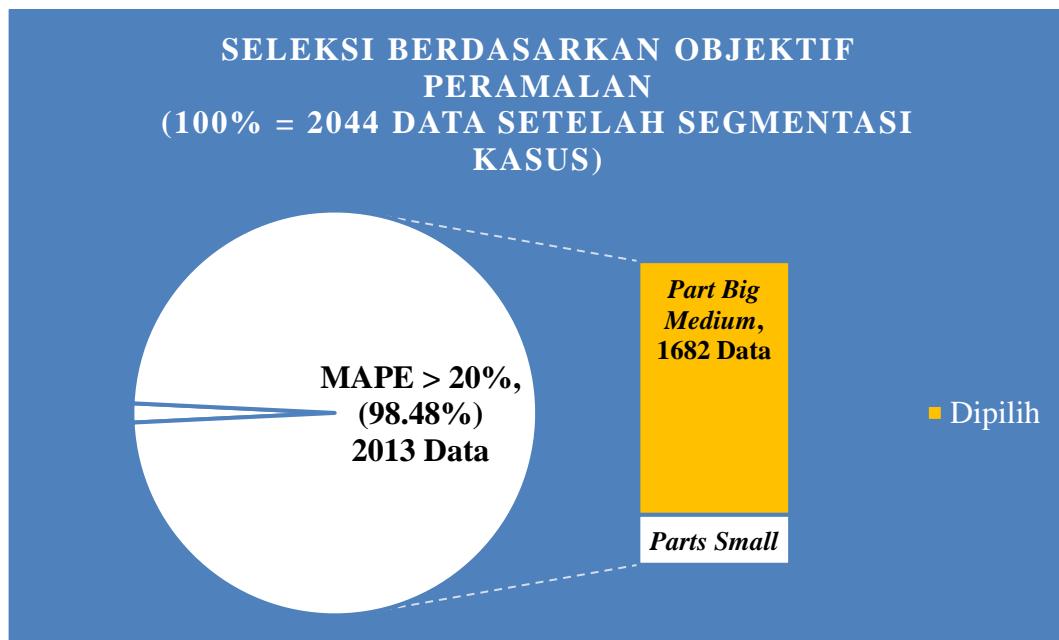
4.1 Seleksi dan Pembersihan Data Riwayat Permintaan Spare Part

Pada proses ini sejumlah 13.468 data riwayat permintaan *spare part* selama Januari 2018 sampai Januari 2019 dari PT. X akan diseleksi berdasarkan segmentasi kasus, yaitu segmen permintaan dari Arab Saudi. Setelah itu *part* dengan frekuensi permintaan dibawah tiga periode akan dibuang untuk menghindari bias peramalan sehingga diperoleh 2044 data hasil segmentasi kasus sebagai berikut (**Gambar 4. 1**).



Gambar 4. 1 Seleksi Data Berdasarkan Segmentasi Kasus

Setelah segmentasi kasus, data kemudian diseleksi kembali berdasarkan objektif peramalan, yaitu dengan memprioritaskan permintaan *part* yang tidak akurat ketika diramalkan dengan metode *existing*, yaitu diukur dengan MAPE lebih dari 20%, dan *part* yang berdimensi *big* dan *medium*, karena paling signifikan pengaruhnya ketika terjadi kesalahan peramalan.



Gambar 4. 2 Seleksi Data Berdasarkan Objektif Peramalan

Pada akhirnya diperoleh sejumlah 1682 data yang akan digunakan untuk proses-proses selanjutnya (**Gambar 4. 2**).

4.2 Transformasi Data Riwayat Permintaan *Spare Part*

Pada proses ini, sejumlah 1682 data yang telah dipilih sebelumnya akan ditransformasi ke bentuk data tertentu sebelum digunakan pada teknik klaster, metode peramalan SVR, dan ANN. Tujuannya adalah untuk meminimalisasi terjadinya bias pada metode-metode *data mining* (Santosa & Umam, 2018). Terdapat dua proses yaitu normalisasi skala data dan konversi ke bentuk *predictor* dan *target*.

4.2.1 Normalisasi Skala

Pada proses ini sejumlah 1682 data yang akan digunakan untuk teknik klaster dinormalisasi skalanya terlebih dahulu dengan menggunakan *min-max normalization* sehingga berada di skala [-1, 1] sebagai berikut.

$$\hat{X} = \frac{X - X_{min}}{X_{max} - X_{min}} * (1 - (-1)) + (-1) \quad (4.1)$$

Pada **Lampiran 1** adalah salah satu contoh implementasi normalisasi skala pada salah satu data permintaan *part* dengan menggunakan *software RStudio*.

4.2.2 Konversi Ke-Bentuk Predictor-Target

Proses ini dilakukan untuk mengonversi data ke dalam bentuk *predictor-target* sebelum menjadi *input* metode peramalan SVR dan ANN. Data dikonversi dengan cara melakukan *windowing* mengikuti indeks sebagai berikut.

$$\begin{aligned} & \text{Predictor} \rightarrow \text{Target} \\ & \text{Dataset} = s[1] \dots s[N] \\ & [s[k-3], s[k-2], s[k-1]] \rightarrow s[k] \\ & \quad : \\ & [s[N-3], s[N-2], s[N-1]] \rightarrow s[N] \end{aligned}$$

Pada **Lampiran 2** adalah salah satu contoh implementasi konversi ke-bentuk *predictor-target* pada salah satu data permintaan *part* dengan menggunakan *software RStudio* dan ukuran *window* tiga periode ($k = 3$).

4.3 Pengelompokkan Pola Permintaan Spare Part

Selanjutnya pada proses ini, sejumlah 1682 pola permintaan *part* yang sudah diperoleh sebelumnya akan dikelompokkan berdasarkan pendekatan klasifikasi Boylan, teknik klaster K-Medoids, dan teknik klaster Aglomeratif. Pada akhirnya, *part* pada masing-masing kelompok hasil ketiga pendekatan tersebut akan

dikelompokkan kembali berdasarkan *family partnya*, yaitu *eksterior*, *interior*, *engine*, dan *suspension*.

4.3.1 Klasifikasi Boylan

Pada pendekatan ini masing-masing part dari total 1682 akan dihitung terlebih dahulu parameter *Average Inter-Demand Interval* (ADI) atau (p) dan *Squared Coefficient of Variation*-nya (CV^2)-nya menggunakan rumus sebagai berikut.

$$ADI = \frac{\sum_{i=1}^n p_i}{n} \quad (4.2)$$

$$CV^2 = \left(\frac{\sqrt{\frac{\sum_{i=1}^N (\varepsilon_i - \bar{\varepsilon})^2}{N}}}{\bar{\varepsilon}} \right)^2 \quad (4.3)$$

$$\text{Di mana } \bar{\varepsilon} = \frac{\sum_{i=1}^N \varepsilon_i}{N}$$

Kemudian *part* dikelompokkan berdasarkan kriteria Boylan menggunakan *cut-off value* dari kedua parameter tersebut seperti pada **Tabel 4. 1** berikut.

Tabel 4. 1 Nilai *Cut-Off* untuk CV^2 dan ADI (2)

	$ADI \leq 1.32$	$ADI > 1.32$
$CV^2 > 0.49$	<i>Erratic</i>	<i>Lumpy</i>
$CV^2 \leq 0.49$	<i>Smooth</i>	<i>Intermittent</i>

Pada **Lampiran 3** adalah contoh implementasi klasifikasi Boylan pada salah satu data permintaan *part* dengan menggunakan *software* RStudio dan *library* “tsintermittent”. Kemudian pada Lampiran 4 adalah hasil klasifikasi Boylan dari keseluruhan data yang berjumlah 1682 pola permintaan *part*. Diketahui bahwa pada **Lampiran 4**, mayoritas pola permintaan *part* di PT. X adalah “LUMPY” sebesar 54.76%, kemudian “INTERMITTENT” sebesar 37.1%, “ERRATIC” sebesar 6.42%, dan “SMOOTH” sebesar 1.72%.

4.3.2 K-Medoids dan Agglomerative Hierarchical Clustering

Selain klasifikasi Boylan, teknik klaster juga digunakan sebagai pembanding untuk mengelompokkan sejumlah 1682 data yang sudah diperoleh sebelumnya. Ada dua macam teknik klaster yang akan diimplementasikan, yaitu klaster K-Medoids dan Aglomeratif. Pada semua metode, perbedaan antar pola permintaan diukur dengan jarak euclidian antar titik waktu (*time points / TP*) yang didefinisikan dengan jarak antara dua *time-series* TS1 dan TS2 sebagai berikut.

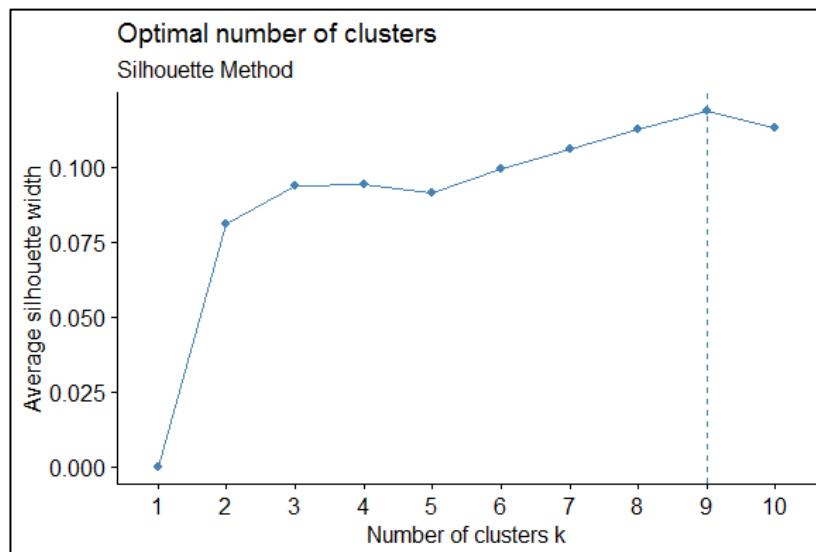
$$d(\text{TS1}, \text{TS2}) = \sqrt{\sum_{i=1}^{TP} (\text{TS1}_i - \text{TS2}_i)^2} \quad (4.4)$$

Kinerja paling baik di antara kedua teknik klaster diukur berdasarkan fungsi kriteria SSE terkecil yang didefinisikan sebagai berikut.

$$\begin{aligned} SSE &= \sum_{i=1}^k \sum_{x \in D_i} \|x - x_i\|^2 \\ &\text{MIN}\{SSE_{kmedoids}, SSE_{agglomeratif}\} \end{aligned} \quad (4.5)$$

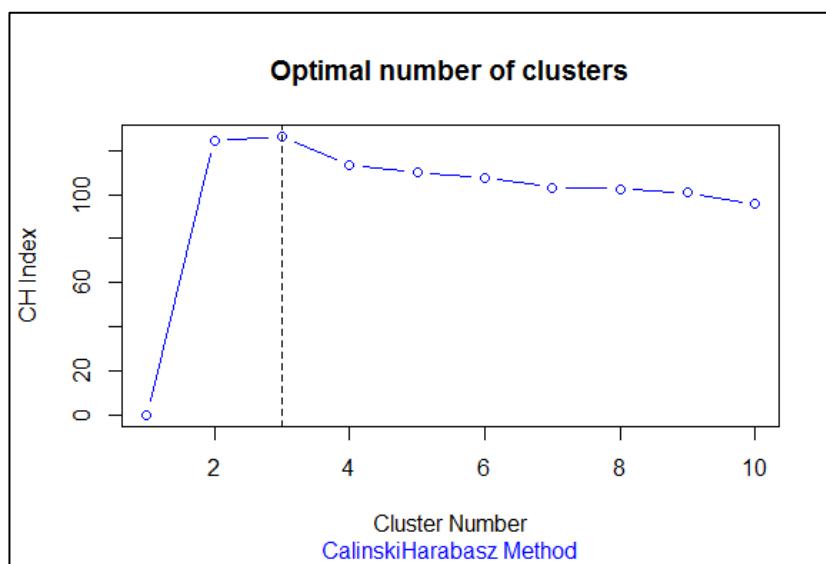
Selanjutnya, pada masing-masing teknik klaster perlu ditentukan terlebih dahulu beberapa parameter pengelompokan. Pada klaster K-Medoids parameter itu adalah jumlah klaster (k) yang optimal. k optimal bisa diperoleh dengan membandingkan *sum of squared error (SSE)* atau *sum of squared within (SSW)* antara hasil rekomendasi jumlah k dari uji CalinskiHarabasz dan Silhouette. k yang dipilih adalah yang memiliki SSE terkecil dan memaksimalkan nilai CalinskiHarabasz index atau Silhouette index.

Gambar 4. 3 merupakan *output* dari uji Silhouette untuk metode K-Medoids melalui *software* RStudio pada 1682 data yang sudah dinormalisasi skalanya.



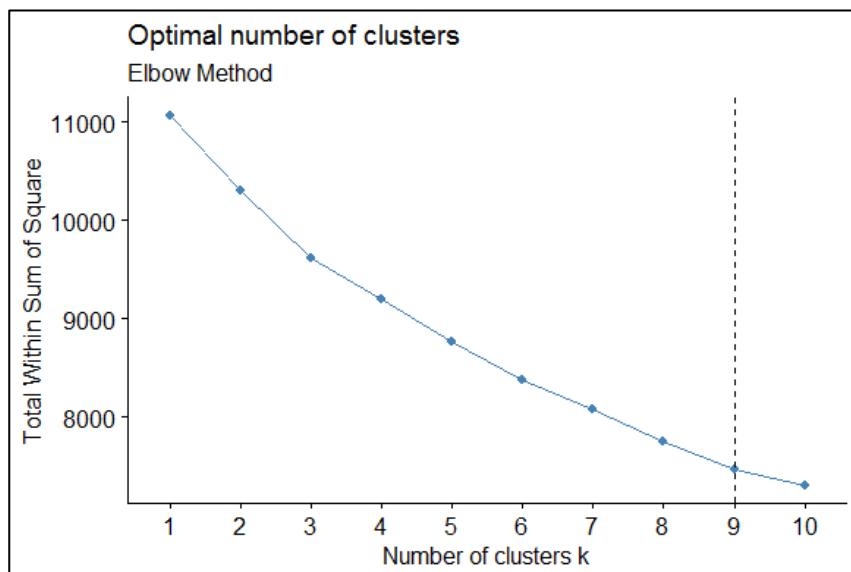
Gambar 4. 3 Rekomendasi Jumlah Klaster ($k = 9$) Hasil Uji Silhouette untuk Teknik K-Medoids

Gambar 4. 4 merupakan *output* dari uji CalinskiHarabasz untuk metode K-Medoids melalui *software* RStudio pada 1682 data yang sudah dinormalisasi skalanya.



Gambar 4. 4 Rekomendasi Jumlah Klaster ($k = 3$) Hasil Uji CalinskiHarabasz untuk Teknik K-Medoids

Kemudian ketika dibandingkan antara $k = 3$ (hasil CalinskiHarabasz) dan $k = 9$ (hasil Silhouette), diketahui bahwa $k = 9$ menghasilkan SSE atau SSW yang paling kecil sehingga jumlah klaster = 9 dipilih pada klaster K-Medoids (**Gambar 4. 5**).



Gambar 4. 5 Jumlah Klaster Optimal ($k = 9$) yang Meminimalkan SSE atau SSW

Sedangkan pada klaster Agglomeratif, parameter yang harus ditentukan adalah ukuran *linkage*. *Linkage* yang paling baik adalah yang memberikan nilai korelasi *cophenetic* paling kuat (mendekati 1 atau -1). Pada **Lampiran 5** adalah implementasi perhitungan korelasi *cophenetic* pada berbagai ukuran *linkage* melalui fungsi “agnes” atau *Agglomerative nesting* di RStudio. Setelah parameter masing-masing teknik klaster diperoleh, selanjutnya adalah proses utama, yaitu pengelompokan sejumlah 1682 pola permintaan *part* dengan kedua teknik klaster melalui software RStudio menggunakan library “cluster” dan “factoextra” seperti pada **Lampiran 6 dan 7**.

Skenario teknik klaster terbaik antara K-Medoids dengan jumlah klaster ($k = 9$) dan klaster Agglomeratif dengan jumlah k yang sama dipilih berdasarkan kriteria SSE atau SSW terkecil. Sehingga seperti yang diketahui dari **Lampiran 6 dan 7**, skenario klaster terbaik adalah K-Medoids dengan SSE paling kecil di antara keduanya, yaitu sebesar 7466.37 pada jumlah klaster ($k = 9$).

Hasil pengelompokan metode K-Medoids dengan jumlah klaster ($k = 9$) pada keseluruhan data yang berjumlah 1682 pola permintaan *part* dan sudah

dinormalisasi skalanya ditunjukkan pada **Lampiran 8**. Dari **Lampiran 8**, diketahui bahwa sebanyak 17.78% dari data, atau 299 permintaan *part*, menjadi klaster 3 (K3), 17.72% menjadi klaster 1 (K1), 16.76% menjadi klaster 4 (K4), 9.45% menjadi klaster 2 (K2), 8.92% menjadi klaster 8 (K8), 8.74% menjadi klaster 6 (K6), 7.31% menjadi klaster 5 (K5), 6.9% menjadi klaster 7 (K7), dan 6.42% menjadi klaster 9 (K9). Pola permintaan *part* yang menjadi pusat dari masing-masing klaster dapat dilihat pada **Lampiran 9**.

4.3.3 Pengelompokkan Berdasarkan Family Part

Setelah diperoleh hasil pengelompokan pola permintaan *part* dari klasifikasi Boylan = {“Smooth”, “Intermittent”, “Erratic”, “Lumpy”} dan K-Medoids = {“K1”, “K2”, … , “K9”}, *part* pada masing-masing kelompok tersebut akan dikelompokkan kembali berdasarkan kategori *family part* = { “Exterior”, “Interior”, “Engine”, “Suspension”}. Sehingga diperoleh data *training* dari kombinasi Boylan berdasarkan *family part* sebanyak 4 x 4 dan dari kombinasi K-Medoids berdasarkan *family part* sebanyak 9 x 4. Sejumlah 52 data *training* tersebut akan diberi berbagai perlakuan pada proses selanjutnya sebelum digunakan untuk peramalan.

4.4 Eksperimentasi Data *Training* Peramalan

Selanjutnya adalah proses eksperimentasi data *training* yang terdiri dari *trend* dan *seasonal adjustment*, serta *transformasi Box-Cox*, sehingga akan diperoleh: data *training* kontrol (tanpa perlakuan), data *training* hasil *adjustment*, dan data *training* hasil transformasi stasioner. Berbagai perlakuan ini pada akhirnya akan dibandingkan pengaruhnya terhadap kinerja peramalan.

4.4.1 Trend dan Seasonal Adjustment

Trend dan *seasonal analysis/detection* perlu dilakukan dahulu sebelum melakukan *trend adjustment* (memisahkan tren) dan *seasonal adjustment* (memisahkan musiman) dari data *time-series*. Untuk melakukan *trend analysis*, pada tahap ini akan dilakukan uji Mann-Kendall (Mann, 1945, Kendall, 1975). Hipotesis *null* dan alternatif pada uji ini adalah sebagai berikut.

H_0 : No monotonic trend

H_a : Monotonic trend is present

Pada **Lampiran 10** adalah contoh implementasi uji Mann-Kendall pada salah satu data permintaan *part* dengan menggunakan *library* “trend” pada *software RStudio*.

Dari uji Mann-Kendall pada **Lampiran 10** diperoleh nilai *p.value* yang signifikan untuk menolak hipotesis *null* ($p = 0.007 \leq 0.05$) sehingga menunjukkan kemunculan tren pada data. Untuk mengetahui kecenderungan tren, dapat dilakukan dengan melihat pada parameter *S* dari uji Mann-Kendall. Pada **Lampiran 10**, *S* yang positif menunjukkan kecenderungan observasi pada periode selanjutnya lebih besar dari observasi pada periode sebelumnya sehingga menunjukkan tren positif. Hal ini bisa dibuktikan dengan melakukan dekomposisi tren dari *time-series* seperti yang ditunjukkan pada **Lampiran 11**.

Selanjutnya *seasonal analysis* bisa dilakukan dengan *Fourier Fast Transform* (FFT) (Fourier, 1822). FFT melakukan dekomposisi terhadap semua frekuensi yang mungkin dari sebuah sinyal yang abstrak. Pada frekuensi (*f*) yang berbeda-beda, suatu sinyal tersebut akan berulang pada periode (*T*) yang berbeda-beda pula sebagai berikut.

$$T = \frac{1}{f} \quad (4.6)$$

Nilai *T* ini kemudian digunakan untuk mendeteksi periode *seasonality* dari suatu *time-series*, yaitu *T* pada frekuensi yang memiliki *power* paling besar. *Power* dari masing-masing frekuensi ini dibandingkan dalam suatu periodogram.

Pada **Lampiran 12** adalah contoh implementasi FFT pada salah satu data permintaan *part* dengan menggunakan *library* “TSA” pada *software RStudio*. Pada periodogram di **Lampiran 12** ditunjukkan bahwa frekuensi dengan *power* paling kuat pada “Dataset 2” berada sebelum 0.05 Hz (tepatnya 0.041 Hz). Sehingga *seasonality*-nya dapat dihitung seperti pada **Lampiran 13**. Dari **Lampiran 13**, diketahui bahwa *seasonality* pada “Dataset 2” adalah 24 periode sedangkan

“Dataset 2” berperiode 24 bulan, maka dari itu “Dataset 2” tidak memiliki komponen musiman. Pada **Lampiran 14** adalah implementasi FFT pada permintaan *part* yang lain. Pada **Lampiran 14** diketahui bahwa *seasonality* pada “Dataset 3” adalah 3 periode.

Setelah melakukan *trend* dan *seasonal analysis* maka selanjutnya dilakukan *adjustment* pada data dengan menyesuaikan kemunculan komponen tren atau musiman pada data tersebut. Pada **Lampiran 15** adalah contoh implementasi *trend adjustment* (memisahkan tren) pada salah satu permintaan *part* menggunakan RStudio. Kemudian pada **Lampiran 16** adalah contoh implementasi *seasonal adjustment* (memisahkan *seasonal*) pada salah satu permintaan *part* menggunakan RStudio.

4.4.2 Transformasi Stasioner Box-Cox dan Differencing

Selanjutnya adalah transformasi data *training* menjadi stasioner. Tujuan dari transformasi stasioner pada *time-series* adalah untuk mengurangi kompleksitas peramalan. Terdapat dua macam stasioneritas, yaitu stasioner dalam varian dan rataan. Stasioner dalam varian bisa diukur dengan melakukan transformasi Box-Cox yang merubah suatu nilai *time-series* (y_t) menjadi:

$$w_t = \begin{cases} \ln(y_t), & \lambda = 0 \\ \frac{y_t^\lambda - 1}{\lambda}, & \lambda \neq 0 \end{cases} \quad (4.7)$$

Sebelum melakukan transformasi Box-Cox, nilai Lambda (λ) perlu dicari terlebih dahulu. Lambda yang optimal bisa diperoleh dengan pendekatan Guerrero (Guerrero, 1993). Pada **Lampiran 17** adalah implementasi transformasi Box-Cox menggunakan *library* “DescTools” dan “forecast” pada RStudio.

Hasil transformasi Box-Cox adalah distribusi data yang seringkali lebih mendekati distribusi normal. Hal tersebut bisa ditunjukkan dengan histogram dan menghitung *skewness* dari data hasil transformasi Box-Cox. *Skewness* didefinisikan sebagai *moment* statistik orde 3:

$$Skewness = \frac{\sum_{i=1}^N (X_i - \bar{X})^3}{(N - 1)\sigma^3} \quad (4.8)$$

Distribusi data yang lebih mendekati distribusi normal memiliki nilai *skewness* yang lebih mendekati nol. Pada **Lampiran 18 dan 19** adalah contoh implementasinya menggunakan library “moments” dan “MASS” pada R.

Selanjutnya, untuk menguji apakah data *time-series* stasioner dalam rataan, bisa melakukan *Augmented Dickey-Fuller Test* (ADF Test) (Dickey & Fuller, 1979). ADF bisa digunakan untuk mengetahui orde atau berapa kali *order-differencing* yang dibutuhkan sehingga *time-series* menjadi stasioner dalam rataan. Orde bernilai nol menunjukkan *time-series* sudah stasioner dalam rataan sehingga tidak memerlukan *order-differencing*. Implementasinya menggunakan *library* “forecast” pada RStudio ditunjukkan pada **Lampiran 20**.

Setelah mengetahui orde untuk *order-differencing*, maka perlu dicek apakah *seasonality* dari *time-series* cukup signifikan untuk mempengaruhi stasioneritas pada *time-series*. Hal ini bisa dilakukan dengan menentukan orde *seasonal differencing* menggunakan pendekatan Canova-Hansen (CH) (Canova & Hansen, 1995). Apabila orde *seasonal differencing* yang diperoleh bernilai lebih dari nol, maka *seasonal differencing* perlu dilakukan terlebih dahulu sebelum *order-differencing*. Sedangkan apabila orde *seasonal differencing* nol, maka *seasonal differencing* tidak perlu dilakukan. Pada **Lampiran 21** ditunjukkan penentuan orde *seasonal differencing* menggunakan *library* “forecast” pada RStudio.

Pada **Lampiran 21** diketahui bahwa pada “Dataset 1” hanya perlu dilakukan *first order differencing* tanpa diawali dengan *seasonal differencing*. Pada **Lampiran 22** adalah hasilnya ketika menggunakan *library* “forecast” pada RStudio. Stasioneritas “Dataset 1” setelah *order-differencing* pada **Lampiran 22** juga bisa dibuktikan dengan uji ADF, di mana hipotesis pada uji ADF adalah sebagai berikut.

$$H_0: y_t \text{ is random walk}$$

$$H_a: y_t \text{ is stationary process}$$

Pada **Lampiran 23** adalah hasil uji ADF-nya ketika menggunakan fungsi “adf.test” dengan *library* “tseries” pada RStudio.

4.4.3 Ringkasan Data Training untuk Peramalan

Setelah semua eksperimentasi diimplementasikan, maka diperoleh berbagai kombinasi data *training* untuk *input* peramalan. Pada tugas akhir ini, proses pengembangan model peramalan akan didemonstrasikan menggunakan sebagian dari berbagai kombinasi tersebut. Sebelumnya, data *training* hasil *trend adjustment* diputuskan untuk disisihkan atau tidak digunakan dalam proses pengembangan model peramalan untuk menghindari bias karena memiliki nilai *missing value* (NA) yang banyak, seperti yang bisa dilihat pada **Lampiran 15**. Kombinasi data *training* yang terpilih adalah sebagai berikut.

1. Data *training* untuk menguji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja model peramalan. Data yang dipilih memiliki label:

“smooth_exterior”	“intermittent_exterior”
“erratic_exterior”	“lumpy_exterior”

Family part exterior dipilih karena ada di semua klasifikasi pola permintaan.

2. Data *training* untuk menguji pengaruh perbedaan *family part* terhadap kinerja model peramalan. Data yang dipilih memiliki label:

“lumpy_exterior”	“lumpy_interior”
“lumpy_engine”	“lumpy_suspension”

Klasifikasi pola permintaan *lumpy* dipilih karena memiliki kinerja peramalan *existing* yang paling buruk dan menyumbang sebagian besar kesalahan peramalan pada PT. X.

3. Data *training* untuk menguji pengaruh perubahan-perubahan akibat eksperimentasi pada data permintaan yang sama terhadap kinerja model

peramalan. Eksperimentasinya adalah: tanpa eksperimentasi (kontrol), *seasonal adjustment*, transformasi Box-Cox, dan *differencing*. Data yang dipilih memiliki label:

“kontrol_intermittent_engine” “seasadj_intermittent_engine”
“boxcox_intermittent_engine” “diff_lumpy_suspension”

Klasifikasi pola permintaan *intermittent* dan *family part engine* dipilih karena memiliki semua kombinasi perlakuan pada data.

4. Data *training* untuk menguji pengaruh perbedaan pengelompokkan *part* hasil K-Medoids *Clustering* terhadap kinerja model peramalan. Masing-masing data mewakili klaster satu sampai sembilan. Data yang dipilih memiliki label:

“k1_suspension” “k2_suspension” “k3_suspension”
“k4_suspension” “k5_suspension” “k6_suspension”
“k7_suspension” “k8_suspension” “k9_suspension”

Family part suspension dipilih karena ada di semua klaster dan memiliki kinerja peramalan *existing* yang paling buruk.

BAB 5

PERAMALAN SPAREPART DAN VALIDASI

BAB 5

PERAMALAN SPARE PART DAN VALIDASI

Pada bab ini akan dijelaskan mengenai proses peramalan dan validasi peramalan permintaan *spare part* di PT. X, yang terdiri dari: (1) pengembangan dan validasi model peramalan Croston, SBA, dan TSB, (2) pengembangan dan validasi model peramalan SVR, (3) pengembangan dan validasi model peramalan ANN, dan (4) implementasi model untuk peramalan *multi-step* dan perhitungan interval peramalan.

5.1 Pengembangan dan Validasi Model Peramalan Croston, SBA, dan TSB

Model peramalan pertama yang akan dikembangkan adalah model dengan metode peramalan Croston, SBA, dan TSB. Sebagaimana diketahui pada **Sub-bab 3.4**, model peramalan ini memiliki *hyperparameter* yang ditentukan sendiri oleh pemodel sehingga perlu dioptimalkan. Proses optimasi *hyperparameter* tersebut akan dijelaskan pada bagian struktur model. Sedangkan tahap validasi model akan dibagi menjadi dua bagian, yaitu: pengembangan data *training* dan *testing* dan uji signifikansi pengaruh skenario *input* peramalan seperti pada **Sub-sub-bab 4.4.3** terhadap kinerja model peramalan Croston, SBA, dan TSB.

5.1.1 Struktur Model

Seperti yang diketahui pada **Tabel 3. 3**, metode Croston, SBA, dan TSB memiliki hyperparameter yaitu *smoothing parameter* α dan β (untuk TSB). Nilai α dan β berada di antara 0 sampai 1 ($0 \leq \alpha, \beta \leq 1$). Nilai α dan β bisa dioptimalkan dengan optimasi nonlinear yang meminimalkan *error* peramalan (dalam hal ini MSE) sebagai berikut.

$$\min MSE = \frac{1}{n} \sum_{t=1}^n e_t^2 \quad (5.1)$$

Subject to:

$$0 \leq \alpha \leq 1 \quad (5.2)$$

$$0 \leq \beta \leq 1 \quad (5.3)$$

Selain itu, pada metode ini, output peramalan, yaitu estimasi permintaan hasil *smoothing* (Y_t) dan estimasi interval permintaan (p_t), seperti yang dijelaskan pada **Sub-bab 2.4**, diperiode awal ($t = 0$) harus diawali dengan suatu nilai inisial. Oleh karena itu, struktur model peramalannya menjadi seperti yang tertera pada **Tabel 5. 1**:

Tabel 5. 1 Struktur Model Peramalan Croston, SBA, dan TSB

α	Dioptimalkan dengan optimasi nonlinear dalam rentang [0,1]
β	Dioptimalkan dengan optimasi nonlinear dalam rentang [0,1]
$Y_{t=0}$	Estimasi nilai permintaan di awal periode memiliki nilai sama dengan nilai permintaan bukan nol pertama pada data <i>training</i> (<i>naive</i>)
$p_{t=0}$	Estimasi interval permintaan di awal periode memiliki nilai sama dengan rata-rata interval permintaan aktual (q_T) pada data <i>training</i> .

5.1.2 Validasi Model

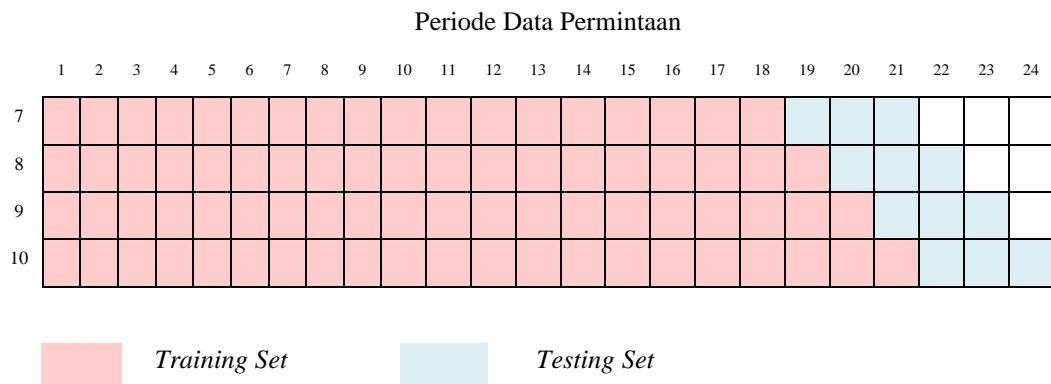
Pada tahap ini akan dijelaskan menjadi dua bagian, yaitu tahap pengembangan data *training* dan *testing* dan uji signifikansi pengaruh skenario *input* peramalan.

- **Pengembangan Data *Training* dan *Testing***

Pada model ini, data *training* dan *testing* yang digunakan untuk mengembangkan model diperoleh dengan melakukan pendekatan *Day-Forward Chaining* seperti yang dijelaskan pada **Sub-bab 3.5**. Pendekatan tersebut pada tahap ini dilakukan untuk menghasilkan sepuluh kombinasi data *training* dan *testing* yang dapat dilihat pada **Tabel 5. 2** berikut.

Tabel 5. 2 Data *Training* dan *Testing* Untuk Model Croston, SBA, dan TSB

		Periode Data Permintaan																							
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	2																								
3	4																								
5	6																								



- **Uji Signifikansi Pengaruh Skenario Berbagai *Input* Peramalan**

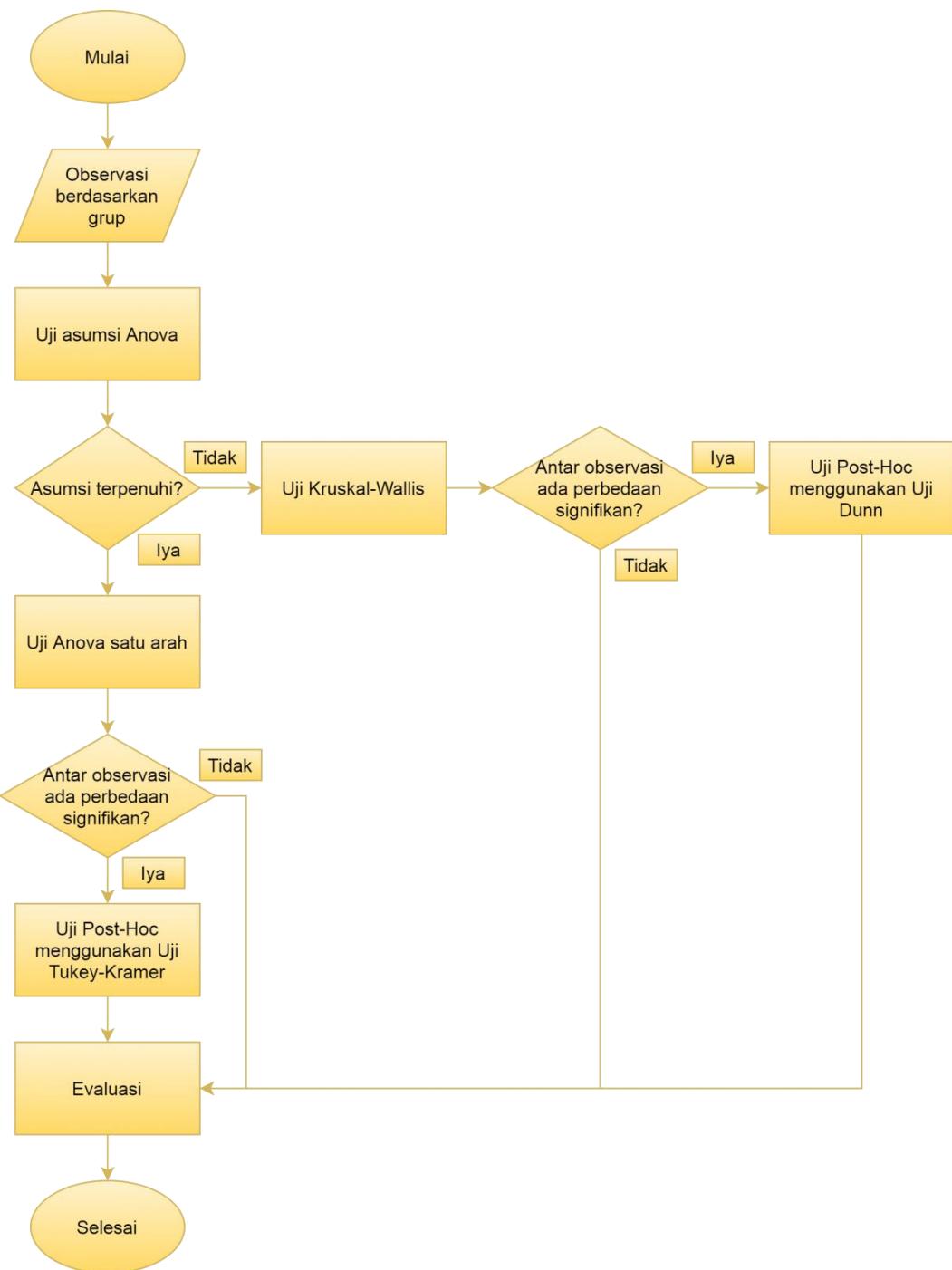
Pada tahap ini, berbagai skenario *input* peramalan seperti yang dibahas pada **Sub-sub-bab 4.4.3** akan diuji pengaruhnya terhadap model peramalan Croston, SBA, dan TSB. Tujuan dari uji ini adalah untuk membandingkan *forecasting error* (dalam hal ini MSE) dan *forecasting robustness* model peramalan di berbagai kondisi *input* peramalan. Metode pengujian yang digunakan adalah metode parametrik analisis varian satu arah (*Anova One-Way*) dan non-parametrik yaitu uji Kruskal-Wallis.

Seperti yang diketahui dari Snedecor & Cochran (1967), terdapat beberapa asumsi yang perlu dipenuhi sebelum melakukan analisis varian satu arah, yaitu:

1. Homogenitas varian – nilai varian dari observasi dalam grup/faktor harus sama. Asumsi ini bisa diuji pada grup lebih dari dua dengan pendekatan Levene dengan hipotesis alternatif bahwa varian sebenarnya tidak sama (Levene, 1960).
2. Independen – observasi dalam grup/faktor harus saling bebas. Asumsi ini bisa diuji dengan parameter statistik student-t dengan hipotesis alternatif bahwa selisih *mean* antar observasi tidak sama dengan nol (ada perbedaan signifikan antar observasi).
3. Normalitas – nilai residual dari Anova harus berdistribusi normal. Asumsi ini bisa diuji dengan pendekatan Shapiro-Wilk dengan hipotesis alternatif bahwa residual tidak berdistribusi normal (Shapiro & Wilk, 1965).

Apabila asumsi-asumsi tersebut tidak terpenuhi, maka pada data akan dilakukan uji alternatif dari Anova, yaitu uji varian satu arah non-parametrik Kruskal-Wallis. Pada uji Kruskal-Wallis, statistik uji yang digunakan adalah

median dari observasi dan asumsi-asumsi tersebut di atas tidak diperlukan (Kruskal & Wallis, 1952). Diagram alur dari tahap ini dapat dilihat pada **Gambar 5. 1**.



Gambar 5. 1 Uji Signifikansi Pengaruh Skenario Berbagai *Input* Peramalan

Proses tersebut di atas akan diterapkan pada keempat skenario *input* peramalan seperti dijelaskan pada **Sub-sub-bab 4.4.3** sebagai berikut hasilnya.

1. Hasil uji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja model peramalan Croston, SBA, dan TSB

Langkah pertama yang dilakukan adalah visualisasi data yang digunakan pada skenario ini dengan metode peramalan Croston, yaitu nilai MSE tahap *training* dan *testing* dari metode peramalan Croston ketika diuji dengan berbagai pola permintaan hasil klasifikasi Boylan ditunjukkan pada **Lampiran 24**. Pada **Lampiran 24** diketahui bahwa pada skenario ini, sebagian nilai rata-rata MSE *testing* dari metode peramalan Croston lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada pola permintaan *erratic* dan *intermittent* tidak terjadi *overfitting*.

Selanjutnya adalah uji asumsi untuk Anova satu arah pada skenario ini dengan metode peramalan Croston, hasilnya ditunjukkan pada **Lampiran 25**. Dari **Lampiran 25**, diketahui bahwa dalam skenario ini, pada MSE *training* dan *testing* dari metode peramalan Croston, nilai varian antar pola pemintaan Boylan berbeda secara signifikan ($p\text{-value} \leq 0.05$), maka dari itu tidak memenuhi salah satu asumsi Anova yaitu homogenitas varian sehingga pada tahap selanjutnya akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja metode peramalan Croston dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 26**. Berdasarkan **Lampiran 26** diketahui bahwa baik pada tahap *training* maupun *testing*, klasifikasi Boylan memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode Croston (karena ada perbedaan signifikan pada lebih dari satu pola permintaan dan menghasilkan $p\text{-value} \leq 0.05$).

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan SBA pada skenario ini ditunjukkan pada **Lampiran 27**. Dari **Lampiran 27**, diketahui bahwa pada skenario ini, sebagian nilai rata-rata MSE *testing* dari metode peramalan SBA lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada pola permintaan *erratic* dan *intermittent* tidak terjadi *overfitting*.

Kemudian, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan SBA ditunjukkan pada **Lampiran 28**. Berdasarkan **Lampiran 28**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$)

sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan SBA dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja metode peramalan SBA dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 29**. Berdasarkan **Lampiran 29** diketahui bahwa baik pada tahap *training* maupun *testing*, klasifikasi Boylan memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode SBA (karena ada perbedaan signifikan pada lebih dari satu pola permintaan dan menghasilkan $p\text{-value} \leq 0.05$).

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan TSB pada skenario ini ditunjukkan pada **Lampiran 30**. Dari **Lampiran 30**, diketahui bahwa pada skenario ini, sebagian besar nilai rata-rata MSE *testing* dari metode peramalan TSB lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, terjadi *overfitting* pada pola permintaan *intermittent*, *lumpy*, dan *smooth*.

Kemudian, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan TSB ditunjukkan pada **Lampiran 31**. Berdasarkan **Lampiran 31**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja metode peramalan TSB dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 32**. Berdasarkan **Lampiran 32** diketahui bahwa baik pada tahap *training* maupun *testing*, klasifikasi Boylan memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode TSB (karena ada perbedaan signifikan pada lebih dari satu pola permintaan dan menghasilkan $p\text{-value} \leq 0.05$).

2. Hasil uji pengaruh perbedaan *family part* terhadap kinerja model peramalan Croston, SBA, dan TSB

Selanjutnya, evaluasi kedua adalah pengaruh perbedaan *family part*. Visualisasi nilai MSE tahap *training* dan *testing* dari metode peramalan Croston terhadap perbedaan *family part* ditunjukkan pada **Lampiran 33**. Dari **Lampiran 33**, diketahui bahwa pada skenario ini, semua nilai rata-rata MSE *testing* dari

metode peramalan Croston lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada semua *family part* terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan Croston dalam skenario ini ditunjukkan pada **Lampiran 34**. Berdasarkan **Lampiran 34**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan Croston dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan *family part* terhadap kinerja metode peramalan Croston dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 35**. Berdasarkan **Lampiran 35** diketahui bahwa pada tahap *training* dan *testing* pengaruh *family part* tidak terlalu signifikan terhadap kinerja peramalan Metode Croston (karena perbedaan terjadi hanya dari satu *family part*).

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan SBA pada skenario ini ditunjukkan pada **Lampiran 36**. Dari **Lampiran 36**, diketahui bahwa pada skenario ini, semua nilai rata-rata MSE *testing* dari metode peramalan SBA lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada semua *family part* terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan SBA dalam skenario ini ditunjukkan pada **Lampiran 37**. Berdasarkan **Lampiran 37**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan SBA dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan *family part* terhadap kinerja metode peramalan SBA dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 38**. Berdasarkan **Lampiran 38** diketahui bahwa pada tahap *training* dan *testing* pengaruh *family part* tidak terlalu signifikan terhadap kinerja peramalan Metode SBA (karena perbedaan terjadi hanya dari satu *family part* dan $p\text{-value}$ dari MSE *testing* melebihi 0.05).

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan TSB pada skenario ini ditunjukkan pada **Lampiran 39**. Dari **Lampiran 39**, diketahui bahwa pada skenario ini, semua nilai rata-rata MSE *testing* dari metode peramalan TSB lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada semua *family part* terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini ditunjukkan pada **Lampiran 40**. Berdasarkan **Lampiran 40**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan *family part* terhadap kinerja metode peramalan TSB dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 41**. Berdasarkan **Lampiran 41**, diketahui bahwa pada tahap *training*, *family part* memberikan pengaruh yang cukup signifikan terhadap kinerja peramalan Metode TSB (karena ada perbedaan signifikan pada lebih dari satu *family part* di tahap *training*).

3. Hasil uji pengaruh perubahan-perubahan akibat eksperimentasi pada data permintaan yang sama terhadap kinerja model peramalan Croston, SBA, dan TSB

Evaluasi ketiga adalah menguji pengaruh eksperimentasi pada data *input* peramalan. Visualisasi nilai MSE tahap *training* dan *testing* dari metode peramalan Croston terhadap berbagai perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama ditunjukkan pada **Lampiran 42**. Dari **Lampiran 42**, diketahui bahwa pada skenario ini, sebagian besar nilai rata-rata MSE *testing* dari metode peramalan Croston tidak berbeda secara signifikan dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada data setelah transformasi Box-Cox, *differencing*, dan *seasonal adjustment* tidak terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan Croston dalam skenario ini ditunjukkan pada **Lampiran 43**.

Berdasarkan **Lampiran 43**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan Croston dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap kinerja metode peramalan Croston dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 44**. Berdasarkan **Lampiran 44**, diketahui bahwa baik pada tahap *training* maupun *testing*, eksperimentasi terhadap data *input* memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode Croston (karena ada perbedaan signifikan pada lebih dari satu eksperimentasi data *input* dan menghasilkan $p\text{-value} \leq 0.05$).

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan SBA pada skenario ini ditunjukkan pada **Lampiran 45**. Dari **Lampiran 45**, diketahui bahwa pada skenario ini, semua nilai rata-rata MSE *testing* dari metode peramalan SBA tidak berbeda secara signifikan dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada semua eksperimentasi data *input* tidak terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan SBA dalam skenario ini ditunjukkan pada **Lampiran 46**. Berdasarkan **Lampiran 46**, diketahui bahwa asumsi homogenitas varian pada tahap *training* terpenuhi ($p\text{-value} > 0.05$). dan pada tahap *testing* tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada tahap selanjutnya data MSE *testing* dari metode peramalan SBA dalam skenario ini akan diuji menggunakan Kruskal-Wallis sedangkan MSE *training* akan diuji terhadap asumsi ketiga Anova yaitu normalitas residual.

Hasil uji asumsi normalitas residual Anova satu arah pada MSE *training* dari metode peramalan SBA dalam skenario ini ditunjukkan pada **Lampiran 47**. Berdasarkan **Lampiran 47**, diketahui bahwa asumsi normalitas residual pada MSE *training* dari metode peramalan SBA tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga data MSE *training* dari metode peramalan SBA dalam skenario ini juga akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap kinerja metode peramalan SBA dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 48**. Berdasarkan **Lampiran 48**, diketahui bahwa baik pada tahap *training* maupun *testing*, eksperimentasi terhadap data *input* memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode SBA (karena ada perbedaan signifikan pada lebih dari satu eksperimentasi data *input* dan menghasilkan $p\text{-value} \leq 0.05$).

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan TSB pada skenario ini ditunjukkan pada **Lampiran 49**. Dari **Lampiran 49**, diketahui bahwa pada skenario ini, semua nilai rata-rata MSE *testing* dari metode peramalan TSB tidak berbeda secara signifikan dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada semua eksperimentasi data *input* tidak terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini ditunjukkan pada **Lampiran 50**. Berdasarkan **Lampiran 50**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap kinerja metode peramalan TSB dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 51**. Berdasarkan **Lampiran 51**, diketahui bahwa baik pada tahap *training* maupun *testing*, eksperimentasi terhadap data *input* memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode TSB (karena ada perbedaan signifikan pada lebih dari satu eksperimentasi data *input* dan menghasilkan $p\text{-value} \leq 0.05$).

4. **Hasil uji pengaruh perbedaan pengelompokan *part* hasil K-Medoids *Clustering* terhadap kinerja model peramalan Croston, SBA, dan TSB**
Evaluasi terakhir dari validasi model peramalan Croston, SBA, dan TSB adalah uji pengaruh pengelompokan pola permintaan hasil K-Medoids *Clustering*.

Visualisasi nilai MSE tahap *training* dan *testing* dari metode peramalan Croston terhadap berbagai klaster pola permintaan ditunjukkan pada **Lampiran 52**. Dari **Lampiran 52**, diketahui bahwa pada skenario ini, sebagian nilai rata-rata MSE *testing* dari metode peramalan Croston lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada klaster 1, 2, 3, 7, dan 9 terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan Croston dalam skenario ini ditunjukkan pada **Lampiran 53**. Berdasarkan **Lampiran 53**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan Croston dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh klaster K-Medoids terhadap kinerja metode peramalan Croston dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 54**. Berdasarkan **Lampiran 54** diketahui bahwa baik pada tahap *training* maupun *testing*, klaster K-Medoids memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode Croston karena ada perbedaan signifikan pada lebih dari empat klaster K-Medoids (melebihi klasifikasi Boylan) sehingga ada cukup bukti yang menyatakan bahwa pola permintaan *spare part* di PT. X bisa dibedakan menjadi 9 klaster pola permintaan pada metode peramalan Croston.

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan SBA pada skenario ini ditunjukkan pada **Lampiran 55**. Dari **Lampiran 55**, diketahui bahwa pada skenario ini, sebagian nilai rata-rata MSE *testing* dari metode peramalan SBA lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada klaster 1, 2, 3, 7, dan 9 terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan SBA dalam skenario ini ditunjukkan pada **Lampiran 56**. Berdasarkan **Lampiran 56**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan SBA dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh klaster K-Medoids terhadap kinerja metode peramalan SBA dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 57**. Berdasarkan **Lampiran 57**, diketahui bahwa baik pada tahap *training* maupun *testing*, klaster K-Medoids memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode SBA karena ada perbedaan signifikan pada lebih dari empat klaster K-Medoids (melebihi klasifikasi Boylan) sehingga ada cukup bukti yang menyatakan bahwa pola permintaan *spare part* di PT. X bisa dibedakan menjadi 9 klaster pola permintaan pada metode peramalan SBA.

Selanjutnya visualisasi data nilai MSE *training* dan *testing* dari metode peramalan TSB pada skenario ini ditunjukkan pada **Lampiran 58**. Dari **Lampiran 58**, diketahui bahwa pada skenario ini, sebagian besar nilai rata-rata MSE *testing* dari metode peramalan TSB lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan dalam skenario ini, pada klaster 1, 2, 3, 4, 7, 8, dan 9 terjadi *overfitting*.

Selanjutnya uji asumsi Anova satu arah pada MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini ditunjukkan pada **Lampiran 59**. Berdasarkan **Lampiran 59**, diketahui bahwa asumsi homogenitas varian tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga pada tahap selanjutnya data MSE *training* dan *testing* dari metode peramalan TSB dalam skenario ini akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh klaster K-Medoids terhadap kinerja metode peramalan TSB dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 60**. Berdasarkan **Lampiran 60**, diketahui bahwa baik pada tahap *training* maupun *testing*, klaster K-Medoids memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode TSB karena ada perbedaan signifikan pada lebih dari empat klaster K-Medoids (melebihi klasifikasi Boylan) sehingga ada cukup bukti yang menyatakan bahwa pola permintaan *spare part* di PT. X bisa dibedakan menjadi 9 klaster pola permintaan pada metode peramalan TSB.

5.2 Pengembangan dan Validasi Model Peramalan SVR

Seperti yang dijelaskan pada **Sub-bab 3.4**, pada model peramalan SVR terdapat *hyperparameter* yang perlu disesuaikan dengan teknik *grid search*. Proses penyesuaian *hyperparameter* (*hyperparameter tuning*) tersebut akan dijelaskan

pada bagian struktur model. Kemudian pada bagian validasi model akan dijelaskan komposisi data *training* dan *testing* yang digunakan untuk mengembangkan model dan hasil uji signifikansi pengaruh berbagai skenario dari **Sub-sub-bab 4.4.3** terhadap kinerja model peramalan SVR.

5.2.1 Struktur Model

Seperti yang dijelaskan pada **Sub-bab 3.4**, *hyperparameter* dari SVR yang perlu disesuaikan adalah batas epsilon (ϵ), biaya penalti melewati epsilon (C), dan parameter gamma untuk fungsi kernel (γ). Kombinasi dari ketiga *hyperparameter* tersebut yang paling optimal, yaitu yang meminimalkan *forecasting error* dan mempertimbangkan *forecasting robustness*, bisa dicari dengan teknik *grid search*. Oleh karena itu masing-masing *hyperparameter* tersebut harus memiliki *grid space*, yaitu rentang atau himpunan nilai-nilai yang harus dicoba pada *grid-search*. Struktur model peramalan SVR sekaligus *grid space* yang digunakan dapat dilihat pada **Tabel 5. 3**.

Tabel 5. 3 Struktur Model Peramalan SVR dengan 2288 kombinasi *hyperparameter*

ϵ	: Dengan <i>grid space</i> = {0, 0.1, 0.2, ..., 0.9, 1} (11 nilai)
C	: Dengan <i>grid space</i> = { $2^2, 2^{2.66}, 2^{3.32}, \dots, 2^{9.26}, 2^{9.92}$ } (13 nilai)
γ	: Dengan <i>grid space</i> = { $10^{-10}, 10^{-9.34}, 10^{-8.68}, \dots, 10^{-0.76}, 10^{-0.10}$ } (16 nilai)
Fungsi Kernel	: Kernel <i>Radial Basis Function</i> (RBF) = $K(x, y) = \exp(-\gamma \ x - x_i\ ^2)$

5.2.2 Validasi Model

Pada tahap ini akan dijelaskan menjadi dua bagian, yaitu tahap pengembangan data *training* dan *testing* dan uji signifikansi pengaruh skenario *input* peramalan.

- **Pengembangan Data Training dan Testing**

Seperti yang dijelaskan pada **Sub-sub-bab 4.2.2** data yang digunakan untuk mengembangkan model SVR adalah bentuk *predictor target* dari masing-masing skenario *input* pada **Sub-sub-bab 4.4.3**. Masing-masing *predictor* (P) dan *target*

(T) tersebut dibagi menjadi enam kombinasi data *training*, *validation*, dan *testing* yang terlihat pada **Tabel 5. 4**.

Tabel 5. 4 Data *Training* dan *Testing* Untuk Model SVR

PERIODE/VARIABEL	Dataset 1				Dataset 2				Dataset 3			
	P1	P2	P3	T	P1	P2	P3	T	P1	P2	P3	T
1 – 4												
2 – 5												
3 – 6												
4 – 7												
5 – 8												
6 – 9												
7 – 10												
8 – 11												
9 – 12												
10 – 13												
11 – 14												
12 – 15												
13 – 16												
14 – 17												
15 – 18												
16 – 19												
17 – 20												
18 – 21												
19 – 22												
20 – 23												
21 - 24												

 Training Set
 Validation Set
 Testing Set

Tabel 5. 4 Data *Training* dan *Testing* Untuk Model SVR (Lanjutan)

PERIODE/VARIABEL	Dataset 4				Dataset 5				Dataset 6			
	P1	P2	P3	T	P1	P2	P3	T	P1	P2	P3	T
1 – 4												
2 – 5												
3 – 6												
4 – 7												
5 – 8												
6 – 9												
7 – 10												
8 – 11												
9 – 12												
10 – 13												
11 – 14												
12 – 15												
13 – 16												
14 – 17												
15 – 18												
16 – 19												
17 – 20												
18 – 21												
19 – 22												
20 – 23												
21 - 24												

 Training Set
 Validation Set
 Testing Set

- **Uji Signifikansi Pengaruh Skenario Berbagai *Input* Peramalan**

Seperti pada model peramalan sebelumnya, pada tahap ini, berbagai skenario *input* peramalan pada **Sub-sub-bab 4.4.3** juga akan diuji pengaruhnya terhadap *forecasting error* (MSE) dan *forecasting robustness* dari model peramalan

SVR baik di tahap *training* maupun *testing*. Hasil dari masing-masing skenario tersebut adalah sebagai berikut.

1. Hasil uji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja model peramalan SVR

Pola permintaan hasil klasifikasi Boylan adalah yang pertama dievaluasi pengaruhnya terhadap kinerja metode peramalan SVR. Data yang digunakan adalah MSE *training* dan *testing* dari metode peramalan SVR yang visualisasinya ditunjukkan pada **Lampiran 61**. Berdasarkan **Lampiran 61**, diketahui bahwa sebagian nilai rata-rata MSE *testing* dari metode peramalan SVR lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan bahwa pada pola permintaan *erratic* dan *lumpy* tidak terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan SVR ditunjukkan pada **Lampiran 62**. Berdasarkan **Lampiran 62**, asumsi homogenitas varian untuk Anova satu arah tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada skenario ini, pola permintaan hasil klasifikasi Boylan akan diuji pengaruhnya terhadap metode peramalan SVR dengan uji Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja metode peramalan SVR dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 63**. Berdasarkan **Lampiran 63**, diketahui bahwa klasifikasi Boylan memiliki pengaruh yang cukup signifikan terhadap kinerja metode peramalan SVR baik pada tahap *training* maupun *testing* (karena ada perbedaan signifikan pada lebih dari satu pola permintaan).

2. Hasil uji pengaruh perbedaan *family part* terhadap kinerja model peramalan SVR

Evaluasi yang kedua adalah uji pengaruh perbedaan *family part* terhadap kinerja metode peramalan SVR. Visualisasi data MSE *training* dan *testing* dari metode peramalan SVR pada skenario ini ditunjukkan pada **Lampiran 64**. Berdasarkan **Lampiran 64**, diketahui bahwa sebagian nilai rata-rata MSE *testing* dari metode peramalan SVR lebih kecil dibandingkan MSE *training*. Fenomena

tersebut menunjukkan bahwa pada *family part exterior* dan *suspension* tidak terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan SVR ditunjukkan pada **Lampiran 65**. Berdasarkan **Lampiran 65**, asumsi homogenitas varian untuk Anova satu arah tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada skenario ini, perbedaan *family part* akan diuji pengaruhnya terhadap metode peramalan SVR dengan uji Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan *family part* terhadap kinerja metode peramalan SVR dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 66**. Berdasarkan **Lampiran 66**, diketahui bahwa perbedaan *family part* memiliki pengaruh yang kurang signifikan terhadap kinerja metode peramalan SVR (karena tidak ada perbedaan signifikan pada kinerja peramalan antar *family part* di tahap *testing* ($p\text{-value} > 0.05$)).

3. Hasil uji pengaruh perubahan-perubahan pada *input* data permintaan yang sama terhadap kinerja model peramalan SVR

Evaluasi yang ketiga adalah uji pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap kinerja metode peramalan SVR. Visualisasi data MSE *training* dan *testing* dari metode peramalan SVR pada skenario ini ditunjukkan pada **Lampiran 67**. Berdasarkan **Lampiran 67**, diketahui bahwa sebagian besar nilai rata-rata MSE *testing* dari metode peramalan SVR lebih besar dibandingkan MSE *training*. Fenomena tersebut menunjukkan bahwa pada semua perlakuan terhadap *input* peramalan (kecuali perlakuan *differencing*) terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan SVR ditunjukkan pada **Lampiran 68**. Berdasarkan **Lampiran 68**, asumsi homogenitas varian untuk Anova satu arah tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada skenario ini, perubahan-perubahan akibat eksperimentasi pada input peramalan akan diuji pengaruhnya terhadap metode peramalan SVR dengan uji Kruskal-Wallis.

Hasil uji signifikansi pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap kinerja metode peramalan SVR dengan

uji Kruskal-Wallis ditunjukkan pada **Lampiran 69**. Berdasarkan **Lampiran 69**, diketahui bahwa perubahan-perubahan akibat eksperimentasi pada *input* peramalan yang sama tidak memiliki pengaruh yang signifikan terhadap kinerja metode peramalan SVR (karena perbedaan hanya terjadi pada satu perlakuan *input* peramalan yaitu transformasi Box-Cox).

4. Hasil uji pengaruh perbedaan pengelompokkan *part* hasil K-Medoids *Clustering* terhadap kinerja model peramalan SVR

Evaluasi yang keempat adalah uji pengaruh pengelompokan pola permintaan hasil klaster K-Medoids terhadap kinerja metode peramalan SVR. Visualisasi data MSE *training* dan *testing* dari metode peramalan SVR pada skenario ini ditunjukkan pada **Lampiran 70**. Berdasarkan **Lampiran 70**, diketahui bahwa sebagian nilai rata-rata MSE *testing* dari metode peramalan SVR lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan bahwa pada klaster 1, 5, 6, 8, dan 9 tidak terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan SVR ditunjukkan pada **Lampiran 71**. Berdasarkan **Lampiran 71**, asumsi homogenitas varian untuk Anova satu arah tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada skenario ini, pengelompokan pola permintaan hasil klaster K-Medoids akan diuji pengaruhnya terhadap metode peramalan SVR dengan uji Kruskal-Wallis.

Hasil uji signifikansi pengaruh klaster K-Medoids terhadap kinerja metode peramalan SVR dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 72**. Berdasarkan **Lampiran 72**, diketahui bahwa klaster K-Medoids memiliki pengaruh yang kurang signifikan terhadap kinerja metode peramalan SVR (karena perbedaan hanya terjadi pada kurang dari atau sama dengan empat klaster) sehingga kurang berpengaruh dibandingkan pengelompokan pola permintaan hasil Klasifikasi Boylan.

5.3 Pengembangan dan Validasi Model Peramalan ANN

Seperti yang dijelaskan pada **Sub-bab 3.4**, pada model peramalan ANN terdapat *hyperparameter* yang perlu disesuaikan dengan teknik *grid search*. Proses

penyesuaian *hyperparameter* (*hyperparameter tuning*) tersebut akan dijelaskan pada bagian struktur model. Kemudian pada bagian validasi model akan dijelaskan komposisi data *training* dan *testing* yang digunakan untuk mengembangkan model dan hasil uji signifikansi pengaruh berbagai skenario dari **Sub-sub-bab 4.4.3** terhadap kinerja model peramalan ANN.

5.3.1 Struktur Model

Seperti yang dijelaskan pada **Sub-bab 3.4**, *hyperparameter* dari model peramalan ANN yang perlu disesuaikan dengan teknik *grid search* di antaranya adalah: jumlah *hidden layer*, jumlah *neuron*, *learning rate* (η), *epoch*, dan *threshold*. Masing-masing *hyperparameter* tersebut memiliki *grid space* atau himpunan nilai yang akan dicoba pada teknik *grid search*.

Selain itu, terdapat beberapa prosedur untuk mencegah *overfitting* pada model ANN, yaitu dengan menambahkan *dropout layer* dan regularisasi. *Dropout layer* terletak di antara *hidden layer* dan berfungsi untuk menyeleksi (dalam persentase) sejumlah neuron yang akan diteruskan ke *layer* selanjutnya secara random. Lalu, regularisasi memiliki parameter lambda (λ), yaitu parameter penalti yang akan ditambahkan pada *error* peramalan. Penjelasan struktur model peramalan ANN yang terdiri dari: *input layer*, *hidden layer*, *output layer*, *dropout layer*, regularisasi, teknik inisialisasi parameter, dan fungsi aktivasi, serta nilai dari masing-masing *hyperparameter* dapat dilihat pada **Tabel 5. 5** berikut.

Tabel 5. 5 Struktur Model Peramalan ANN dengan 8748 kombinasi *hyperparameter*, *dropout*, dan regularisasi

η	: Dengan <i>grid space</i> = {0.0001, 0.001, 0.01} (3 nilai)
<i>epoch</i>	: Dengan nilai 100 (<i>default</i> sistem)
<i>threshold</i>	: Dengan nilai 0 (<i>default</i> sistem)
<i>Input Layer</i>	: Memiliki jumlah neuron = {16, 32, 64} (3 nilai), teknik inisialisasi parameter Glorot <i>uniform</i> , regularisasi (λ) = {0.0001, 0.001, 0.01} (3 nilai), dan fungsi aktivasi ReLU
<i>Hidden Layer 1</i>	: Memiliki jumlah neuron = {16, 32, 64} (3 nilai) dan fungsi aktivasi ReLU

<i>Dropout</i> : <i>Layer 1</i>	Dengan persentase $dropout = \{0.166, 0.232, \dots, 0.430, 0.496\}$ (6 nilai)
<i>Hidden</i> : <i>Layer 2</i>	Memiliki jumlah neuron = {16, 32, 64} (3 nilai) dan fungsi aktivasi ReLU
<i>Dropout</i> : <i>Layer 2</i>	Dengan persentase $dropout = \{0.166, 0.232, \dots, 0.430, 0.496\}$ (6 nilai)
<i>Output</i> : <i>Layer</i>	Memiliki jumlah neuron 1 dan fungsi aktivasi ReLU

5.3.2 Validasi Model

Pada tahap ini akan dijelaskan menjadi dua bagian, yaitu tahap pengembangan data *training* dan *testing* dan uji signifikansi pengaruh skenario *input* peramalan.

- **Pengembangan Data *Training* dan *Testing***

Data *training* dan *testing* yang digunakan untuk mengembangkan model peramalan ANN sama seperti yang dijelaskan pada **Sub-sub-bab 5.2.2** untuk model peramalan SVR, yaitu bentuk *predictor target* dari berbagai skenario yang dijelaskan pada **Sub-sub-bab 4.4.3**.

- **Uji Signifikansi Pengaruh Skenario Berbagai *Input* Peramalan**

Seperti pada model peramalan sebelumnya, pada tahap ini, berbagai skenario *input* peramalan pada **Sub-sub-bab 4.4.3** juga akan diuji pengaruhnya terhadap *forecasting error* (MSE) dan *forecasting robustness* dari model peramalan ANN baik di tahap *training* maupun *testing*. Hasil dari masing-masing skenario tersebut adalah sebagai berikut.

1. **Hasil uji pengaruh perbedaan pola permintaan hasil klasifikasi**

Boylan terhadap kinerja model peramalan ANN

Evaluasi pertama yang dilakukan adalah uji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan. Data yang digunakan adalah MSE *training* dan *testing* dari metode peramalan ANN. Visualisasi data tersebut ditunjukkan pada **Lampiran 72**. Berdasarkan **Lampiran 72**, diketahui bahwa sebagian besar nilai rata-rata MSE *testing* dari metode peramalan ANN lebih kecil dibandingkan MSE

training. Fenomena tersebut menunjukkan bahwa pada pola permintaan *erratic*, *intermittent*, dan *lumpy* tidak terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan ANN ditunjukkan pada **Lampiran 74**. Berdasarkan **Lampiran 74**, diketahui bahwa asumsi homogenitas varian pada tahap *training* terpenuhi ($p\text{-value} > 0.05$), dan pada tahap *testing* tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada tahap selanjutnya data MSE *testing* dari metode peramalan ANN dalam skenario ini akan diuji menggunakan Kruskal-Wallis sedangkan MSE *training* akan diuji terhadap asumsi ketiga Anova yaitu normalitas residual.

Hasil uji asumsi normalitas residual Anova satu arah pada MSE *training* dari metode peramalan ANN dalam skenario ditunjukkan pada **Lampiran 75**. Berdasarkan **Lampiran 75**, diketahui bahwa asumsi normalitas residual pada MSE *training* dari metode peramalan ANN tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga data MSE *training* dari metode peramalan ANN dalam skenario ini juga akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja metode peramalan ANN dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 76**. Berdasarkan **Lampiran 76**, diketahui bahwa klasifikasi Boylan memiliki pengaruh yang kurang signifikan terhadap kinerja metode peramalan ANN (karena perbedaan hanya terjadi pada satu pola permintaan yaitu *intermittent*).

2. Hasil uji pengaruh perbedaan *family part* terhadap kinerja model peramalan ANN

Evaluasi yang kedua adalah uji pengaruh perbedaan *family part* terhadap kinerja metode peramalan ANN. Visualisasi data MSE *training* dan *testing* dari metode peramalan ANN pada skenario ini ditunjukkan pada **Lampiran 77**. Berdasarkan **Lampiran 77**, diketahui bahwa sebagian nilai rata-rata MSE *testing* dari metode peramalan ANN lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan bahwa pada *family part exterior* dan *suspension* tidak terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan ANN ditunjukkan pada **Lampiran 78**. Berdasarkan **Lampiran 78**, asumsi homogenitas varian untuk Anova satu arah tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada skenario ini, perbedaan *family part* akan diuji pengaruhnya terhadap metode peramalan ANN dengan uji Kruskal-Wallis.

Hasil uji signifikansi pengaruh perbedaan *family part* terhadap kinerja metode peramalan ANN dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 79**. Berdasarkan **Lampiran 79**, diketahui bahwa pada tahap *training*, *family part* memberikan pengaruh yang cukup signifikan terhadap kinerja peramalan Metode ANN (karena ada perbedaan signifikan pada lebih dari satu *family part* di tahap *training*).

3. Hasil uji pengaruh perubahan-perubahan pada *input* data permintaan yang sama terhadap kinerja model peramalan ANN

Evaluasi yang ketiga adalah uji pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap kinerja metode peramalan ANN. Visualisasi data MSE *training* dan *testing* dari metode peramalan ANN pada skenario ini ditunjukkan pada **Lampiran 80**. Berdasarkan **Lampiran 80**, diketahui bahwa sebagian nilai rata-rata MSE *testing* dari metode peramalan ANN lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan bahwa pada perlakuan *input* peramalan transformasi Box-Cox dan *differencing* tidak *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan ANN ditunjukkan pada **Lampiran 81**. Berdasarkan **Lampiran 81**, diketahui bahwa asumsi homogenitas varian pada tahap *testing* terpenuhi ($p\text{-value} > 0.05$), dan pada tahap *training* tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada tahap selanjutnya data MSE *training* dari metode peramalan ANN dalam skenario ini akan diuji menggunakan Kruskal-Wallis sedangkan MSE *testing* akan diuji terhadap asumsi ketiga Anova yaitu normalitas residual.

Hasil uji asumsi normalitas residual Anova satu arah pada MSE *testing* dari metode peramalan ANN dalam skenario ini ditunjukkan pada **Lampiran 82**. Berdasarkan **Lampiran 82**, diketahui bahwa asumsi normalitas residual dari MSE

testing pada skenario ini terpenuhi ($p\text{-value} > 0.05$). Oleh karena itu pada tahap selanjutnya, pengaruh eksperimentasi terhadap MSE *testing* dari metode peramalan ANN akan diuji dengan Anova satu arah.

Hasil uji signifikansi pengaruh perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama terhadap MSE *training* dari metode peramalan ANN dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 83**. Hasil uji signifikansi pengaruh perubahan-perubahan akibat eksperimentasi pada *input* peramalan yang sama terhadap MSE *testing* dari metode peramalan ANN dengan Anova satu arah ditunjukkan pada **Lampiran 84**. Berdasarkan kedua uji di **Lampiran 83 dan 84**, diketahui bahwa perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama memiliki pengaruh yang kurang signifikan terhadap kinerja metode peramalan ANN (karena perbedaan hanya terjadi pada satu perlakuan *input* peramalan yaitu transformasi Box-Cox).

4. Hasil uji pengaruh perbedaan pengelompokan *part* hasil K-Medoids *Clustering* terhadap kinerja model peramalan ANN

Evaluasi yang keempat adalah uji pengaruh pengelompokan pola permintaan hasil klaster K-Medoids terhadap kinerja metode peramalan ANN. Visualisasi data MSE *training* dan *testing* dari metode peramalan ANN pada skenario ini ditunjukkan pada **Lampiran 85**. Berdasarkan **Lampiran 85**, diketahui bahwa sebagian besar nilai rata-rata MSE *testing* dari metode peramalan ANN lebih kecil dibandingkan MSE *training*. Fenomena tersebut menunjukkan bahwa pada klaster 1, 3, 4, 5, 6, 8, dan 9 tidak terjadi *overfitting*.

Selanjutnya, hasil uji asumsi Anova satu arah pada skenario ini dengan metode peramalan ANN ditunjukkan pada **Lampiran 86**. Berdasarkan **Lampiran 86**, diketahui bahwa asumsi homogenitas varian pada tahap *testing* terpenuhi ($p\text{-value} > 0.05$) dan pada tahap *training* tidak terpenuhi ($p\text{-value} \leq 0.05$). Oleh karena itu, pada tahap selanjutnya data MSE *training* dari metode peramalan ANN dalam skenario ini akan diuji menggunakan Kruskal-Wallis sedangkan MSE *testing* akan diuji terhadap asumsi ketiga Anova yaitu normalitas residual.

Hasil uji asumsi normalitas residual Anova satu arah pada MSE *testing* dari metode peramalan ANN dalam skenario ini ditunjukkan pada **Lampiran 87**.

Berdasarkan **Lampiran 87**, diketahui bahwa asumsi normalitas residual pada MSE *training* dari metode peramalan ANN tidak terpenuhi ($p\text{-value} \leq 0.05$) sehingga data MSE *training* dari metode peramalan ANN dalam skenario ini juga akan diuji menggunakan Kruskal-Wallis.

Hasil uji signifikansi pengaruh klaster K-Medoids terhadap kinerja metode peramalan ANN dengan uji Kruskal-Wallis ditunjukkan pada **Lampiran 88**. Berdasarkan **Lampiran 88**, diketahui bahwa baik pada tahap *training* maupun *testing*, klaster K-Medoids memberikan pengaruh yang signifikan terhadap kinerja peramalan Metode ANN karena ada perbedaan signifikan pada lebih dari empat klaster K-Medoids (melebihi klasifikasi Boylan) sehingga ada cukup bukti yang menyatakan bahwa pola permintaan *spare part* di PT. X bisa dibedakan menjadi 9 klaster pola permintaan pada metode peramalan ANN.

5.4 Implementasi Peramalan Multi-Periode dan Perhitungan Interval Peramalan

Selanjutnya adalah tahap pemilihan model peramalan terbaik berdasarkan tahap training dan testing sebelumnya dan implementasi peramalan multi-periode dan perhitungan interval peramalan menggunakan model terbaik.

5.4.1 Pemilihan Model Peramalan Terbaik

Pada tahap pengembangan model sebelumnya, telah dilakukan uji signifikansi pengaruh berbagai skenario peramalan pada **Sub-sub-bab 4.4.3** terhadap kinerja peramalan (dalam hal ini *forecasting error* dan *forecasting robustness*) dari model peramalan metode Croston, SBA, TSB, SVR, dan ANN.

Rekapitulasi perbandingan *forecasting error* (MSE) dari semua model peramalan pada semua skenario di tahap *training* adalah sebagai berikut.

Tabel 5. 6 Perbandingan *Training Forecasting Error* Semua Model Peramalan Pada Semua Skenario Peramalan

		MIN					MAX					
		Croston	SBA	TSB	SVR	ANN						
		MSE <i>Training</i>	MSE <i>Training</i>	MSE <i>Training</i>	MSE <i>Training</i>	MSE <i>Training</i>						
KLASIFIKASI BOYLAN	smooth_exterior	4468.88	4022.95	3565.37	1252.75	2085.48						
	intermittent_exterior	116078.00	128313.18	78914.28	65983.05	83123.12						
	erratic_exterior	51334.34	39521.31	57199.88	28238.58	17550.51						
	lumpy_exterior	8994.78	8132.36	7467.60	5138.53	7842.39						
FAMILY PART	lumpy_exterior	8994.78	8132.36	7467.60	5138.53	7842.39						
	lumpy_interior	20685.71	21007.08	26604.21	14516.77	14929.65						
	lumpy_engine	442.39	648.19	307.61	4.77	4.94						
	lumpy_suspension	986214.58	1022150.60	558904.94	598709.90	597250.64						
EKSPERIMENTASI INPUT	kontrol_intermittent_engine	131.86	129.19	119.82	59.61	157.38						
	seasadj_intermittent_engine	0.04	0.06	0.04	31.87	107.14						
	boxcox_intermittent_engine	0.17	0.17	0.16	16.71	35.95						
	diff_intermittent_engine	0.11	0.10	0.11	204.83	237.69						

Tabel 5. 6 Perbandingan *Training Forecasting Error* Semua Model Peramalan Pada Semua Skenario Peramalan (Lanjutan)

		MIN					MAX				
		Croston	SBA	TSB	SVR	ANN	MSE	MSE	MSE	MSE	MSE
		Training	Training	Training	Training	Training	Training	Training	Training	Training	Training
KLASTER K-MEDOIDS	k1_suspension	986214.58	1022150.60	558904.94	598709.90	597250.64					
	k2_suspension	11920.73	11914.50	9234.75	7612.56	7612.64					
	k3_suspension	92765.26	102104.71	90920.50	51560.45	219628.80					
	k4_suspension	626930.52	630313.30	143952.75	172686.30	183664.23					
	k5_suspension	208697.57	208697.57	214999.66	292609.13	257987.95					
	k6_suspension	105878.28	82087.81	109964.76	175285.24	96309.15					
	k7_suspension	111.04	110.67	110.54	3.37	9.50					
	k8_suspension	88816.58	93081.97	64625.29	65963.02	90316.17					
	k9_suspension	48951.23	49032.12	43562.11	43400.06	46953.22					
Rata-Rata MSE Keseluruhan		160363.40	163407.18	94134.62	101291.71	106233.31					

Berdasarkan **Tabel 5. 6** diketahui bahwa model peramalan yang cenderung memiliki *forecasting error* yang kecil pada seluruh skenario atau kinerjanya lebih baik dibandingkan dengan yang lain di tahap *training* adalah metode TSB, SVR, dan ANN.

Selanjutnya, rekapitulasi perbandingan *forecasting error* (MSE) dari semua model peramalan pada semua skenario di tahap *testing* adalah sebagai berikut.

Tabel 5. 7 Perbandingan *Testing Forecasting Error* Semua Model Peramalan Pada Semua Skenario Peramalan

		MIN					MAX					
		Croston	SBA	TSB	SVR	ANN						
		MSE <i>Testing</i>	MSE <i>Testing</i>	MSE <i>Testing</i>	MSE <i>Testing</i>	MSE <i>Testing</i>						
KLASIFIKASI BOYLAN	smooth_exterior	5478.68	5468.38	6758.11	3621.94	5007.22						
	intermittent_exterior	105333.51	89737.63	88531.67	90289.82	80304.88						
	erratic_exterior	6889.00	1722.25	212.20	0.00	0.00						
	lumpy_exterior	28135.00	25634.01	28066.23	2271.77	2413.76						
FAMILY PART	lumpy_exterior	28135.00	25634.01	28066.23	2271.77	2413.76						
	lumpy_interior	161352.29	101037.68	151615.05	144859.07	148475.62						
	lumpy_engine	16269.92	10953.61	16407.57	17634.87	16451.96						
	lumpy_suspension	1601438.22	1578626.96	1857213.79	6912.75	7125.14						
EKSPERIMENTASI INPUT	kontrol_intermittent_engine	191.78	149.89	150.21	92.49	161.52						
	seasadj_intermittent_engine	0.06	0.06	0.07	222.52	274.29						
	boxcox_intermittent_engine	0.26	0.23	0.21	26.02	21.49						
	diff_intermittent_engine	0.06	0.06	0.06	169.83	151.87						

Tabel 5. 7 Perbandingan *Testing Forecasting Error* Semua Model Peramalan Pada Semua Skenario Peramalan (Lanjutan)

	KLASTER K-MEDOIDS	MIN					MAX				
		Croston		SBA		TSB		SVR		ANN	
		MSE <i>Testing</i>									
k1_suspension	1601438.22	1578626.96	1857213.79	6912.75	7125.14						
k2_suspension	48808.95	49132.44	48881.58	44781.99	82502.82						
k3_suspension	245238.99	232825.59	243045.96	82324.89	69923.53						
k4_suspension	316052.51	111508.48	318543.63	173211.20	169113.17						
k5_suspension	10620.00	10620.00	6633.93	15.43	0.12						
k6_suspension	6724.00	1681.00	683.59	0.00	4.52						
k7_suspension	685.23	712.39	775.92	1273.30	1418.15						
k8_suspension	69772.26	71409.33	67839.26	6906.79	8.44						
k9_suspension	87144.22	77462.57	101660.97	4845.03	7397.40						
Rata-Rata MSE Keseluruhan	206652.77	189187.79	229633.33	28030.68	28585.47						

Berdasarkan **Tabel 5. 7** diketahui bahwa model peramalan yang cenderung memiliki *forecasting error* yang kecil pada seluruh skenario atau kinerjanya lebih baik dibandingkan dengan yang lain di tahap *testing* adalah metode SVR dan ANN.

Selanjutnya, rekapitulasi perbandingan apakah terjadi *overfitting* ($\text{MSE } \textit{Training} < \text{MSE } \textit{Testing}$) pada masing-masing model peramalan di semua skenario adalah sebagai berikut.

Tabel 5. 8 Perbandingan Apakah Terjadi *Overfitting* Pada Masing-Masing Model Peramalan di Semua Skenario

		Croston	SBA	TSB	SVR	ANN
		<i>Overfitting?</i>	<i>Overfitting?</i>	<i>Overfitting?</i>	<i>Overfitting?</i>	<i>Overfitting?</i>
KLASIFIKASI BOYLAN	smooth_exterior	Y	Y	Y	Y	Y
	intermittent_exterior	N	N	Y	Y	N
	erratic_exterior	N	N	N	N	N
	lumpy_exterior	Y	Y	Y	N	N
FAMILY PART	lumpy_exterior	Y	Y	Y	N	N
	lumpy_interior	Y	Y	Y	Y	Y
	lumpy_engine	Y	Y	Y	Y	Y
	lumpy_suspension	Y	Y	Y	N	N

Tabel 5. 8 Perbandingan Apakah Terjadi *Overfitting* Pada Masing-Masing Model Peramalan di Semua Skenario (Lanjutan)

		Croston	SBA	TSB	SVR	ANN
		<i>Overfitting</i>	<i>Overfitting</i>	<i>Overfitting</i>	<i>Overfitting</i>	<i>Overfitting</i>
EKSPERIMENTASI INPUT	kontrol_intermittent_engine	Y	Y	Y	Y	Y
	seasadj_intermittent_engine	Y	Y	Y	Y	Y
	boxcox_intermittent_engine	Y	Y	Y	Y	N
	diff_intermittent_engine	N	N	N	N	N
	k1_suspension	Y	Y	Y	N	N
	k2_suspension	Y	Y	Y	Y	Y
	k3_suspension	Y	Y	Y	Y	N
	k4_suspension	N	N	Y	Y	N
	k5_suspension	N	N	N	N	N
	k6_suspension	N	N	N	N	N
KLASTER K-MEDOIDS	k7_suspension	Y	Y	Y	Y	Y
	k8_suspension	N	N	Y	N	N
	k9_suspension	Y	Y	Y	N	N
Jumlah <i>Overfitting</i>		14	14	17	11	7

Berdasarkan **Tabel 5. 8** diketahui bahwa model peramalan yang paling jarang mengalami *overfitting* ($MSE\ training < MSE\ testing$) pada seluruh skenario atau kinerjanya lebih konsisten dibandingkan dengan yang lain adalah metode SVR dan ANN.

Selanjutnya, rekapitulasi hasil uji signifikansi pengaruh masing-masing skenario peramalan terhadap pada semua model peramalan dengan pendekatan Anova satu arah atau uji Kruskal-Wallis adalah sebagai berikut.

Tabel 5. 9 Rekapitulasi Hasil Uji Signifikansi Pengaruh Berbagai Skenario Peramalan Terhadap Semua Model Peramalan (1)

	Croston		SBA		TSB	
	One-Way Anova	Kruskal-Wallis	One-Way Anova	Kruskal-Wallis	One-Way Anova	Kruskal-Wallis
Klasifikasi Boylan	*	Signifikan	*	Signifikan	*	Signifikan
Family Part	*	Tidak Signifikan	*	Tidak Signifikan	*	Signifikan
Eksperimentasi Input	*	Signifikan	*	Signifikan	*	Signifikan
Klaster K-Medoids	*	Signifikan	*	Signifikan	*	Signifikan

Tabel 5. 9 Rekapitulasi Hasil Uji Signifikansi Pengaruh Berbagai Skenario Peramalan Terhadap Semua Model Peramalan (2)

	SVR		ANN	
	One-Way Anova	Kruskal-Wallis	One-Way Anova	Kruskal-Wallis
Klasifikasi Boylan	*	Signifikan	*	Tidak Signifikan
Family Part	*	Tidak Signifikan	*	Signifikan
Eksperimentasi Input	*	Tidak Signifikan	Tidak Signifikan	*
Klaster K-Medoids	*	Tidak Signifikan	*	Signifikan

Berdasarkan Tabel 5. 9 diketahui bahwa model peramalan yang lebih tidak sensitif terhadap berbagai skenario peramalan atau lebih *robust* terhadap perubahan pada

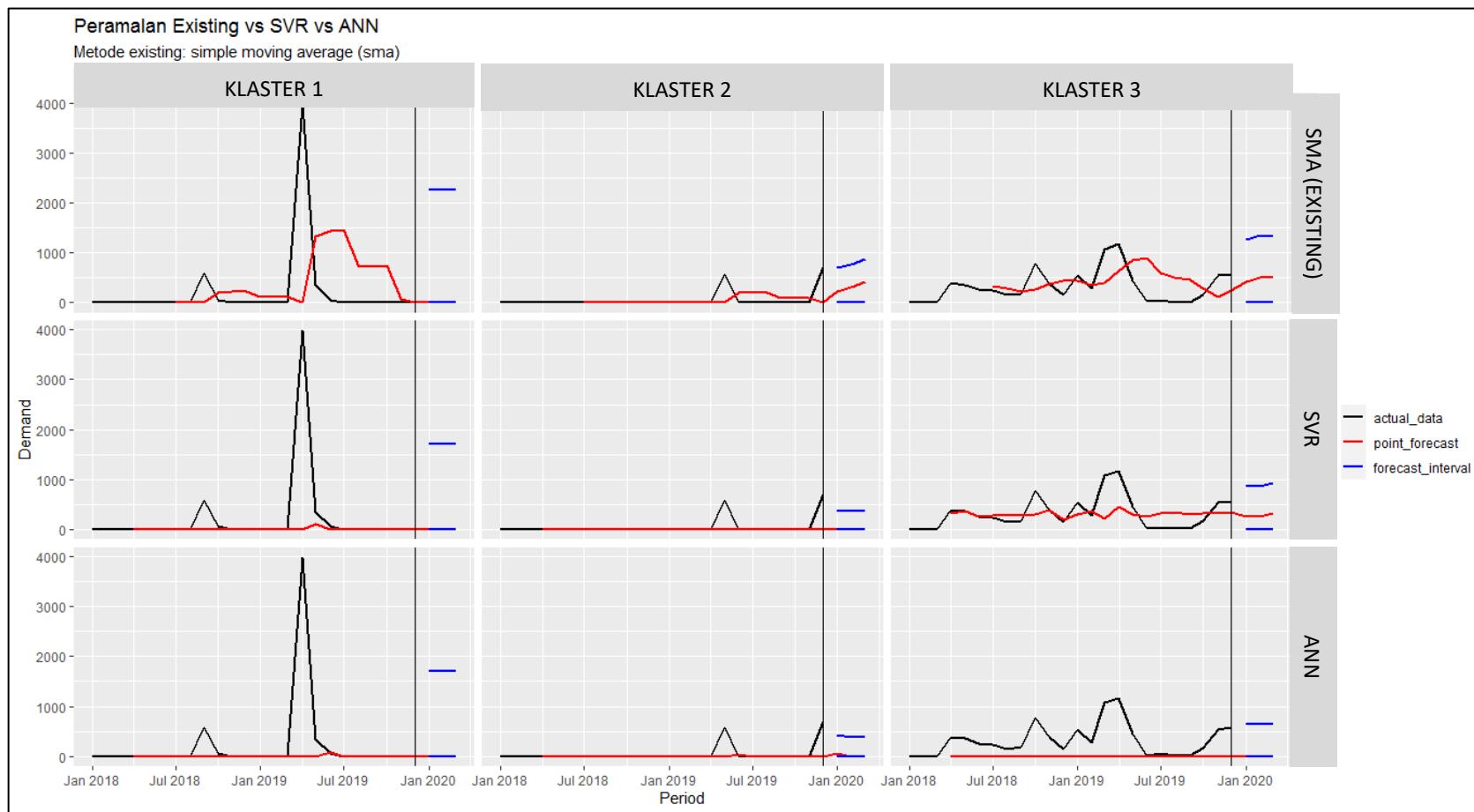
input peramalan dibandingkan dengan yang lain adalah model peramalan SVR dan ANN.

5.4.2 *Implementasi Peramalan Multi-Periode dan Perhitungan Interval Peramalan Dengan Model Terpilih*

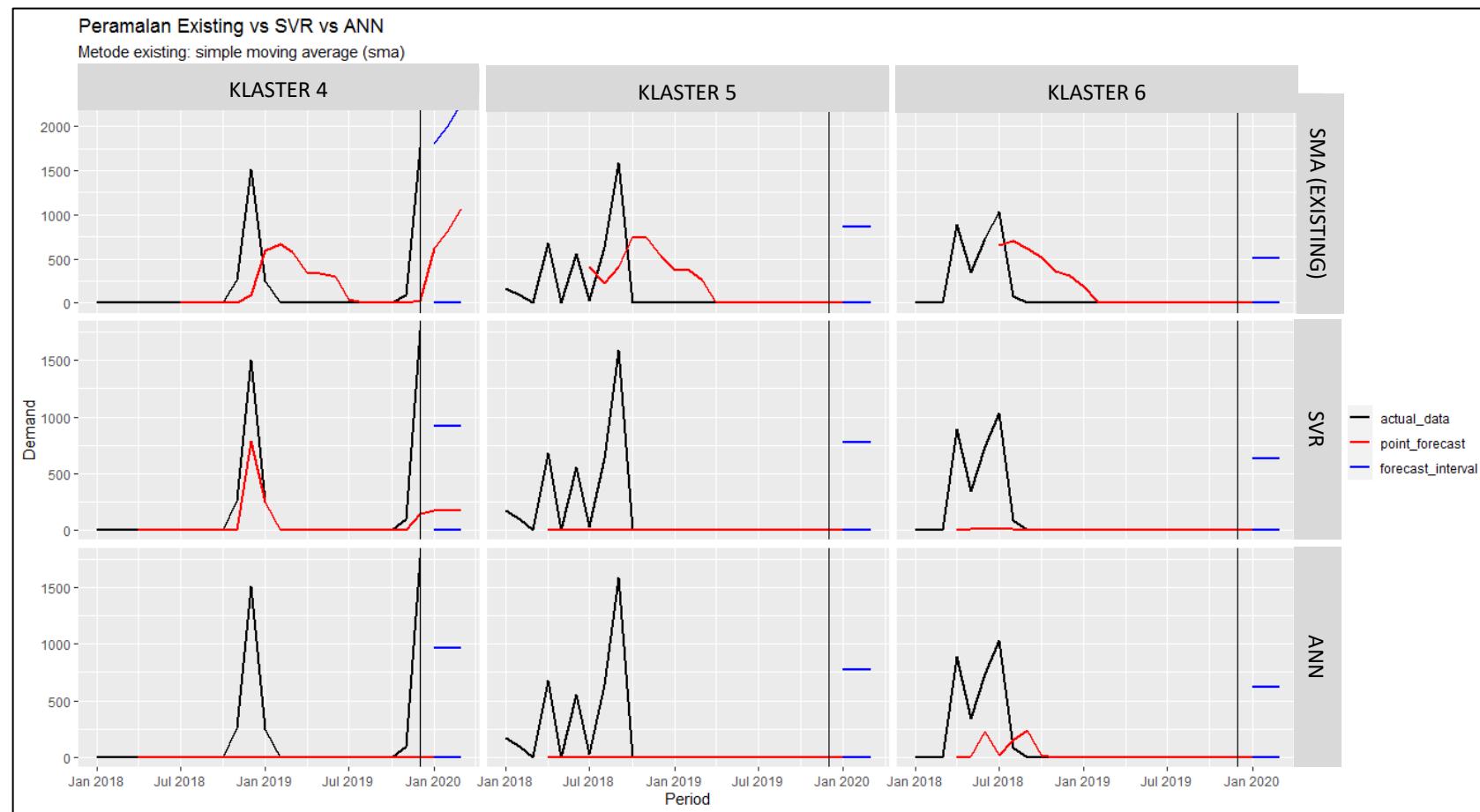
Model usulan terbaik, yaitu SVR dan ANN, selanjutnya akan digunakan untuk meramalkan permintaan multi-periode ke depan (dalam hal ini tiga periode ke depan) dan dibandingkan kinerjanya terhadap metode peramalan *existing* di PT. X yaitu *simple moving average* (SMA).

Peramalan akan diimplementasikan pada masing-masing satu permintaan *part* dari setiap grup (klaster dan pola permintaan hasil klasifikasi Boylan). Permintaan-permintaan *part* yang dipilih tersebut merupakan permintaan yang paling sulit diramalkan dari setiap grup (dalam hal ini adalah yang menghasilkan *forecasting error* paling besar pada metode peramalan *existing*). Oleh karena itu, perbaikan kinerja peramalan pada permintaan-permintaan *part* tersebut akan berpengaruh signifikan terhadap permintaan-permintaan *part* yang lainnya.

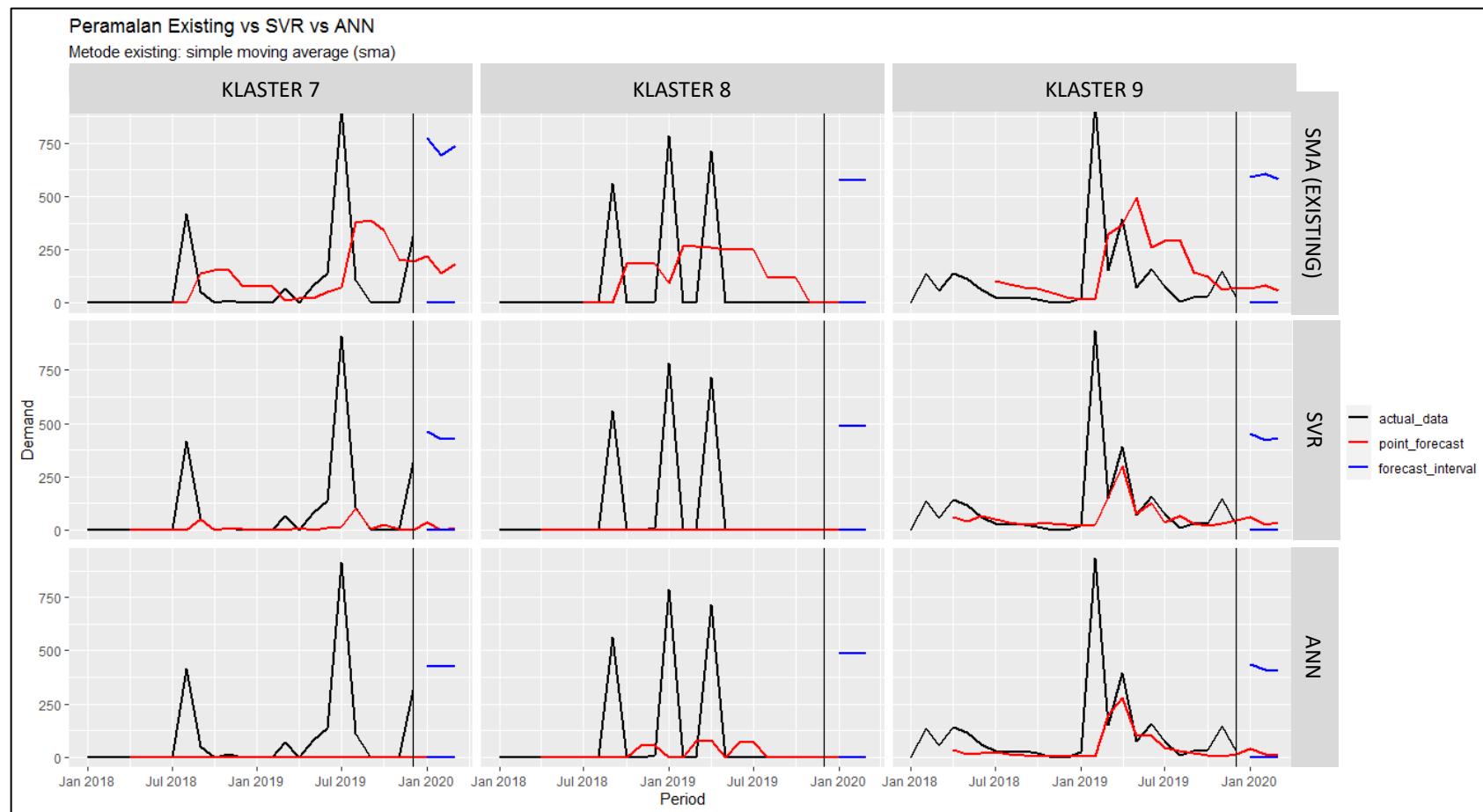
Selain itu, pada setiap peramalan akan dihitung interval (batas atas dan bawah) peramalannya, seperti yang dijelaskan pada **Sub-bab 3.5**, sehingga dapat membantu untuk proses pengambilan keputusan dalam praktiknya di operasional PT. X. Hasil peramalan akan ditampilkan pada **Gambar 5.2 - 5.5**, dan rekapitulasi kinerja peramalannya akan ditampilkan pada **Tabel 5.10** di bawah ini.



Gambar 5. 2 Peramalan Multi Periode (Periode = T+3) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode *Existing* (SMA)
Pada Permintaan *Part* di Klaster 1, 2, dan 3

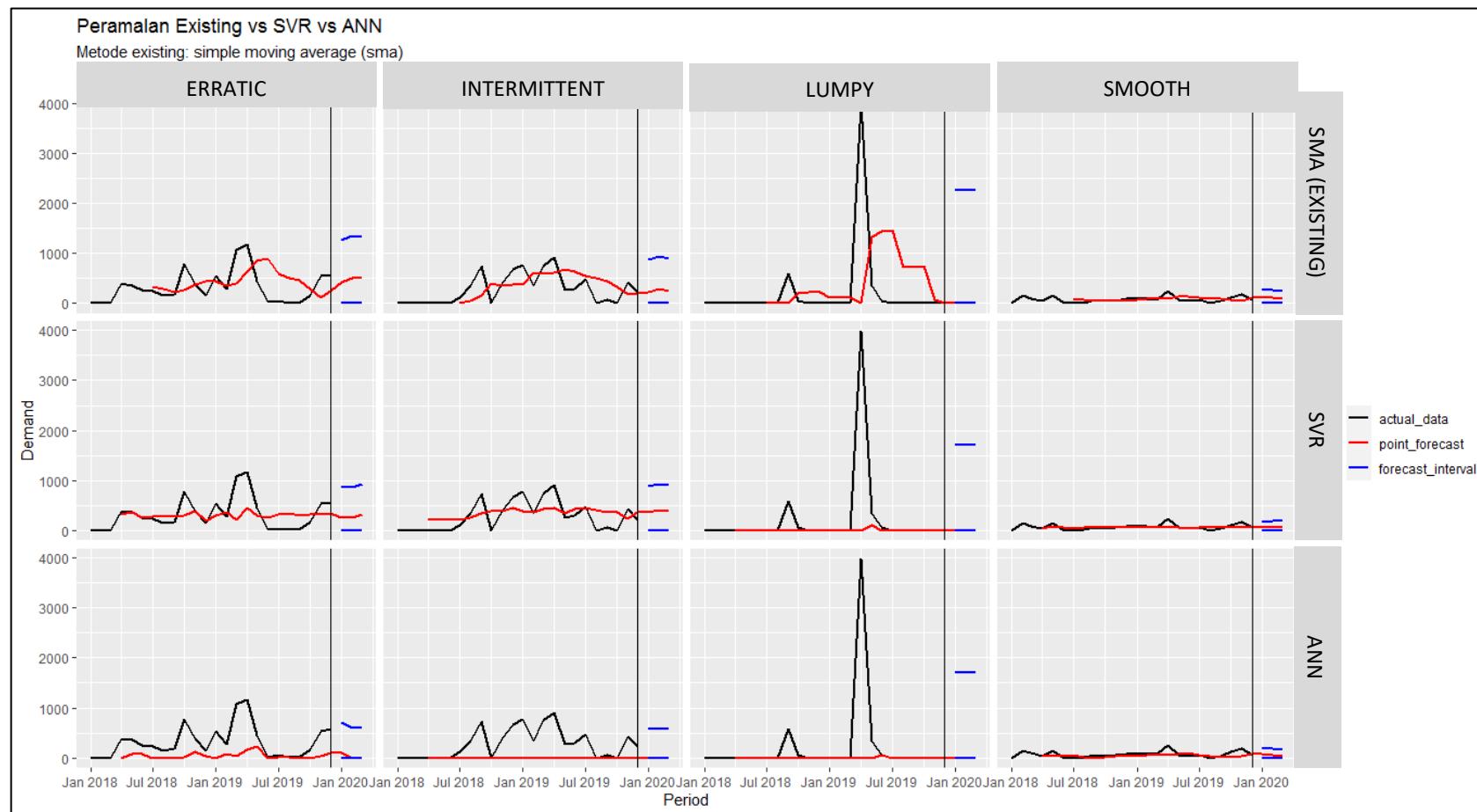


Gambar 5. 3 Peramalan Multi Periode (Periode = T+3) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode *Existing* (SMA)
Pada Permintaan *Part* di Klaster 4, 5, dan 6



Gambar 5. 4 Peramalan Multi Periode (Periode = T+3) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode *Existing* (SMA)

Pada Permintaan *Part* di Klaster 7, 8, dan 9



Gambar 5. 5 Peramalan Multi Periode (Periode = T+3) Dengan Metode Usulan Terbaik (SVR dan ANN) dan Metode *Existing* (SMA)
Pada Pola Permintaan *Part Smooth*, *Intermittent*, *Erratic*, dan *Lumpy*

Tabel 5. 10 Perbandingan *In-Sample Forecasting Error* Metode Usulan Terbaik (SVR dan ANN) Dengan Metode Existing (SMA)

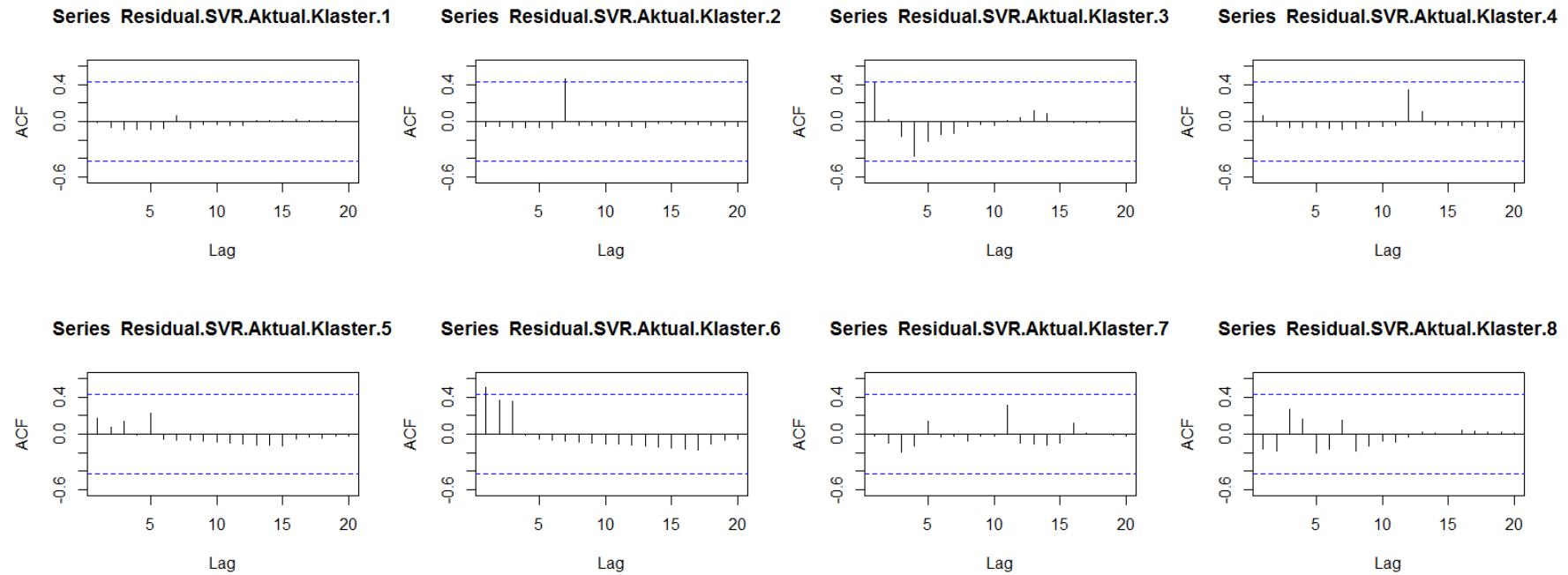
	Parameter <i>In-Sample Forecasting Error</i>											
	RMSE*			MAPE**			SMAPE			MASE		
Grup / Metode	SMA (Existing)	SVR	ANN	SMA (Existing)	SVR	ANN	SMA (Existing)	SVR	ANN	SMA (Existing)	SVR	ANN
Klaster 1	1127.62	880.04	882.08	14640.20%	81.10%	92.36%	93.57%	87.34%	91.63%	160.74%	58.80%	60.07%
Klaster 2	228.93	195.70	196.24	71.27%	60.59%	98.77%	87.81%	83.89%	98.93%	147.44%	75.50%	77.98%
Klaster 3	416.35	311.63	486.94	767.54%	354.22%	100.00%	44.07%	34.49%	100.00%	150.90%	98.13%	163.8%
Klaster 4	592.45	391.93	512.57	186.65%	70.36%	99.95%	92.78%	89.78%	99.96%	163.00%	62.13%	88.67%
Klaster 5	438.59	419.98	419.98	1292.93%	100.00%	99.99%	92.11%	100.00%	99.99%	110.97%	66.24%	66.25%
Klaster 6	282.76	344.82	328.07	397.55%	97.21%	89.06%	87.69%	94.79%	93.94%	121.16%	105.55%	106.2%
Klaster 7	274.72	229.04	233.84	229.91%	67.12%	100.00%	80.74%	85.00%	100.00%	140.04%	68.60%	73.49%
Klaster 8	285.58	261.92	254.06	557.28%	100.00%	202.83%	93.98%	100.00%	97.37%	120.78%	55.00%	61.10%
Klaster 9	262.42	203.55	210.77	417.74%	74.70%	70.43%	57.46%	33.54%	51.62%	110.48%	51.97%	59.47%
<i>Smooth Demand</i>	68.60	58.18	71.29	80.56%	44.25%	57.85%	35.85%	29.85%	46.95%	86.69%	60.66%	84.25%
<i>Intermittent Demand</i>	324.54	260.50	437.60	97.10%	767.19%	99.98%	48.45%	45.21%	99.97%	123.29%	95.44%	139.9%
<i>Erratic Demand</i>	416.35	311.63	438.93	767.54%	354.22%	87.41%	44.07%	34.49%	77.95%	150.90%	98.13%	143.2%
<i>Lumpy Demand</i>	1127.62	880.04	882.05	14640.20%	81.10%	88.49%	93.57%	87.34%	90.32%	160.74%	58.80%	59.77%
Total / Rata-Rata Error	449.73	365.30	411.88	2627%	173%	99%	73%	70%	88%	134%	73%	91%

* Nilai akar kuadrat dari MSE, yaitu Root MSE (RMSE), digunakan pada tabel agar angka yang ditampilkan tidak terlalu besar

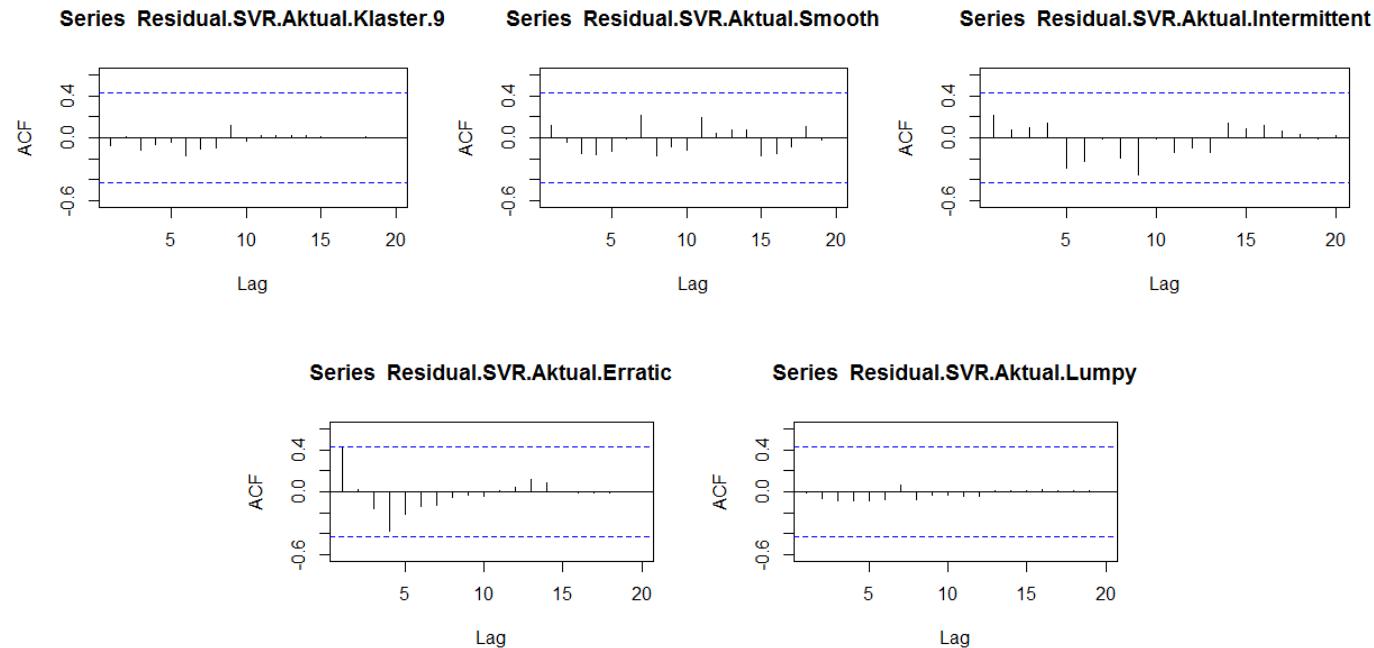
** Parameter MAPE yang digunakan mengabaikan permintaan aktual nol

Seperti yang ditunjukkan pada **Tabel 5. 10**, kinerja peramalan SVR, kemudian disusul oleh ANN, secara keseluruhan terbukti lebih baik dibandingkan metode *Existing* ketika diukur dengan berbagai parameter pengukuran *forecasting error*, yaitu *absolut error* RMSE dan *relative error*: MAPE, SMAPE, dan MASE. Apabila ditinjau dari parameter RMSE, maka ada perbaikan kinerja peramalan (atau pengurangan *forecasting error*) yang cukup besar, yaitu 18.8% oleh SVR dan 8.42% oleh ANN dari metode peramalan *existing*, walaupun belum memenuhi kriteria “*good forecast*” dari Lewis (1982), yaitu $\text{MAPE} < 20\%$.

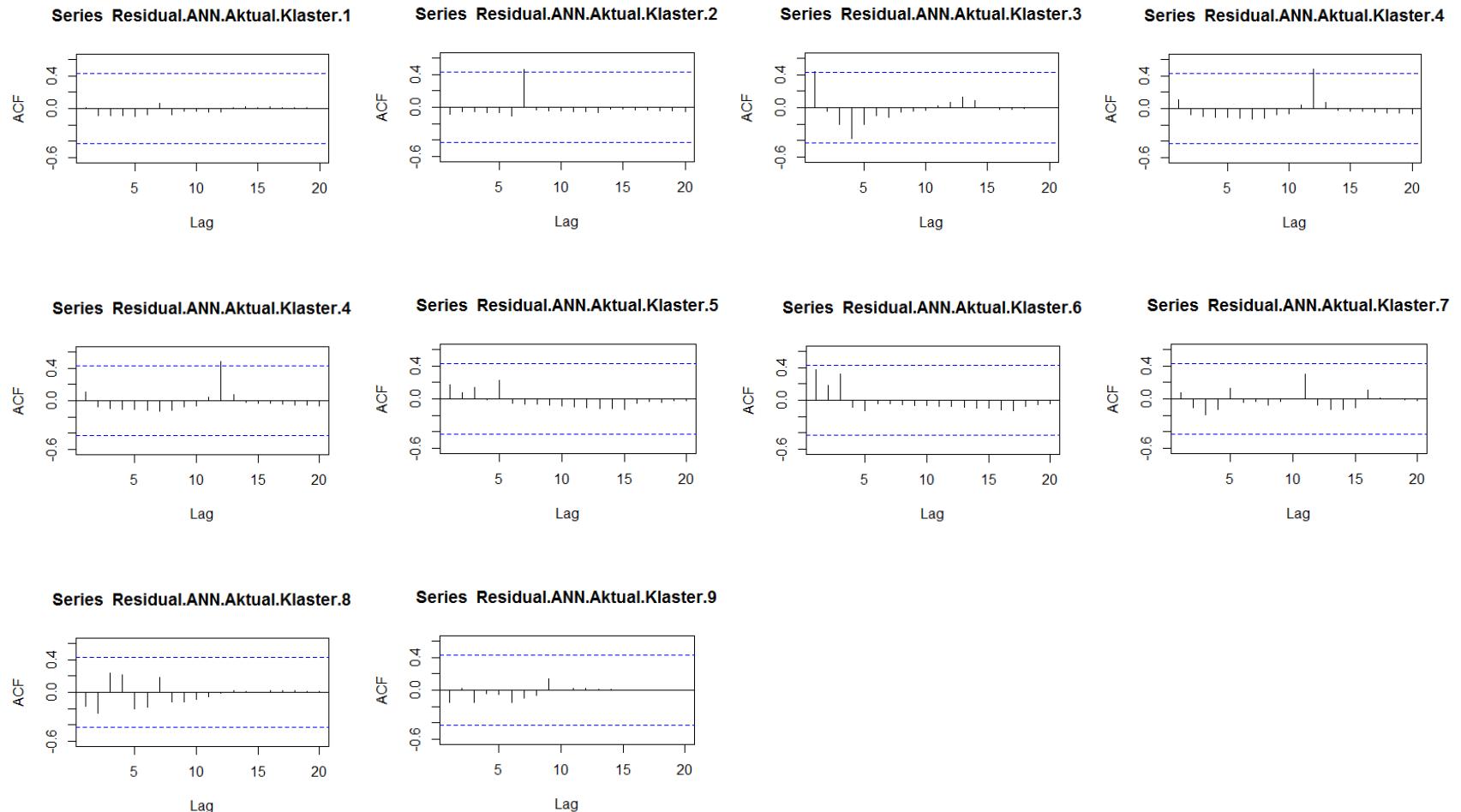
Namun, apabila meninjau dari Hyndman & Athanasopoulos (2018), kriteria “*good forecast*” juga bisa dievaluasi dari nilai residual peramalan, atau selisih nilai aktual permintaan dan peramalan, yang tidak saling berkorelasi (*non-auto correlated*). Hal tersebut menunjukkan bahwa tidak ada lagi pola dari *time-series* yang belum dijelaskan oleh nilai peramalannya, jika pun ada tetapi tidak signifikan. Signifikan atau tidak signifikannya residual peramalan bisa dilihat dari batas atas dan bawah plot ACF (*Autocorrelation Function*). **Gambar 5. 6 – 5. 7** adalah plot ACF dari peramalan metode SVR, dan ANN terhadap permintaan aktual semua klaster dan pola permintaan hasil klasifikasi Boylan.



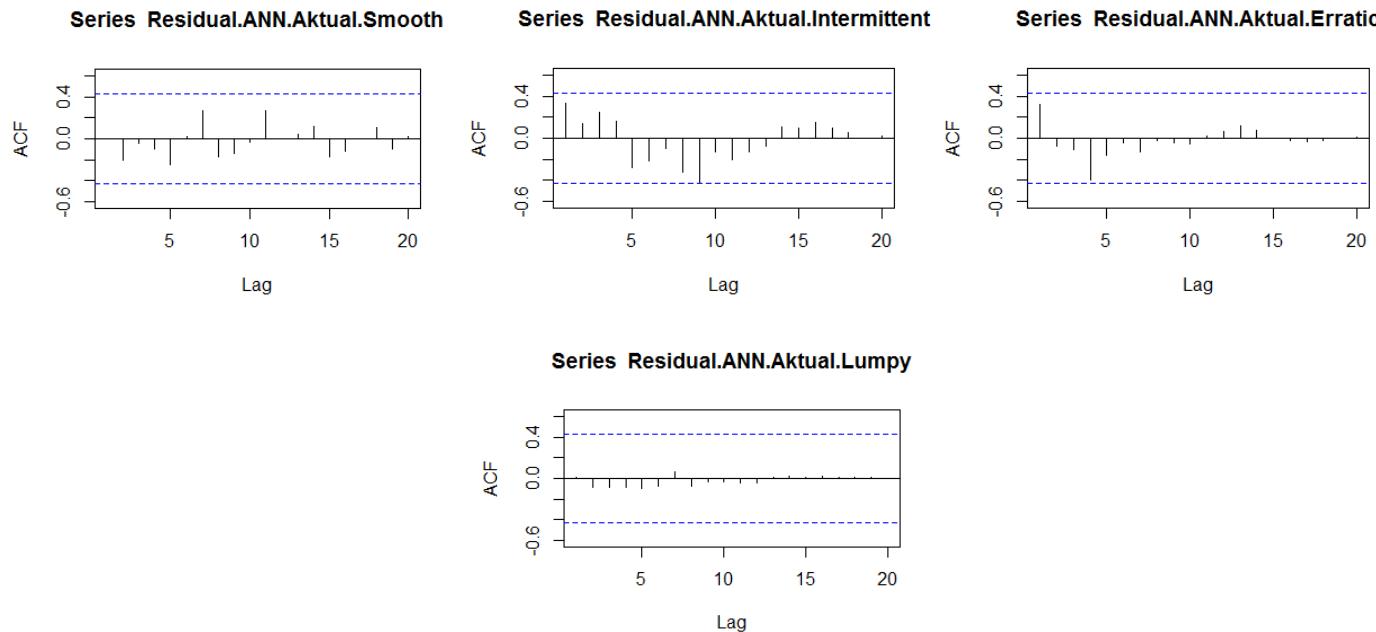
Gambar 5. 6 Plot ACF Dari Residual Peramalan SVR Terhadap Permintaan Aktual Semua Klaster Dan Pola Permintaan Boylan



Gambar 5. 6 Plot ACF Dari Residual Peramalan SVR Terhadap Permintaan Aktual Semua Klaster Dan Pola Permintaan Boylan
(Lanjutan)



Gambar 5. 7 Plot ACF Dari Residual Peramalan ANN Terhadap Permintaan Aktual Semua Klaster Dan Pola Permintaan Boylan



Gambar 5. 7 Plot ACF Dari Residual Peramalan ANN Terhadap Permintaan Aktual Semua Klaster Dan Pola Permintaan Boylan (Lanjutan)

Seperti yang ditunjukkan pada **Gambar 5. 6** dan **Gambar 5. 7**, sebagian besar residual hasil peramalan dari metode SVR dan ANN tidak saling berkorelasi secara signifikan (tidak melewati batas atas atau bawah plot ACF). Oleh karena itu, pada sebagian besar permintaan *spare part* di atas, metode SVR dan ANN menghasilkan peramalan yang memenuhi kriteria “*good forecast*” menurut (Hyndman & Athanasopoulos, 2018).

BAB 6

INTERPRETASI DAN

ANALISIS

BAB 6

INTERPRETASI DAN ANALISIS

Pada bab ini akan dijelaskan mengenai interpretasi dan analisis dari hasil pengelompokan pola permintaan *spare part* di PT. X, hasil perbandingan kinerja metode peramalan Croston, SBA, TSB, SVR, dan ANN, serta rencana implementasi sistem peramalan yang lebih akurat dan *robust* dalam operasional di PT. X.

6.1 Analysis Sistem Pengelompokan Pola Permintaan Spare Part di PT. X

Pengelompokan *spare part* berdasarkan pola permintaannya berfungsi sebagai pendukung dalam pengambilan keputusan terhadap hasil peramalan permintaan. Sebagaimana ditunjukkan pada **Sub-sub-bab 5.4.1**, perbedaan pada pola permintaan memiliki pengaruh yang signifikan terhadap kinerja suatu metode peramalan. Hal ini karena variabilitas dan frekuensi pola permintaan satu dengan yang lainnya berbeda-beda. Oleh karena itu, kompleksitas dan ketidakpastian peramalannya berbeda-beda sehingga keputusan yang diambil terhadap hasil peramalan tersebut bisa berbeda-beda tergantung pola permintaannya.

Dalam tugas akhir ini, terdapat dua pendekatan pengelompokan pola permintaan *part* di PT. X yang dibandingkan, yaitu pendekatan teoritis klasifikasi Boylan dan pendekatan berdasarkan kedekatan antar nilai permintaan menggunakan teknik klaster K-Medoids. Hasil klasifikasi Boylan menunjukkan bahwa mayoritas pola permintaan *part* di PT. X bisa dikategorikan sebagai “LUMPY” (54.76%) dan “INTERMITTENT” (37.1%) dan sisanya adalah “ERRATIC” (6.42%) dan “SMOOTH” (1.72%). Sedangkan hasil klaster K-Medoids menunjukkan bahwa pola permintaan *part* di PT. X bisa dibedakan menjadi sembilan kelompok yang proporsinya cukup merata seperti yang bisa dilihat pada **Sub-sub-bab 4.3.2**. Hal tersebut menunjukkan bahwa teknik klaster K-Medoids mampu menangkap lebih banyak perbedaan pada pola permintaan *part* di PT. X. Selain itu apabila jumlah klaster dari K-Medoids disamakan dengan jumlah klasifikasi Boylan yaitu empat pola permintaan, maka hasil klasternya kurang optimal, seperti yang ditunjukkan pada **Gambar 4. 3 dan 4. 5** bahwa jumlah

klaster empat memiliki nilai *silhouette* dan *sum of squared within* yang lebih buruk dari jumlah klaster sembilan.

Meskipun K-Medoids lebih efektif dalam membedakan pola permintaan *part* di PT. X, akan tetapi hasil pengelompokan dari K-Medoids belum bisa digunakan sebagai pendukung peramalan pada operasional di PT. X. Hal tersebut karena hasil dari K-Medoids berasal dari pembelajaran computer (*machine learning*) sehingga belum memiliki label dan kriteria pembeda yang bisa dengan jelas dipahami oleh pengguna. Oleh karena itu, kriteria pembedanya masih perlu didiskusikan agar bisa digunakan dalam operasional sehingga menyisakan ruang untuk pengembangan bagi penelitian selanjutnya. Maka dari itu, dalam tugas akhir ini pengelompokan dengan pendekatan teoritis klasifikasi Boylan bisa dipilih terlebih dahulu dalam rencana implementasi di operasional PT. X. Hal ini juga didukung dengan hasil pada **Sub-sub-bab 5.4.1** yang menunjukkan bahwa kedua pendekatan pengelompokan ini sama-sama memiliki pengaruh yang signifikan terhadap kinerja dari suatu metode peramalan.

Selain kedua pendekatan pengelompokan sebelumnya, dalam tugas akhir ini, *part* juga dibedakan berdasarkan *family part* nya, yaitu “EXTERIOR”, “INTERIOR”, “ENGINE”, dan “SUSPENSION”, yang masing-masing mewakili sumber kerusakan yang berbeda-beda. Namun, perbandingan pada **Sub-sub-bab 5.4.1** menunjukkan bahwa pada sebagian besar metode peramalan, perbedaan *family part* tidak memberikan pengaruh yang signifikan terhadap kinerja peramalan.

6.2 Analisis Perbandingan Kinerja Berbagai Model Peramalan terhadap Permintaan *Spare Part* di PT. X

Dalam tugas akhir ini, kriteria metode peramalan yang lebih baik ditentukan dengan dua parameter, yaitu *forecasting error* (yang mengukur keakurasaian peramalan) dan *forecasting robustness* (yang mengukur konsistensi akurasi peramalan). Parameter *forecasting error* yang paling baik untuk menyeleksi model peramalan adalah nilai *error absolut mean squared error* atau MSE. Hal tersebut berdasarkan Montgomery, et al. (2008) pada **Sub-bab 2.7**. Sedangkan *forecasting robustness* diukur dengan melakukan eksperimentasi pada *input* peramalan

kemudian melakukan uji signifikansi pengaruh eksperimentasi tersebut terhadap *forecasting error*. Kinerja model peramalan yang konsisten lebih tidak sensitif terhadap pengaruh eksperimentasi tersebut. Baik *forecasting error* dan *robustness* akan diukur pada tahap *training* (pengembangan model) dan *testing* (uji model). Model yang hanya memiliki kinerja yang baik di tahap *training* berarti mengalami *overfitting* dan tidak dipilih dalam tahap implementasi model untuk peramalan multi-periode.

Berdasarkan hasil pada **Sub-sub-bab 5.4.1** diketahui bahwa model peramalan yang memiliki kinerja keseluruhan yang lebih baik di antara yang lain adalah model peramalan dengan pendekatan *data mining*, yaitu SVR kemudian ANN. Oleh karena itu, metode SVR dan ANN akan diimplementasikan untuk peramalan multi periode. Implementasi ini digunakan sebagai simulasi dari operasional peramalan permintaan *spare part* di PT. X, yaitu meramalkan permintaan tiga periode ke depan. Model usulan terbaik, yaitu SVR dan ANN, akan diukur kembali kinerjanya dan dibandingkan dengan metode peramalan *existing* sehingga bisa diukur seberapa besar perbaikan kinerja peramalan yang dihasilkan.

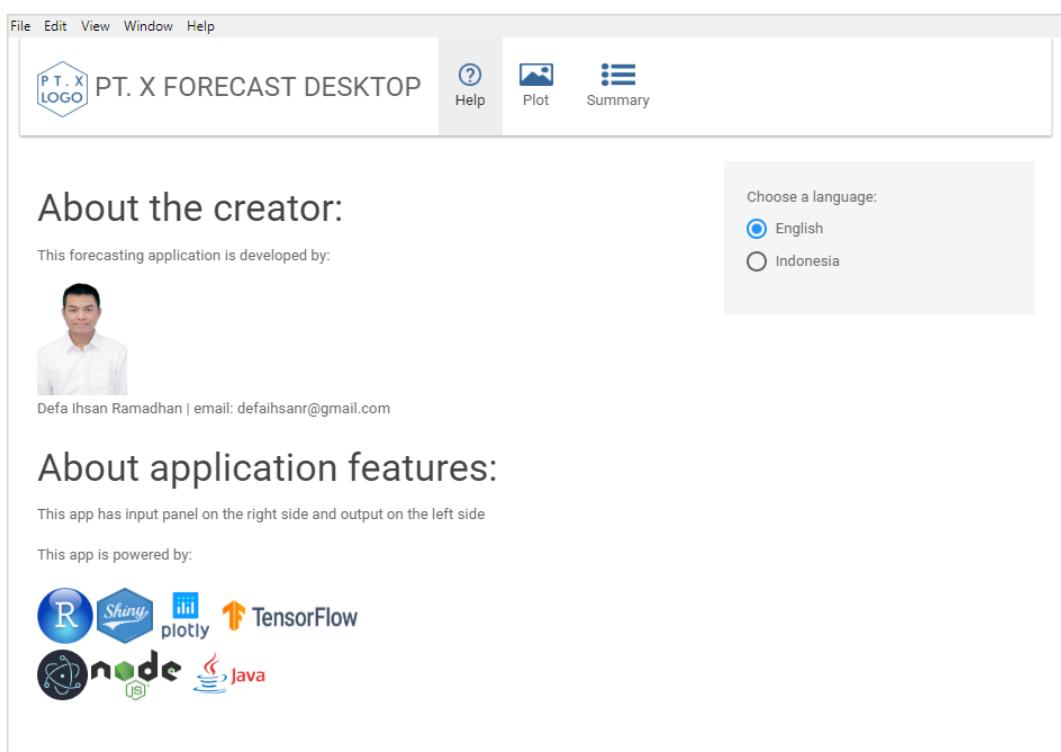
Perbaikan kinerja yang dihasilkan oleh metode SVR dan ANN dari metode *existing* apabila ditinjau dari MSE keseluruhan permintaan *part* yang digunakan pada implementasi ini adalah 18.8% (atau dalam kata lain terjadi pengurangan *error* sebesar 18.8%). Perbaikan tersebut cukup besar walaupun nilai *relative error* MAPE yang dihasilkan metode SVR dan ANN pada implementasi ini belum memenuhi kriteria “*good forecast*” dari Lewis (1982) yaitu MAPE dibawah 20%. Hal ini disebabkan adanya *trade off* antara akurasi dan risiko *overfitting* sehingga ada kompromi terhadap akurasi peramalan untuk menghasilkan model peramalan yang lebih general (tidak rawan *overfitting*).

Meskipun demikian, seperti yang ditunjukkan pada **Sub-sub-bab 5.4.2**, model peramalan SVR dan ANN yang lebih general mampu menghindari duplikasi yang berlebihan terhadap kejadian dari riwayat permintaan yang tidak berulang di periode selanjutnya (seperti kejadian *outlier*) sehingga tidak rawan mengalami *overforecast* atau *underforecast* dibandingkan metode *existing*. Selain itu, apabila mengikuti kriteria ACF pada **Sub-sub-bab 5.4.2** dari Hyndman & Athanasopoulos

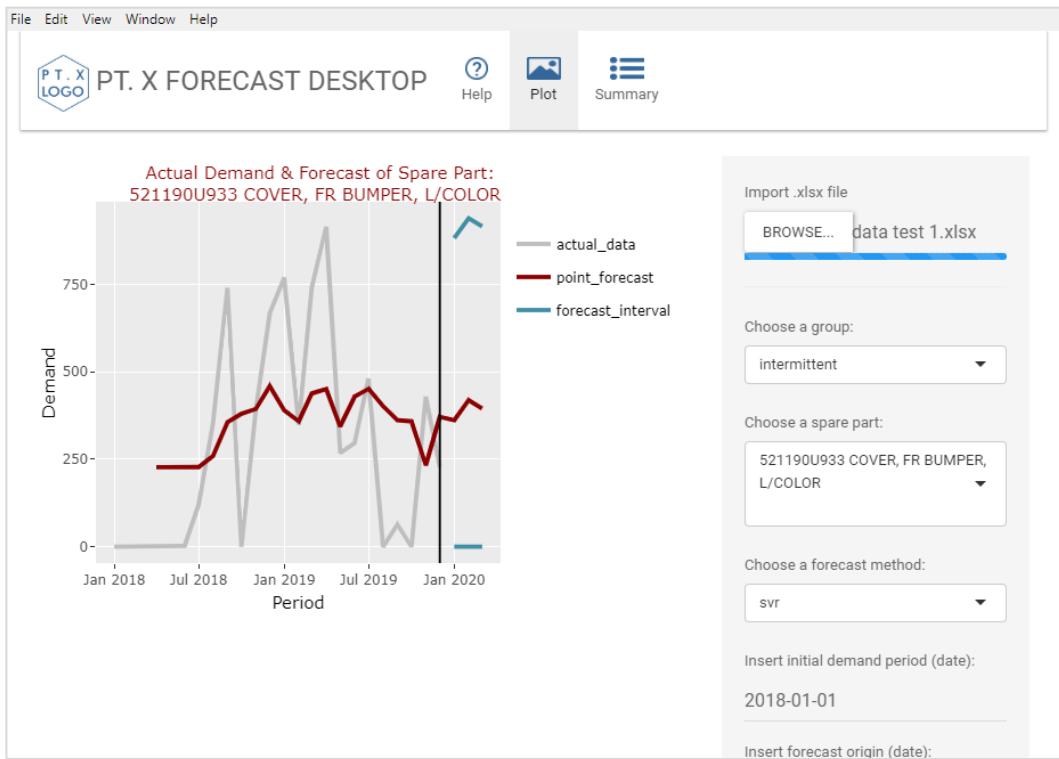
(2018), hasil peramalan SVR dan ANN tetap bisa dikategorikan sebagai “*good forecast*”.

6.3 Rencana Implementasi Sistem Peramalan dalam Operasional di PT. X

Seperti yang diketahui dari hasil pada analisis sistem pengelompokan pola permintaan *part* dan perbandingan kinerja berbagai model peramalan terhadap permintaan *part* di PT. X, klasifikasi Boylan sebagai pendukung keputusan terhadap hasil peramalan dan model peramalan SVR dan ANN dipilih sebagai usulan untuk perbaikan sistem peramalan permintaan *spare part* di PT. X. Dalam implementasinya pada operasional di PT. X maka usulan tersebut akan dibuat menjadi suatu aplikasi peramalan yang interaktif dan fiturnya bisa disesuaikan oleh pengguna. Skema dari aplikasi ini dapat dilihat pada **Lampiran 89**. Kemudian tampilan dari aplikasi ini dapat dilihat pada **Gambar 6. 1** berikut.



Gambar 6. 1 Tampilan Depan Pada Aplikasi



Gambar 6. 2 Tampilan Utama Pada Aplikasi

The screenshot shows the main interface of the PT.X Forecast Desktop application. At the top, there's a menu bar with File, Edit, View, Window, Help, a logo for PT.X Forecast Desktop, and buttons for Help, Plot, and Summary.

The central part of the screen displays a table titled "Report". It has a "DOWNLOAD" button above it. The table has four columns: time, point_forecast, forecast_upper_interval, and forecast_lower_interval. The data is as follows:

time	point_forecast	forecast_upper_interval	forecast_lower_interval
Jan 2020	362.08	882.79	0.00
Feb 2020	419.04	939.76	0.00
Mar 2020	395.27	915.98	0.00

Gambar 6. 3 Tampilan Report Pada Aplikasi

Seperti yang bisa dilihat pada **Gambar 6.2** dan **6.3**, setiap hasil peramalan akan dihitung juga interval (batas atas dan bawah) dari peramalan. Interval peramalan tersebut juga berfungsi sebagai pendukung dalam pengambilan keputusan pasca peramalan. Dalam operasional di PT. X, Interval peramalan bisa membantu pengambil keputusan dalam melakukan respon terhadap hasil peramalan, seringkali respon tersebut berupa keputusan perencanaan terhadap stok *spare part* dan interval peramalan berperan sebagai batas ketika melebihkan hasil peramaan permintaan *part* untuk *security stock*.

(Halaman ini sengaja dikosongkan)

BAB 7

KESIMPULAN DAN SARAN

BAB 7

KESIMPULAN DAN SARAN

Pada bab ini akan dijelaskan kesimpulan yang menjawab tujuan pada tugas akhir ini dan saran yang diusulkan untuk perbaikan sistem peramalan di PT. X dan pengembangan penelitian selanjutnya.

7.1 Kesimpulan

Kesimpulan yang bisa diambil dari analisis pada tugas akhir ini adalah sebagai berikut.

1. Sistem pengelompokan *part* di PT. X berdasarkan pola permintaannya bisa berfungsi sebagai pendukung keputusan terhadap hasil peramalan karena perbedaan pola permintaan *part* menghasilkan kompleksitas dan ketidakpastian yang berbeda-beda terhadap hasil peramalan. Sistem pengelompokan dengan klaster K-Medoids lebih efektif dalam menangkap perbedaan pola permintaan *part* akan tetapi perlu diteliti lebih lanjut agar bisa digunakan pada operasional di PT. X. Oleh karena itu, klasifikasi Boylan dipilih untuk diimplementasikan pada operasional di PT. X karena memiliki pengaruh terhadap kinerja peramalan yang sama-sama signifikan dengan klaster K-Medoids.
2. Metode-metode peramalan yang menjadi usulan perbaikan, yaitu: Croston, SBA, TSB, SVR, dan ANN, saling dibandingkan kinerjanya dengan diukur menggunakan parameter *forecasting error* dan *robustness*. Metode SVR dan ANN menjadi metode yang dipilih karena memiliki kinerja yang lebih baik diantara yang lain di tahap *training* maupun *testing*. Ketika diimplementasikan untuk peramalan multi periode, metode SVR dan ANN juga mengungguli kinerja metode peramalan *existing* dan memperbaiki atau mengurangi kesalahan peramalan secara keseluruhan sebesar 18.8%. Berdasarkan evaluasi ACF menurut Hyndman & Athanasopoulos (2018), hasil peramalan SVR dan ANN pada tugas akhir ini juga memenuhi kriteria “*good forecast*”.

3. Dalam tugas akhir ini, rencana implementasi sistem peramalan yang lebih akurat dan *robust* pada operasional di PT. X diusulkan dalam bentuk aplikasi peramalan yang interaktif dan bisa disesuaikan oleh pengguna. Aplikasi ini memiliki fitur: 1) pengelompokan pola permintaan *part* berdasarkan klasifikasi Boylan, yaitu: “SMOOTH”, “INTERMITTENT”, “ERRATIC”, dan “LUMPY”, 2) peramalan permintaan dengan metode *data mining*, yaitu SVR dan ANN, dan 3) perhitungan interval peramalan sebagai pendukung keputusan pasca peramalan yang bisa berfungsi sebagai batas dalam perencanaan *security stock*.

7.2 Saran

Saran-saran yang bisa dipertimbangkan dari hasil tugas akhir ini diantaranya adalah sebagai berikut.

1. PT. X disarankan untuk mempertimbangkan perbedaan karakteristik pada pola permintaan masing-masing *part*. Oleh karena itu, perlu ada respon yang berbeda-beda terhadap hasil peramalan permintaan *part* menyesuaikan karakter pola permintaannya.
2. PT. X disarankan untuk mengganti metode peramalan *existing* dengan metode yang memberikan hasil peramalan yang lebih general, seperti metode SVR dan ANN dari tugas akhir ini. Hal ini bertujuan untuk meminimalisasi hasil peramalan yang ekstrem (*overforecast* atau *underforecast*) yang bisa berdampak pada biaya operasional.
3. PT. X disarankan untuk mempertimbangkan implementasi aplikasi peramalan secara luas di operasionalnya sehingga bisa mengurangi pekerjaan yang berlebihan (*excessive*) oleh pengguna, mengurangi kesalahan perhitungan, dan meningkatkan efisiensi waktu dan biaya.
4. Pada penelitian selanjutnya diharapkan adanya evaluasi secara berkelanjutan pada metode SVR dan ANN di tugas akhir ini terhadap metode yang lebih relevan kedepannya. Selain itu, skenario peramalan menggunakan data dengan periode yang lebih singkat, seperti harian dan mingguan, bisa dievaluasi apakah memberikan hasil yang lebih baik dari tugas akhir ini. Terakhir, hasil pengelompokan *part* dari klaster K-

Medoids bisa didiskusikan lebih lanjut dengan *stakeholder* PT. X sehingga bisa digunakan pada operasional di PT. X.

REFERENSI DAN DAFTAR PUSTAKA

REFERENSI DAN DAFTAR PUSTAKA

- Adhikari, R. & Agrawal, R. K., 2013. *An Introductory Study on Time-Series Modeling & Forecasting*. s.l.:LAP Lambert Academic Publishing.
- Aghabozorgi, S. et al., 2014. A Hybrid Algorithm for Clustering of Time-Series Data Based on Affinity Search Technique. *The Scientific World Journal*, 25 Maret, hh. 1-12.
- Altay, N. & Litteral, L. A., 2011. *Spare Part Management: Demand Forecasting & Inventory Control*. London: Springer-Verlag.
- Amirkolaii, K. N., Baboli, A., Shahzad, M. K. & Tonadre, R., 2017. Demand Forecasting for Irregular Demands in Business Aircraft Spare Part Supply Chains by using Artificial Intelligence (AI). *IFAC-PapersOnLine*, 50(1), hh. 15221-15226.
- Amstrong, J. S., 1985. *Long-range Forecasting: From Crystal Ball to Computer. 2nd Edition*. s.l.:Wiley.
- Babai, M., Dallery, Y., Boubaker, S. & Kalai, R., 2017. A New Method to Forecast Intermittent Demand in The Presence of Inventory Obsolescence. *International Journal of Production Economics*, hh. 1-12.
- Bhande, A., 2018. *What is Underfitting & Overfitting in Machine Learning & How to Deal with It. [Online]*
Dari: <https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76>
[Diakses 10 Maret 2020].
- Biedermann, H., 2008. *Spare Part Management: Efficient Spare Part Logistics. 2nd Edition*. Heidelberg: Springer-Verlag.
- Boylan, J. E., Syntetos, A. A. & Croston, J. D., 2005. On The Categorization of Demand Patterns. *Journal of The Operational Research Society*, Volume 56, hh. 495-503.
- Canova, F. & Hansen, B., 1995. Are Seasonal Patterns Constant over Time? A Test for Seasonal Stability. *Journal of Business & Economic Statistics*, 13(3), pp. 237-252.

- Cao, L. J. & Tay, F. E. H., 2003. Support Vector Machine with Adaptive Parameters in Financial Time-Series Forecasting. *IEEE Transactions on Neural Networks*, 14(6), hh. 1506-1518.
- Chawla, G. & Miceli, V. M., 2019. *Demand Forecasting & Inventory Management for Spare Part*, s.l.: MASSACHUSETTS INSTITUTE OF TECHNOLOGY.
- Chen, C., 1997. Robustness Properties Of Some Forecasting Methods For Seasonal Time-Series: A Monte Carlo Study. *International Journal of Forecasting*, 13(2), pp. 269-280.
- Chopra, S. & Meindl, P., 2016. *Supply Chain Management: Strategy, Planning, & Operation. 6th Edition*. Essex, NE: Pearson Education.
- Cochrane, C., 2018. *Time-Series Nested Cross-Validation*. [Online] Dari: <https://towardsdatascience.com/time-series-nested-cross-validation-76adba623eb9> [Diakses 18 Maret 2020].
- Croston, J. D., 1972. Forecasting & Stock Control for Intermittent Demands. *Operational Research Quarterly (1970-1977)*, 23(3), hh. 289-303.
- Dickey, D. & Fuller, W., 1979. Distribution of the Estimators for Autoregressive Time-Series with a Unit Root. *Journal of the American Statistical Association*, Volume 74, pp. 427-431.
- Dombrowski, U. & Schulze, S., 2008. Life Cycle-Oriented Spare Part Management: New Challenges. Dalam: *Contributions to A Theory of Logistics*. Heidelberg: Springer-Verlag, hh. 439-462.
- Driessens, M. A. et al., 2014. Maintenance Spare Part Planning & Control: A Framework for Control & Agenda for Future Research. *Production Planning & Control*, hh. 407-426.
- Eaves, A. H. C. & Kingsman, B. G., 2004. Forecasting for The Ordering & Stock-Holding of Spare Part. *Journal of the Operational Research Society*, 55(4), hh. 431-437.
- EY, 2018. *Finding the Automotive Distribution Model of Tomorrow*. [Online] Dari: [https://www.ey.com/Publication/vwLUAssets/POV_Auto_Retail/\\$FILE/P](https://www.ey.com/Publication/vwLUAssets/POV_Auto_Retail/$FILE/P)

oV-Auto-Retail-2018.pdf

[Diakses 11 Februari 2020].

Fourier, J., 1822. *Analytical Theory of Heat*. Paris: Didot.

Gaikindo, 2019. *Indonesian Automobile Industry Data*. [Online]

Dari: <https://www.gaikindo.or.id/indonesian-automobile-industry-data/>

[Diakses 11 Februari 2020].

Glorot, X. & Bengio, Y., 2010. *Understanding The Difficulty of Training Deep Feedforward Neural Networks*. s.l., Proceedings of The Thirteenth International Conference on Artificial Intelligence & Statistics.

Guerrero, V., 1993. Time-Series Analysis Supported by Power Transformations. *Journal of Forecasting*, 12(1), pp. 37-48.

Gutierrez, R. S., Solis, A. O. & Mukhopadhyay, S., 2008. Lumpy Demand Forecasting Using Neural Networks. *International Journal of Production Economics*, 111(2), hh. 409-420.

Hakim, I. M. & Dwantara, H., 2018. Forecasting Spare Part Demand on Automotive Industry Using Artificial Neural Network (ANN). *International Journal of Engineering & Technology*, 7(3.7), hh. 444-447.

Hamilton, J. D., 1994. *Time-Series Analysis. 1st Edition*. s.l.:Princeton University Press.

Henkelmann, R., 2018. *A Deep Learning based Approach for Automotive Spare Part Demand Forecasting, Master's Thesis*, Magdeburg: Otto von Guericke University Magdeburg.

Hoover, J., 2006. Measuring Forecast Accuracy: Omissions in Today's Forecasting Engines & Demand-Planning Software. *Foresight: The International Journal of Applied Forecasting, International Institute of Forecasters, Issue 1*, hh. 32-35.

Hua, Z. & Zhang, B., 2006. A Hybrid Support Vector Machines & Logistic Regression Approach for Forecasting Intermittent Demand of Spare Part. *Applied Mathematics & Computation*, 181(2), hh. 1035-1048.

Hyndman, R. J., 2016. *Cross-Validation for Time-Series*. [Online]

Dari: <https://robjhyndman.com/hyndsoft/tscv/>

[Diakses 18 Maret 2020].

- Hyndman, R. J. & Athanasopoulos, G., 2018. *Forecasting Principles & Practice*.
2nd Edition. s.l.:OTexts.
- Hyndman, R. J. & Koehler, A. B., 2006. Another Look at Measures of Forecast Accuracy. *International Journal of Forecasting*, 22(4), hh. 679-688.
- Jiang, P., Liu, X., Huang, Y. & Yuan, Y., 2017. *An Adaptive Data-driven Model for Erratic Demand Forecasting*. Toulouse, The 20th IFAC World Congress.
- Kaya, G. O. & Turkyilmaz, A., 2018. Intermittent Demand Forecasting Using Data Mining Techniques. *Applied Computer Science*, 14(2), hh. 38-47.
- Kendall, M., 1975. *Rank Correlation Methods*. 4th Edition. London: Charles Griffin.
- Klug, F., 2018. *Logistics Management in The Automotive Industry: Basics of Logistics in Automotive Engineering*. 2nd Edition. Heidelberg: Springer-Verlag.
- Kodinariya, T. M. & Makwana, P. R., 2013. Review on Determining Number of Cluster in K-Means Clustering. *International Journal of Advance Research in Computer Science & Management Studies*, 1(6), hh. 90-95.
- Kruskal & Wallis, 1952. Use of Ranks In One-Criterion Variance Analysis. *Journal of the American Statistical Association*, 47(260), pp. 583-621.
- Kutbay, U., 2018. Partitional Clustering. Dalam: *Recent Applications in Data Clustering*. s.l.:IntechOpen, hh. 19-34.
- Lee, H. & Kim, J., 2018. *A Predictive Model for Forecasting Spare Part Demand in Military Logistics*. Bangkok, 2018 IEEE International Conference on Industrial Engineering & Engineering Management (IEEM).
- Levene, H., 1960. Robust Tests for Equality of Variances. *Contributions to Probability & Statistics: Essays in Honor of Harold Hotelling*, pp. 278-292.
- Lewis, C. D., 1982. *Industrial & Business Forecasting Methods*. London: Butterworths.
- Loukmidis, G., 2014. *Adaptive Spare Part Requirements Planning, PhD Dissertation*, Aachen: RWTH.

- Loukmidis, G. & Luczak, H., 2006. Life Cycle Oriented Planning Strategies for Spare Part Needs. Dalam: *Successful with After Sales Service: Business Strategies for Service Management & Spare Part Logistics*. Heidelberg: Springer-Verlag, hh. 251-270.
- Mahdevari, S., Shahriar, K., Yagiz, S. & Shirazi, M. A., 2014. A Support Vector Regression Model for Predicting Tunnel Boring Machine Penetration Rates. *International Journal of Rock Mechanics & Mining Sciences*, Volume 72, hh. 214-229.
- Mann, H., 1945. Non-Parametric Tests Against Trend. *Econometrica*, Volume 13, pp. 163-171.
- McKinsey&Company, 2018. *Ready For Inspection - The Automotive Aftermarket in 2030*. [Online]
- Dari:
- <https://www.mckinsey.com/~/media/McKinsey/Industries/Automotive%20and%20Assembly/Our%20Insights/Ready%20for%20inspection%20The%20automotive%20aftermarket%20in%202030/Ready-for-inspection-The-automotive-aftermarket-in-2030-vF.ashx>
- [Diakses 11 Februari 2020].
- McKinsey&Company, 2018. *The Changing Aftermarket Game - & How Automotive Suppliers Can Benefit from Arising Opportunities*. [Online]
- Dari:
- <https://www.mckinsey.com/~/media/McKinsey/Industries/Automotive%20and%20Assembly/Our%20Insights/The%20changing%20aftermarket%20game%20and%20how%20automotive%20suppliers%20can%20benefit%20from%20arising%20opportunities/The-changing-aftermarket-game.ashx>
- [Diakses 11 Februari 2020].
- Mitchell, T. M., 1997. *Machine Learning. 1st Edition*. New York: McGraw-Hill, Inc..
- Montgomery, D. C., Jennings, C. L. & Kulahci, M., 2008. *Introduction to Time-Series Analysis & Forecasting*. New Jersey: John Wiley & Sons, Inc..

- Nijs, R. d. & Klausen, T. L., 2013. On The Expected Difference Between Mean & Median. *Electronic Journal of Applied Statistical Analysis*, 6(1), hh. 110-117.
- Nwankpa, C. E., Ijomah, W., Gachagan, A. & Marshall, S., 2018. *Activation Functions: Comparison of Trends in Practice & Research for Deep Learning*. [Online]
- Dari: <https://arxiv.org/pdf/1811.03378.pdf>
[Diakses 20 Februari 2020].
- Palit, A. K. & Popovic, D., 2005. *Computational Intelligence in Time-Series Forecasting: Theory & Engineering Applications*. London: Springer-Verlag.
- Patel, S., 2017. *Chapter 2 : SVM (Support Vector Machine) — Theory*. [Online]
- Dari: <https://medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72>
[Diakses 17 Maret 2020].
- Pfohl, H. C., 2010. *Logistics Systems: Business Basics. 8th Edition*. Berlin: Springer-Verlag.
- Piatesky-Shapiro, G., Fayyad, U. & Smyth, P., 1996. *Knowledge Discovery & Data Mining: Towards a Unifying Framework*. Portland, KDD-96 Proceedings.
- PT. X, 2019. *Supplier Getsudo Meeting*, s.l.: s.n.
- Radhakrishnan, P., 2017. *What are Hyperparameters ? & How to tune the Hyperparameters in a Deep Neural Network?*. [Online]
- Dari: <https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a>
[Diakses 17 Maret 2020].
- Roelofsen, P., 2018. *Time-Series Clustering, Master Thesis Business Analytics*, Amsterdam: Vrije Universiteit Amsterdam.
- Rumelhart, D. E., Hinton, G. E. & Williams, R. J., 1986. Learning Representations by Back-Propagating Errors. *Nature*, Volume 323, hh. 533-536.

- Şahin, M., Kizilaslan, R. & Demirel, F. O., 2013. Forecasting Aviation Spare Part Demand Using Croston Based Methods & Artificial Neural Network. *Journal of Economic & Social Research*, 15(2), hh. 1-21.
- Santosa, B. & Umam, A., 2018. *Data Mining dan Big Data Analytics: Teori dan Implementasi Menggunakan Python & Apache Spark*. Edisi ke-2. Yogyakarta: Penebar Media Pustaka.
- Sardá-Espinosa, A., 2018. *Comparing Time-Series Clustering Algorithms in R Using the dtwclust Package*. [Online]
- Dari:
- https://www.researchgate.net/publication/312159618_Comparing_Time-Series_Clustering_Algorithms_in_R_Using_the_dtwclust_Package [Diakses 17 Februari 2020].
- Schmidhuber, J., 2014. Deep Learning in Neural Networks: An Overview. *Neural Networks*, Volume 61, hh. 85-117.
- Schölkopf, B. & Smola, A. J., 2001. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, & Beyond*. s.l.:The MIT Press.
- Schröter, M., 2006. *Strategic Spare Part Management in Closed-Loop Supply Chains: A System Dynamic Approach*. 1st Edition. Dt. Univ.-Verl: Wiesbaden.
- Schuh, G. & Stich, V., 2013. *Logistics Management: Production & Management Manual*. 2nd Edition. Heidelberg: Springer-Verlag.
- Sembiring, R. W., Zain, J. M. & Embong, A., 2010. A Comparative Agglomerative Hierarchical Clustering Method to Cluster Implemented Course. *JOURNAL OF COMPUTING*, 2(12).
- Shahrabi, J., Mousavi, S. S. & Heydar, M., 2009. Supply Chain Demand Forecasting; A Comparison of Machine Learning Techniques & Traditional Methods. *Journal of Applied Sciences*, 9(3), hh. 521-527.
- Shapiro, S. S. & Wilk, M. B., 1965. An Analysis of Variance Test for Normality (Complete Samples). *Biometrika*, 52(3-4), pp. 591-611.
- Snedecor, G. W. & Cochran, W. G., 1967. *Statistical Methods*. 6th Edition. s.l.:Iowa State University Press.

- Stalfort, J., 2019. *Hyperparameter Tuning Using Grid Search & Random Search: A Conceptual Guide*. [Online]
- Dari: <https://medium.com/@jackstalfort/hyperparameter-tuning-using-grid-search-and-random-search-f8750a464b35>
[Diakses 17 Maret 2020].
- Statista, 2020. *Global Automotive Market Share in 2019, by Brand*. [Online]
- Dari: <https://www.statista.com/statistics/316786/global-market-share-of-the-leading-automakers/>
[Diakses 11 Februari 2020].
- Statista, 2020. *Market Share of New Car Sales in Saudi Arabia in 2019, by Select Brand*. [Online]
- Dari: <https://www.statista.com/statistics/673095/saudi-arabia-car-sales-market-share-by-brand/>
[Diakses 11 Februari 2020].
- Syntetos, A. A. & Boylan, J. E., 2005. The Accuracy of Intermittent Demand. *International Journal of Forecasting*, 21(2), hh. 303-314.
- Teunter, R. H., Syntetos, A. A. & Babai, M. Z., 2011. Intermittent Demand: Linking Forecasting to Inventory Obsolescence. *European Journal of Operational Research* 214, hh. 606-615.
- Toyota Motor Corporation, 2019. *How Many Part is Each Car Made of*. [Online]
- Dari: <https://www.toyota.co.jp/en/kids/faq/d/01/04/>
[Diakses 11 Februari 2020].
- Udofia, U., 2018. *Basic Overview of Convolutional Neural Network (CNN)*. [Online]
- Dari: <https://medium.com/dataseries/basic-overview-of-convolutional-neural-network-cnn-4fcc7dbb4f17>
[Diakses 20 Februari 2020].
- Vapnik, V., 2000. *The Nature of Statistical Learning Theory*. 2nd Edition. New York: Springer-Verlag.
- Vapnik, V., Golowich, S. E. & Smola, A., 1997. Support Vector Method for Function Approximation, Regression Estimation, & Signal Processing.

- Advances in Neural Information Processing Systems*, Volume 9, hh. 281-287.
- Widrow, B. & Hoff, M. E., 1960. *Adaptive Switching Circuits*, hh. 96-104. s.l., IRE WESCON Convention Record.
- Williams, T., 1984. Stock Control with Sporadic & Slow-Moving Demand. *Journal of the Operational Research Society*, 35(10), hh. 939-948.
- Xu, H. & Mannor, S., 2010. *Robustness and Generalization*. [Online] Dari: <https://arxiv.org/pdf/1005.2243.pdf> [Diakses 19 05 2020].
- Zhang, Z. et al., 2018. *Kernel-Induced Label Propagation by Mapping for Semi-Supervised Classification*. s.l., IEEE Transactions on Big Data 2018.

LAMPIRAN

LAMPIRAN

- **Normalisasi Skala**

The screenshot shows the RStudio interface with the console tab active. The code in the console is as follows:

```
R
File Edit Code View Plots Session Build Debug Profile Tools Help
+ | Go to file/function | Addins

Console Terminal ×
C:/Users/X302U/Desktop/
> dataset1
  Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018  0   0   0   0   0   2 120 351 740 0   394 669
2019 770 349 742 915 268 296 481 0   63 0   429 228
> (max<-max(dataset1))
[1] 915
> (min<-min(dataset1))
[1] 0
> normal_dataset1 <- ((dataset1 - min) / (max - min)) * (1-(-1)) + (-1)
> normal_dataset1
      Jan        Feb        Mar        Apr        May        Jun
2018 -1.00000000 -1.00000000 -1.00000000 -1.00000000 -1.00000000 -0.99562842
2019  0.68306011 -0.23715847  0.62185792  1.00000000 -0.41420765 -0.35300546
      Jul        Aug        Sep        Oct        Nov        Dec
2018 -0.73770492 -0.23278689  0.61748634 -1.00000000 -0.13879781  0.46229508
2019  0.05136612 -1.00000000 -0.86229508 -1.00000000 -0.06229508 -0.50163934
> |
```

Lampiran 1 *Min-Max Normalization* Menggunakan RStudio

- **Konversi Ke-Bentuk *Predictor-Target***

The screenshot shows the RStudio interface with the console tab active. The code in the console is as follows:

```
R
File Edit Code View Plots Session Build Debug Profile Tools Help
+ | Go to file/function | Addins

Console Terminal ×
C:/Users/X302U/Desktop/
> head(dataset1,10)
  [,1]
[1,] 0
[2,] 0
[3,] 0
[4,] 0
[5,] 0
[6,] 2
[7,] 120
[8,] 351
[9,] 740
[10,] 0
> hasil<-konversi(data = dataset1, k = 3)
> colnames(hasil)<-c("pred1","pred2","pred3","target")
> head(hasil,7)
      pred1 pred2 pred3 target
[1,] 0     0     0     0
[2,] 0     0     0     0
[3,] 0     0     0     2
[4,] 0     0     2     120
[5,] 0     2     120   351
[6,] 2     120   351   740
[7,] 120   351   740   0
> |
```

Lampiran 2 Konversi Ke-Bentuk *Predictor Target* Menggunakan RStudio

- **Klasifikasi Boylan**

RStudio interface showing the console tab. The command `BCF.class(ADT_cv2)` is run, resulting in the output [1] "Category: Intermittent".

```

R
File Edit Code View Plots Session Build Debug Profile Tools Help
+ - Go to file/function Addins
Console Terminal x
C:/Users/Y302U/Desktop/
> dataset1
   Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018    0   0   0   0   0   2 120 351 740   0 394 669
2019 770 349 742 915 268 296 481   0   63   0 429 228
> (ADI <- idclass(t(dataset1),type="SBC")$p)
[1] 1.5
> (cv2 <- idclass(t(dataset1),type="SBC")$cv2)
[1] 0.4101521
> SBC.class(ADT_cv2)
[1] "Category: Intermittent"
>

```

Lampiran 3 Klasifikasi Boylan Menggunakan RStudio (1)

RStudio interface showing the console tab. The command `xtabs(~all_dataset$BOYLAN,data = all_dataset)/1682` is run, resulting in a table where the row for 'BOYLAN' has been highlighted.

```

R
File Edit Code View Plots Session Build Debug Profile Tools Help
+ - Go to file/function Addins
Console Terminal x
C:/Users/Y302U/Desktop/
> colnames(all_dataset)
[1] "PART.NO." "NAMA.PART" "DIMENSI"      "MODEL"       "FAMILY"      "TUJUAN"      "X2018.01"
[8] "X2018.02"  "X2018.03"  "X2018.04"    "X2018.05"    "X2018.06"    "X2018.07"    "X2018.08"
[15] "X2018.09"  "X2018.10"  "X2018.11"    "X2018.12"    "X2019.01"    "X2019.02"    "X2019.03"
[22] "X2019.04"  "X2019.05"  "X2019.06"    "X2019.07"    "X2019.08"    "X2019.09"    "X2019.10"
[29] "X2019.11"  "X2019.12"  "BOYLAN"      "K.MEDOID"
> nrow(all_dataset)
[1] 1682
> xtabs(~all_dataset$BOYLAN,data = all_dataset)
all_dataset$BOYLAN
  ERRATIC INTERMITTENT      LUMPY      SMOOTH
  108        624        921        29
> xtabs(~all_dataset$BOYLAN,data = all_dataset)/1682
all_dataset$BOYLAN
  ERRATIC INTERMITTENT      LUMPY      SMOOTH
  0.06420927  0.37098692  0.54756243  0.01724138
>

```

Lampiran 4 Klasifikasi Boylan Menggunakan RStudio (2)

- K-Medoids dan *Agglomerative Hierarchical Clustering*

```

Console Terminal ×
C:/Users/X302U/Desktop/
> m <- c("average","single","complete","ward")
> names(m) <- c("average","single","complete","ward")
> ahc1 <- function(x) {
+   agnes(n_dataset,method=x)$ac
+ }
> (bestlink1<-map_dbl(m,ahc1))
  average single complete      ward
0.7012211 0.6514547 0.7769133 0.9628100
>

```

Lampiran 5 Penentuan Ukuran *Linkage* Terbaik (Ward) Menggunakan RStudio

```

R RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function Addins
Console Terminal ×
C:/Users/X302U/Desktop/
> dist1 <- dist(n_dataset,method="euclidean")
>
> # K-Medoids Clustering
> elb1 <- fviz_nbclust(n_dataset,pam,method="wss",diss=dist1,k.max=10)+ 
+   labs(subtitle="Elbow Method")
> write.csv(elb1$data,file="/Users/X302U/Desktop/elb1_out.csv",row.names=F)
> wss1<-elb1$data[c(ns1,nch1),]
> opt_clust1<-wss1[wss1[2]==min(wss1[2]),]
> names(opt_clust1)<-c("clusters","sum of squared within")
> opt_clust1
  clusters sum of squared within
9           9          7466.369
>

```

Lampiran 6 Proses K-Medoids *Clustering* Seluruh 1682 Pola Permintaan Part

```

> # Agglomerative Clustering
> link1
[1] "ward.D2"
> hc1<-hclust(dist1,method=link1)
> k_hc1<-cutree(hc1,k=9)
> write.csv(k_hc1,file="/Users/X302U/Desktop/hc1_out.csv",
+ row.names=F)
> ssw1<-fviz_nbclust(n_dataset,FUN=hcut,method="wss")
> ssw1<-ssw1$data[9,]
> names(ssw1)<-c("clusters","sum of squared within")
> ssw1
  clusters sum of squared within
9           9          7898.041
>

```

Lampiran 7 Proses Agglomerative Clustering Seluruh 1682 Pola Permintaan Part

R

File Edit Code View Plots Session Build Debug Profile Tools Help

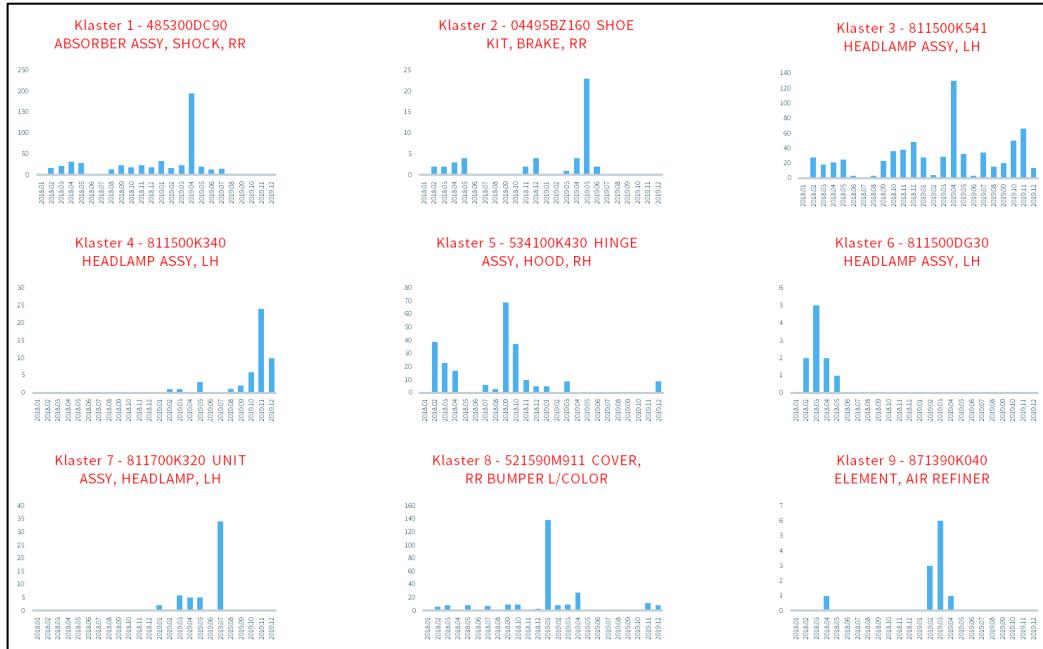
Go to file/function Addins

Console Terminal

C:/Users/X302U/Desktop/

```
> colnames(all_dataset)
[1] "PART.NO." "NAMA.PART" "DIMENSI" "MODEL" "FAMILY" "TUJUAN" "X2018.01"
[8] "X2018.02" "X2018.03" "X2018.04" "X2018.05" "X2018.06" "X2018.07" "X2018.08"
[15] "X2018.09" "X2018.10" "X2018.11" "X2018.12" "X2019.01" "X2019.02" "X2019.03"
[22] "X2019.04" "X2019.05" "X2019.06" "X2019.07" "X2019.08" "X2019.09" "X2019.10"
[29] "X2019.11" "X2019.12" "BOYLAN" "K.MEDOIDIS"
> nrow(all_dataset)
[1] 1682
> xtabs(~all_dataset$K.MEDOIDIS,data = all_dataset)
all_dataset$K.MEDOIDIS
  K1   K2   K3   K4   K5   K6   K7   K8
298 159 299 282 123 147 116 150 108
> xtabs(~all_dataset$K.MEDOIDIS,data = all_dataset)/1682
all_dataset$K.MEDOIDIS
  K1   K2   K3   K4   K5   K6   K7   K8
0.17717004 0.09453032 0.17776457 0.16765755 0.07312723 0.08739596 0.06896552 0.08917955
  K9
0.06420927
>
```

Lampiran 8 Hasil K-Medoids Clustering Menggunakan RStudio



Lampiran 9 Pola Permintaan Part Yang Menjadi Pusat Masing-Masing Klaster

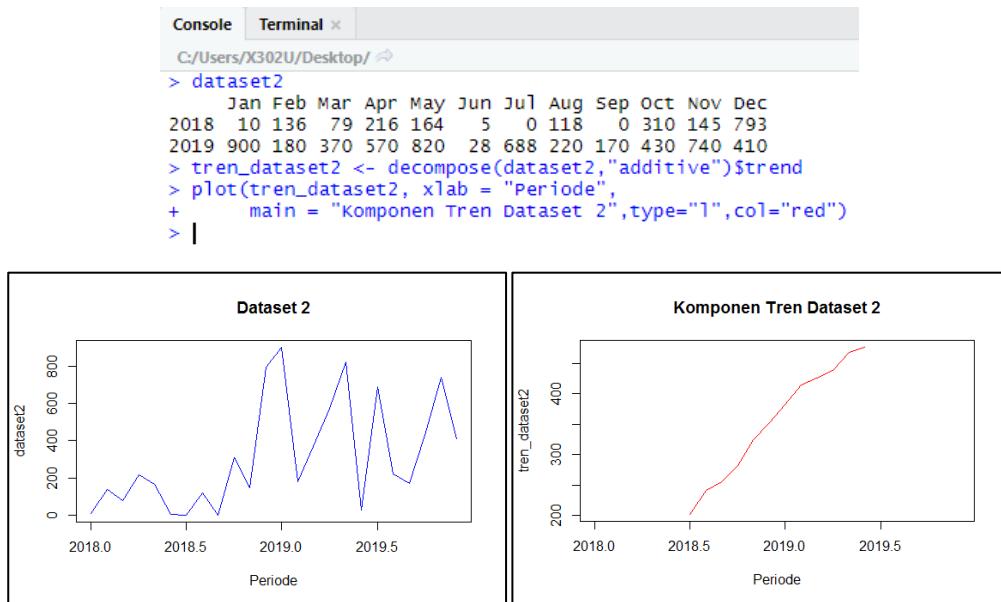
- *Trend dan Seasonal Adjustment*

```

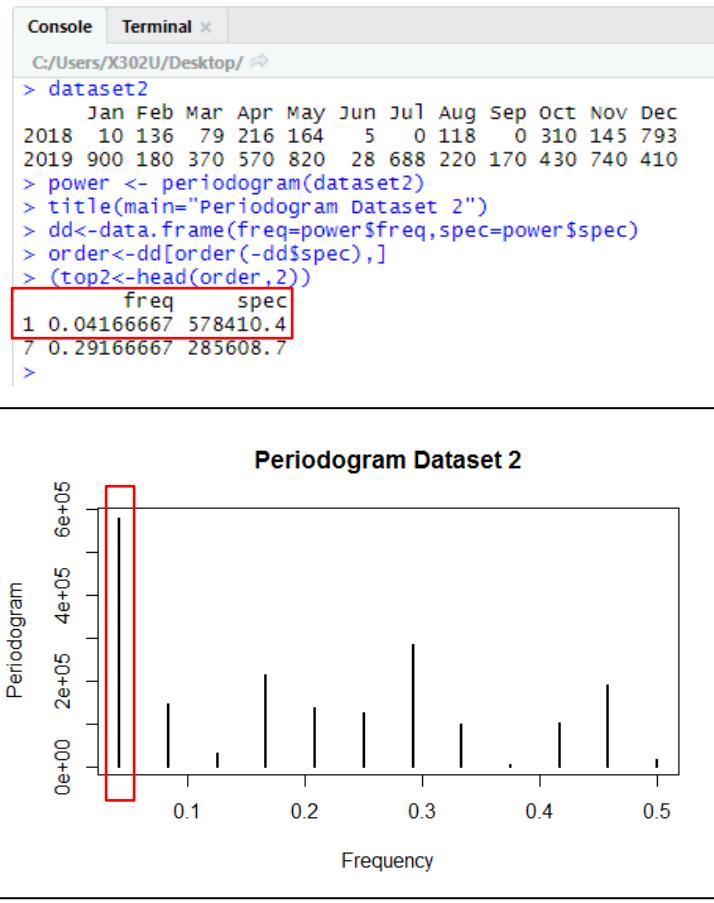
Console Terminal x
C:/Users/X302U/Desktop/ ↵
> dataset2
  Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018 10 136 79 216 164 5 0 118 0 310 145 793
2019 900 180 370 570 820 28 688 220 170 430 740 410
> (p <- mk.test(dataset2)$p.value)
[1] 0.007368817
> p <= 0.05
[1] TRUE
> param <- mk.test(dataset2)$estimates
> param
  S           vars          tau
109.0000000 1624.3333333   0.3956449
>

```

Lampiran 10 Hasil Uji Mann-Kendall Menggunakan RStudio



Lampiran 11 Dekomposisi Tren (Merah) Menggunakan RStudio



Lampiran 12 Transformasi Fourier Menggunakan RStudio

Console Terminal ×

C:/Users/X302U/Desktop/ ↗

```
> dataset2
   Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018  10 136  79 216 164   5   0 118   0 310 145 793
2019 900 180 370 570 820  28 688 220 170 430 740 410
> top2
   freq      spec
1 0.04166667 578410.4
7 0.29166667 285608.7
> (seasonality <- 1/top2$freq[1])
[1] 24
>
```

Lampiran 13 Penentuan *Seasonality* Menggunakan RStudio (1)

Console Terminal x

C:/Users/X302U/Desktop/ ↗

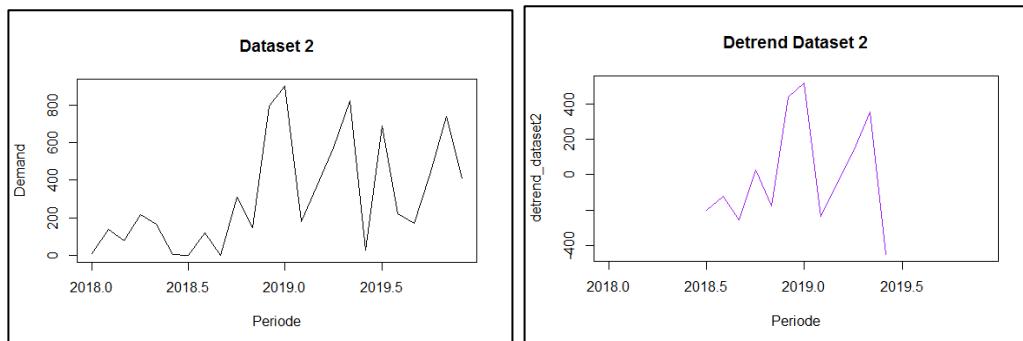
```
> dataset3
   Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018  0 148 94 48 151 0 0 14 70 48 59 89
2019 103 85 72 249 70 46 63 0 43 117 183 62
> top2
      freq      spec
7 0.2916667 24058.59
4 0.1666667 17624.33
> (seasonality <- 1/top2$freq[1])
[1] 3.428571
> |
```

Lampiran 14 Penentuan *Seasonality* Menggunakan RStudio (2)

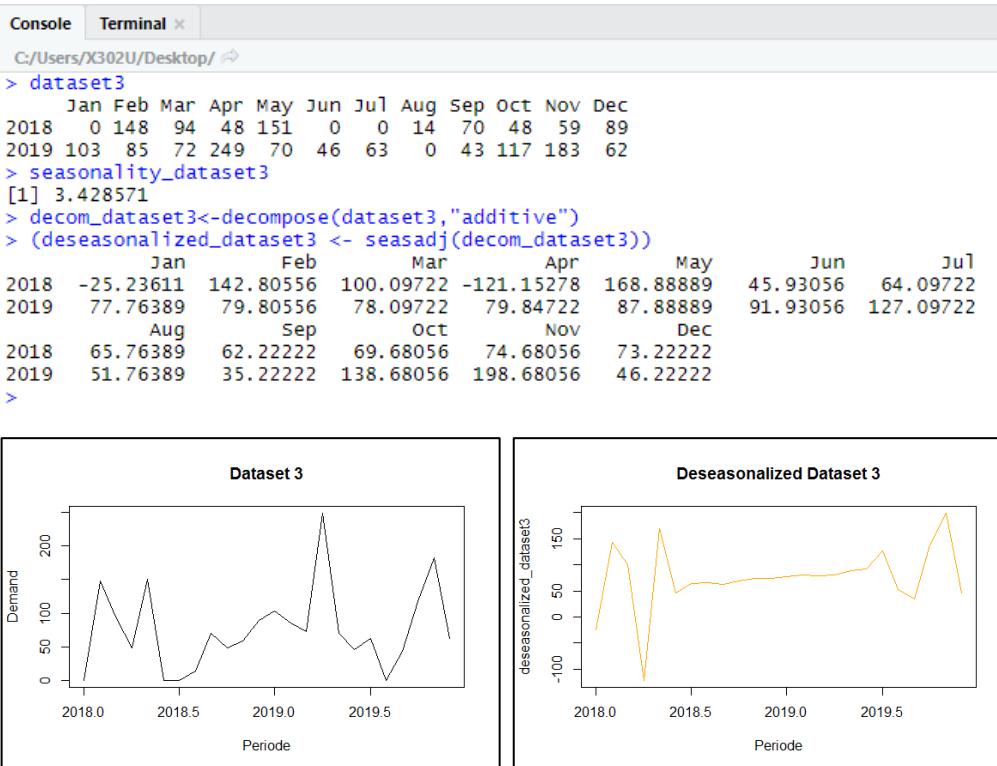
Console Terminal x

C:/Users/X302U/Desktop/ ↗

```
> dataset2
   Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018 10 136 79 216 164 5 0 118 0 310 145 793
2019 900 180 370 570 820 28 688 220 170 430 740 410
> (detrend_dataset2 <- dataset2-decompose(dataset2,"additive")$trend)
   Jan      Feb      Mar      Apr      May      Jun      Jul      Aug
2018 NA       NA       NA       NA       NA       NA -201.7500 -122.6667
2019 518.5000 -234.4167 -55.7500 132.1667 352.3750 -448.4583 NA       NA
   Sep      Oct      Nov      Dec
2018 -254.6250 28.5000 -178.5833 441.1250
2019 NA       NA       NA       NA
> |
```

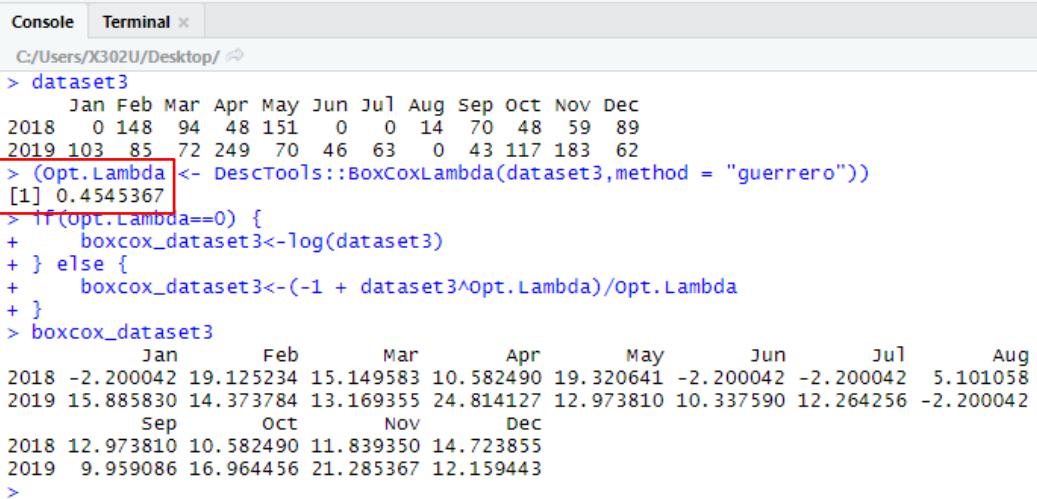


Lampiran 15 Transformasi Data Setelah *Trend Adjustment* (Kanan) Menggunakan RStudio



Lampiran 16 Transformasi Data Setelah *Seasonal Adjustment* (Kanan) Menggunakan RStudio

- **Transformasi Stasioner Box-Cox dan Differencing**



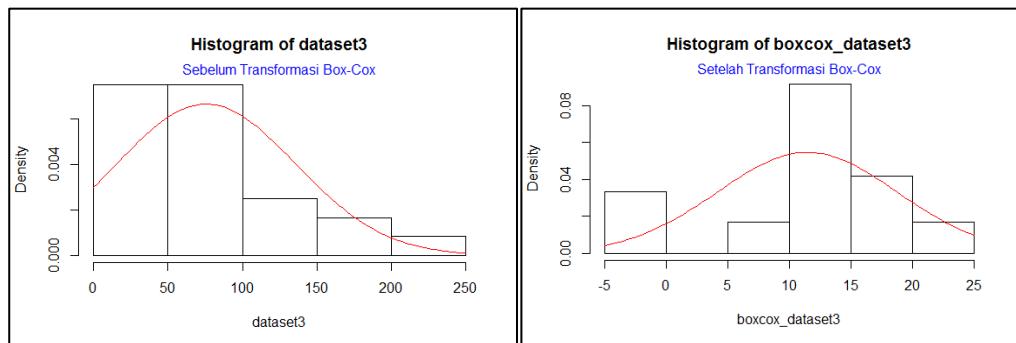
Lampiran 17 Mencari Lambda Optimal ($\lambda = 0.45$) dan Melakukan Transformasi Box-Cox pada RStudio

Console Terminal x

C:/Users/X302U/Desktop/ ↵

```
> hist(dataset3, probability = T)
> fit<-fitdistr(dataset3,"normal")
> curve(dnorm(x, fit$estimate[1], fit$estimate[2]), col = 2, add = TRUE)
> skewness(dataset3)
[1] 1.020558
>
> # Setelah Transformasi Box-Cox
> hist(boxcox_dataset3, probability = T)
> fit<-fitdistr(boxcox_dataset3,"normal")
> curve(dnorm(x, fit$estimate[1], fit$estimate[2]), col = 2, add = TRUE)
> skewness(boxcox_dataset3)
[1] -0.6188436
>
```

Lampiran 18 Skewness Transformasi Box-Cox Lebih Mendekati Nol



Lampiran 19 Transformasi Box-Cox (Kanan) mendekati Distribusi Normal

Console Terminal x

C:/Users/X302U/Desktop/ ↵

```
> dataset1
  Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018   0   0   0   0   0   2 120 351 740   0 394 669
2019 770 349 742 915 268 296 481   0 63   0 429 228
> (orde_dataset1 <- ndiffs(dataset1, alpha = 0.05, test = "adf"))
[1] 1
>
> dataset4
  Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018   0   2   20  15  24   1   5   2   20  23  10  17
2019  16   3   14   0  27   0   0   0  28  14   0   0
> (orde_dataset4 <- ndiffs(dataset4, alpha = 0.05, test = "adf"))
[1] 0
>
```

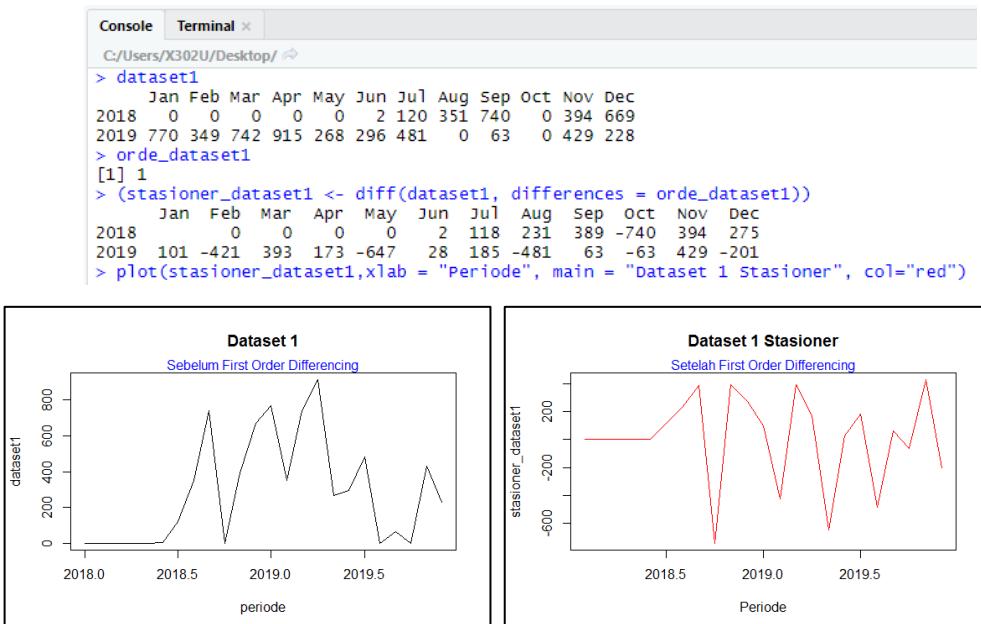
Lampiran 20 Penentuan Orde Differencing dengan Uji Augmented Dickey-Fuller Menggunakan RStudio

```

Console Terminal ×
C:/Users/X302U/Desktop/ ↵
> dataset1
  Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018   0   0   0   0   0   2 120 351 740   0 394 669
2019 770 349 742 915 268 296 481   0   63   0 429 228
> (orde_dataset1 <- ndiffs(dataset1, alpha = 0.05, test = "adf"))
[1] 1
>
> (seasonal.orde_dataset1 <- nsdiffs(dataset1, alpha = 0.05, test = "ch"))
[1] 0
>

```

Lampiran 21 Orde Seasonal Differencing Bernilai Nol



Lampiran 22 Data Menjadi Stasioner Dalam Rataan (Kanan) Setelah *First Order Differencing*

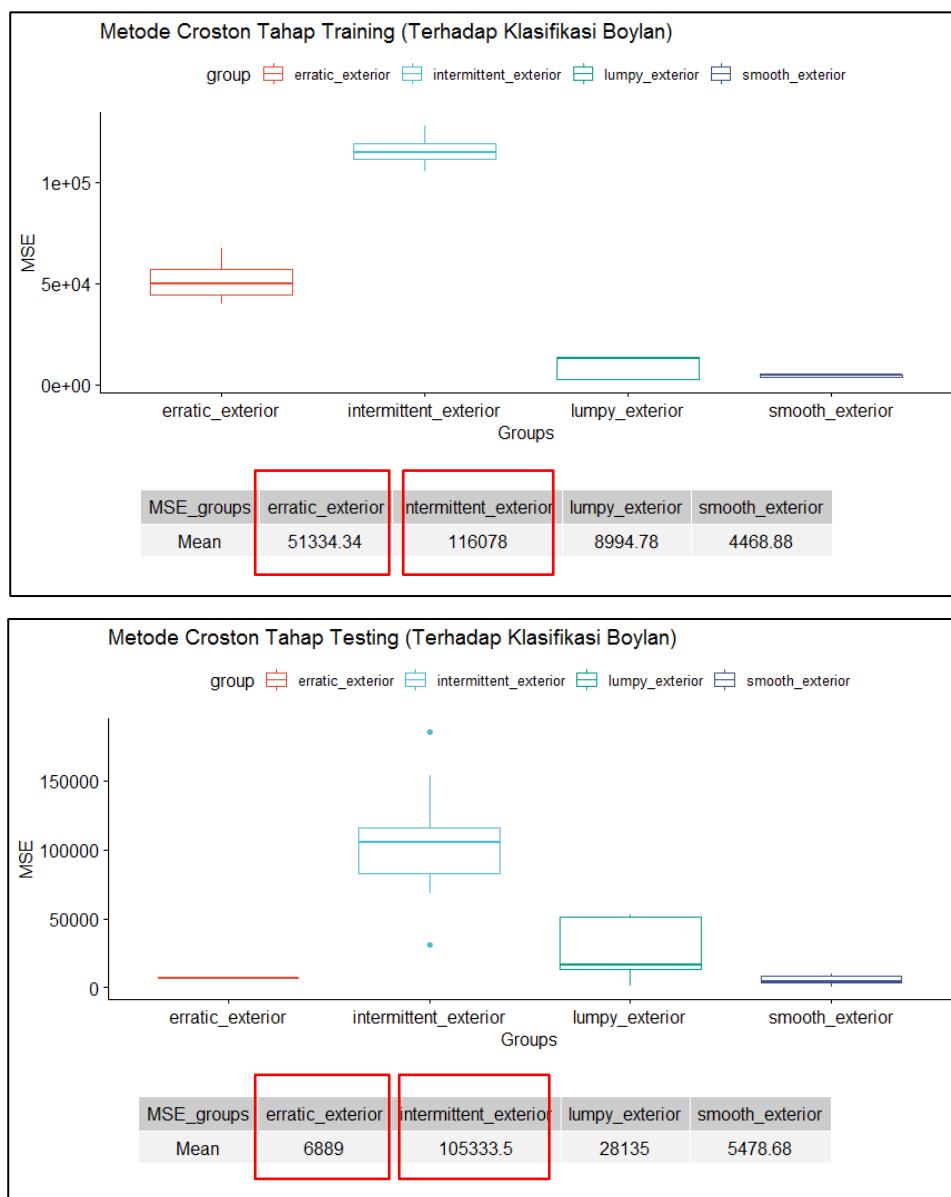
```

> # Uji p.value ADF
> # p.value <= 0.05, data is stasionary process
> # p.value > 0.05, data is random walk
>
> # Sebelum first order differencing
> adf.test(dataset1)$p.value
[1] 0.8620563
> # Setelah first order differencing
> adf.test(stasioner_dataset1)$p.value
[1] 0.0171227

```

Lampiran 23 Time-Series Sudah Stasioner Dalam Rataan Pada Uji Augmented Dickey-Fuller Test

- Hasil uji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja model peramalan Croston, SBA, dan TSB



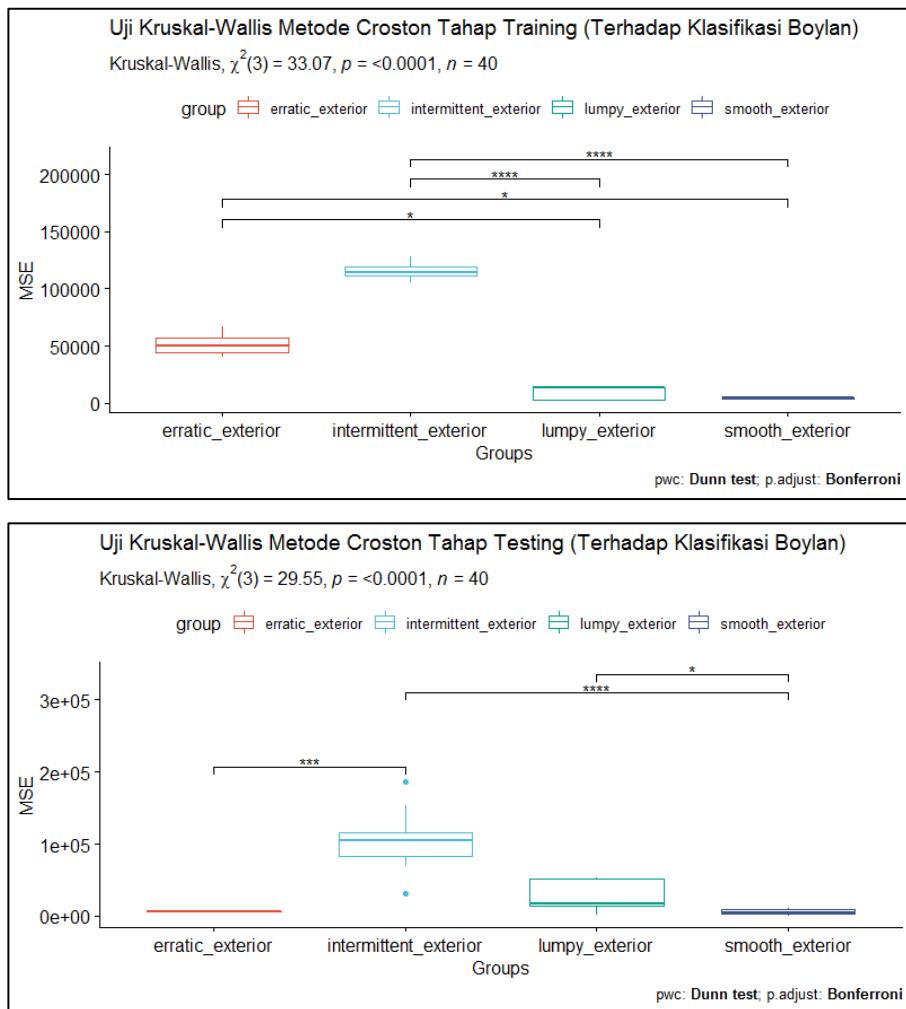
Lampiran 24 Boxplot MSE *Training* dan *Testing* Metode Croston Terhadap Berbagai Pola Permintaan Boylan

```

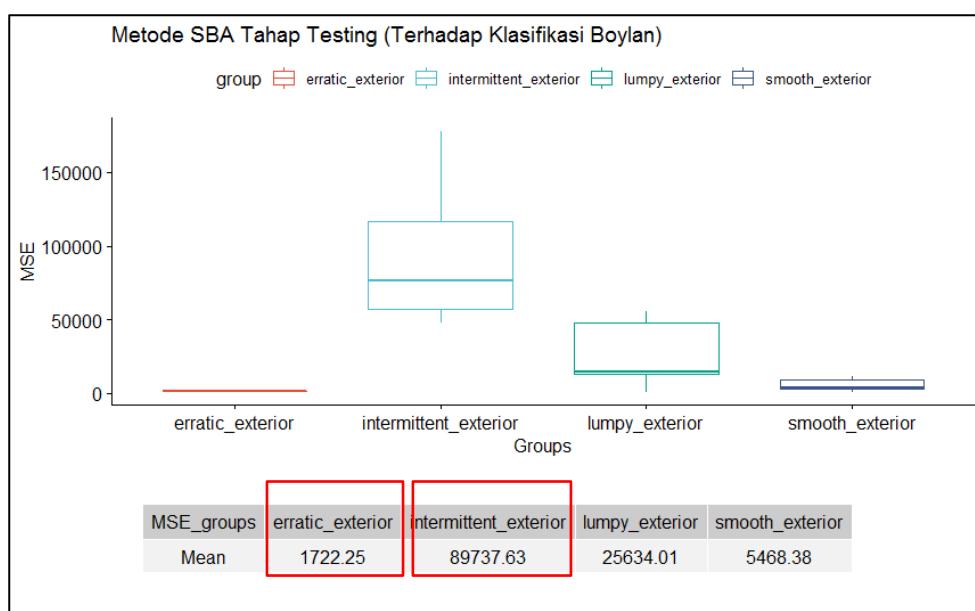
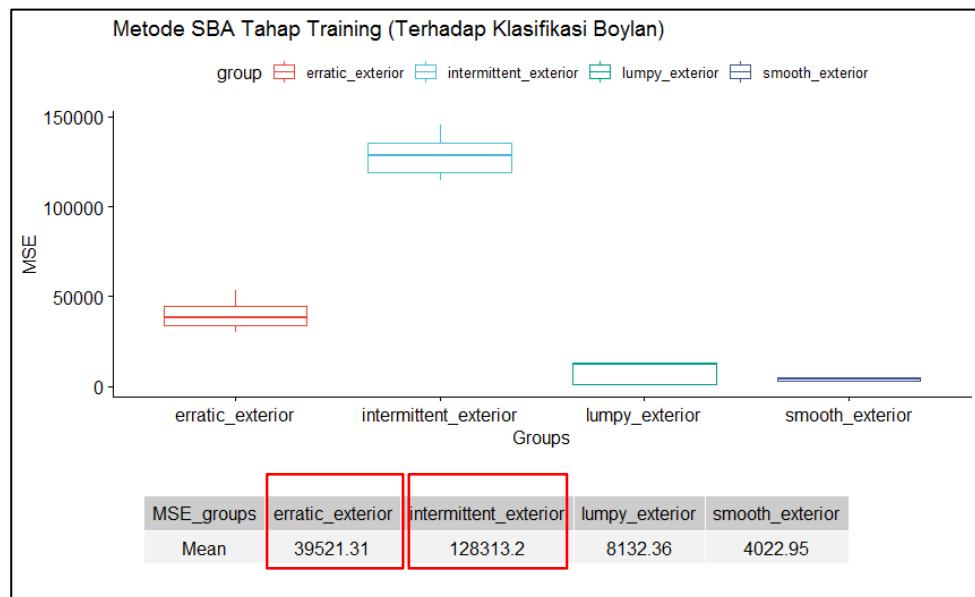
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = Varian Tidak Homogen)
> # MSE Training (Klasifikasi Boylan)
> leveneTest(crost_all_boylan_train$MSE, crost_all_boylan_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3  4.0281 0.01436 *
            36
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(crost_all_boylan_test$MSE, crost_all_boylan_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3  6.4367 0.001329 **
            36
---
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>

```

Lampiran 25 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*
Metode Croston Terhadap Berbagai Pola Permintaan Boylan



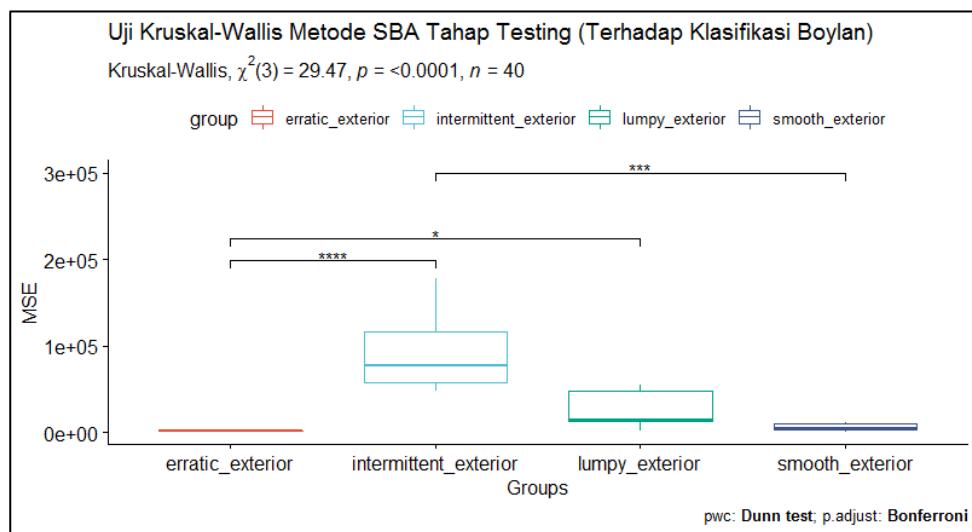
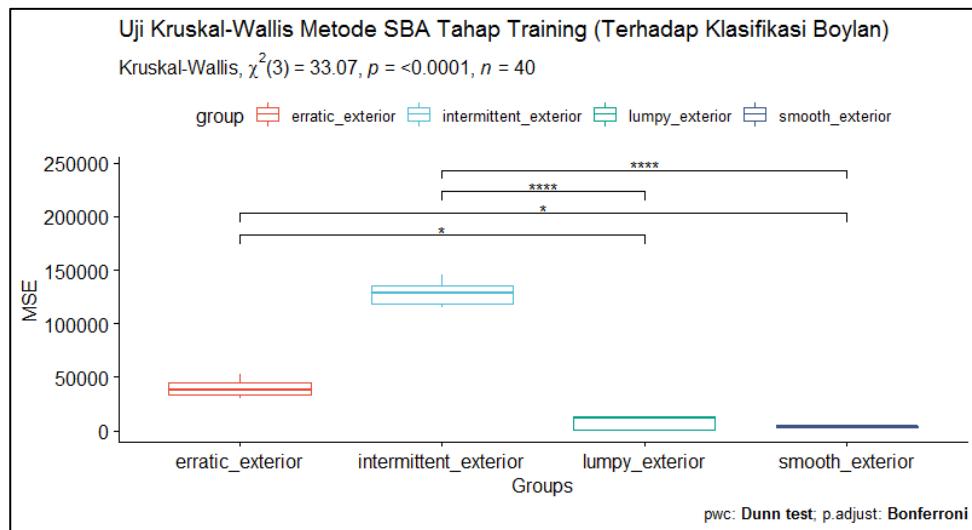
Lampiran 26 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode Croston



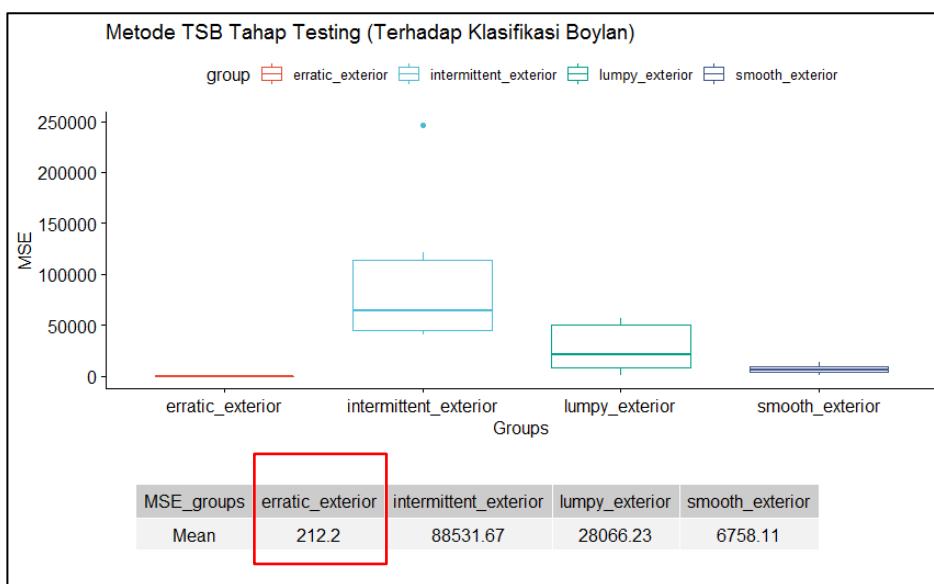
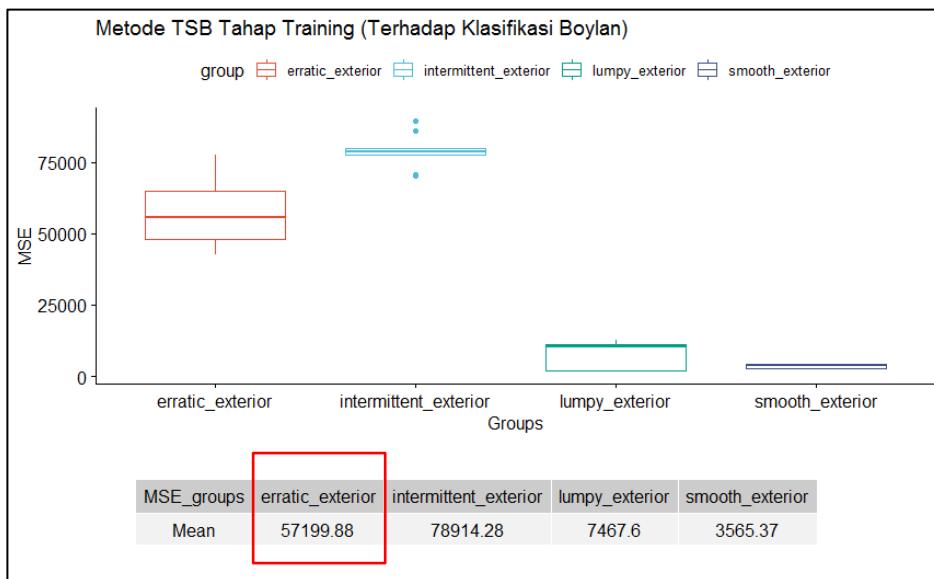
Lampiran 27 Boxplot MSE *Training* dan *Testing* Metode SBA Terhadap Berbagai Pola Permintaan Boylan

```
Console | Terminal x | Jobs x
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Klasifikasi Boylan)
> leveneTest(sba_all_boyilan_train$MSE, sba_all_boyilan_train$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  df F value Pr(>F)
group  3  4.3448 0.01034 *
  36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(sba_all_boyilan_test$MSE, sba_all_boyilan_test$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  df F value Pr(>F)
group  3  9.4773 9.436e-05 ***
  36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 28 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*
Metode SBA Terhadap Berbagai Pola Permintaan Boylan



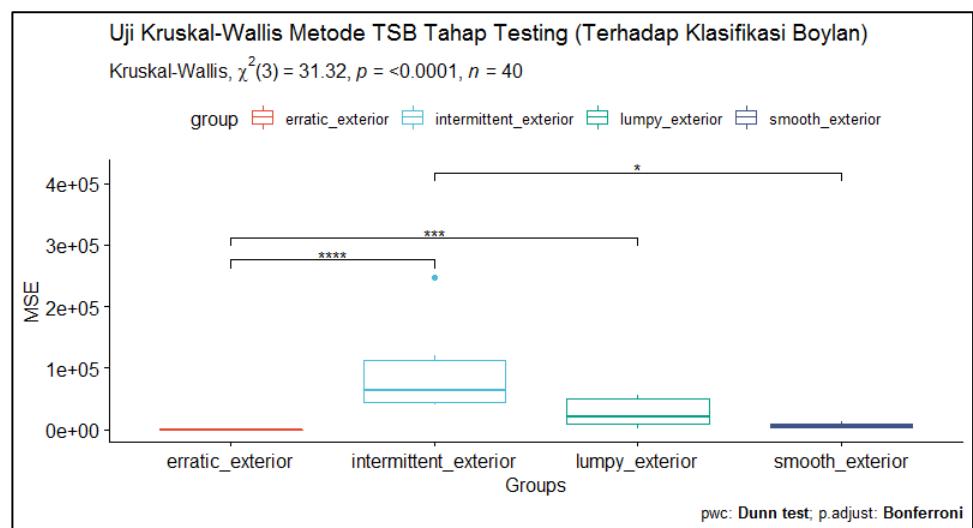
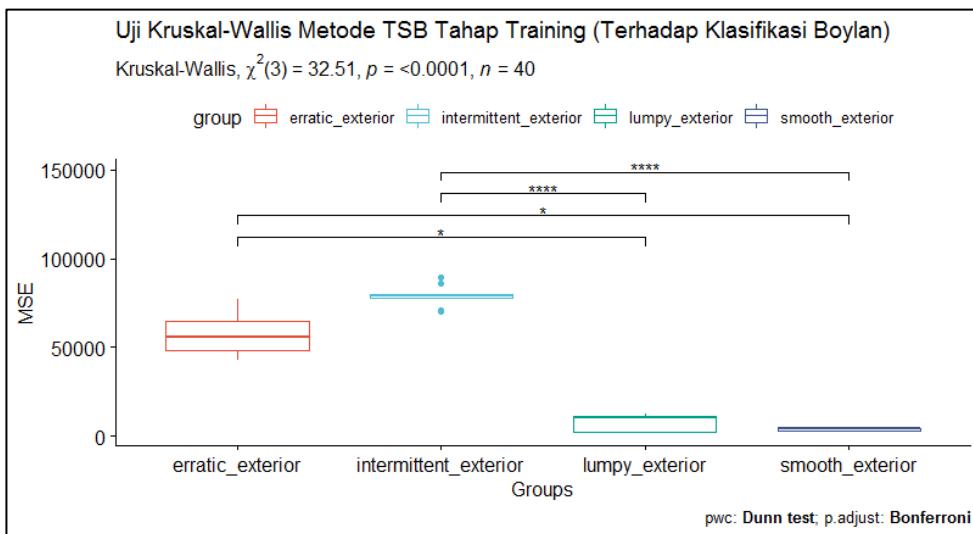
Lampiran 29 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode SBA



Lampiran 30 Boxplot MSE *Training* dan *Testing* Metode TSB Terhadap Berbagai Pola Permintaan Boylan

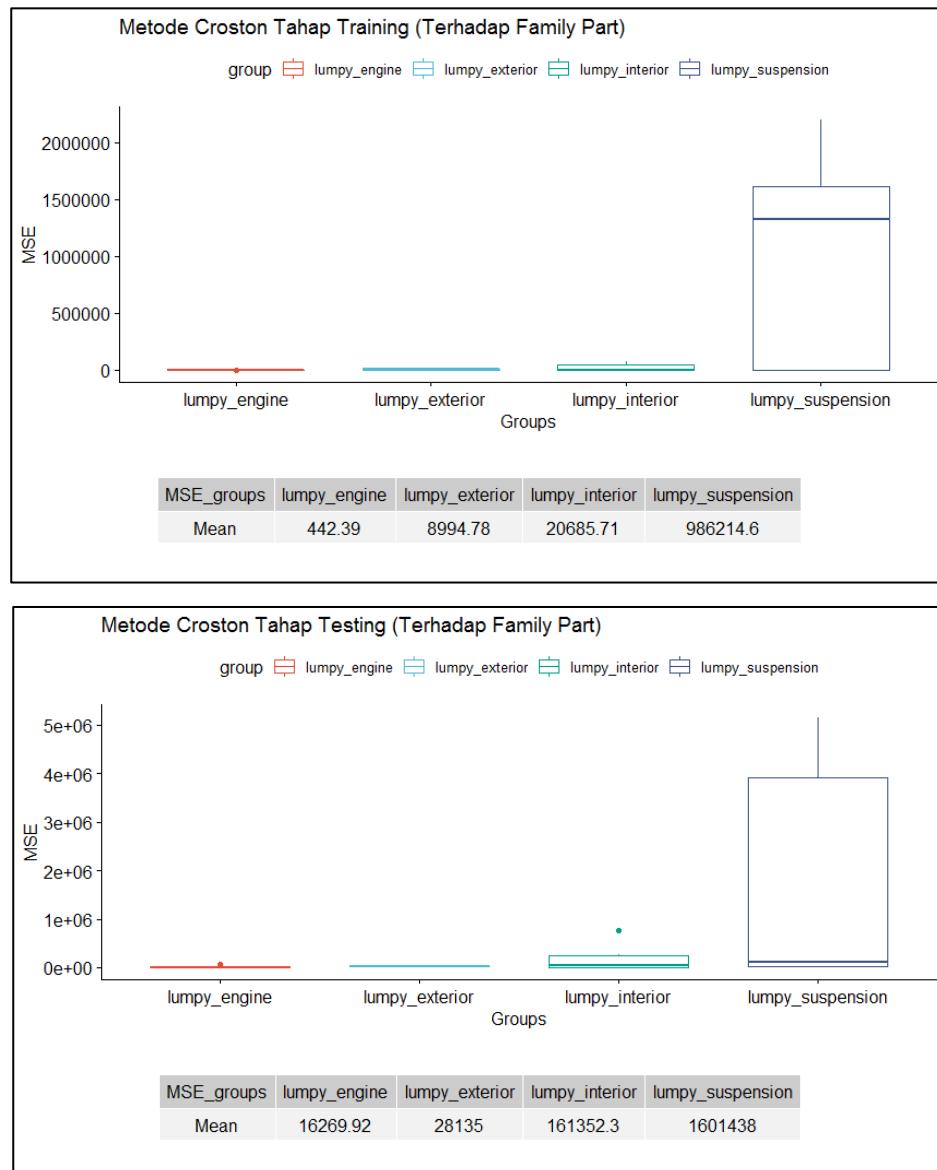
```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Klasifikasi Boylan)
> leveneTest(tsb_all_boylan_train$MSE, tsb_all_boylan_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  3  7.3556 0.0005744 ***
               36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(tsb_all_boylan_test$MSE, tsb_all_boylan_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  3  6.0345 0.00194 **
               36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 31 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*



Lampiran 32 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode TSB

- Hasil uji pengaruh perbedaan *family part* terhadap kinerja model peramalan Croston, SBA, dan TSB



Lampiran 33 Boxplot MSE *Training* dan *Testing* Metode Croston Terhadap Berbagai *Family Part*

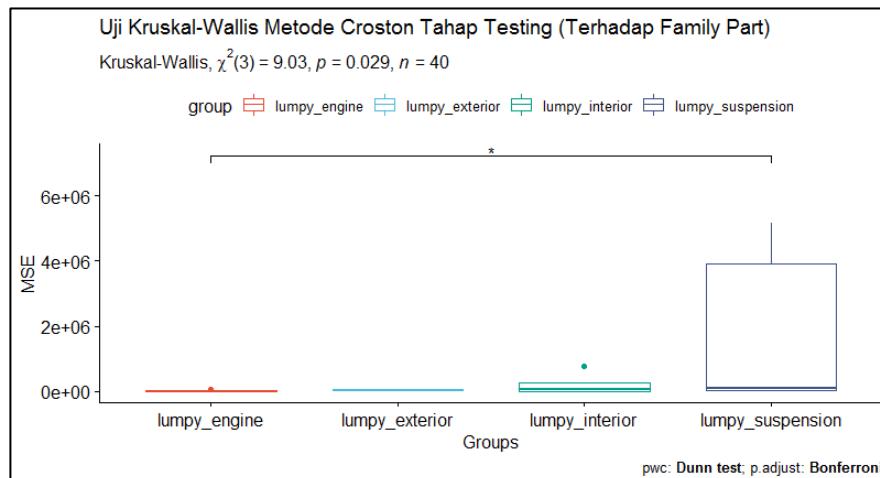
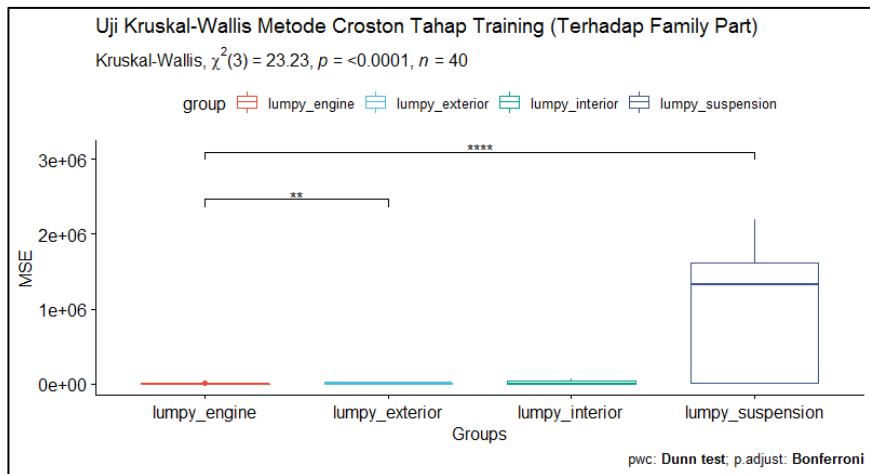
```

Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Family Part)
> leveneTest(crost_all_family_train$MSE, crost_all_family_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 15.652 1.117e-06 ***
36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Family Part)
> leveneTest(crost_all_family_test$MSE, crost_all_family_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 3.9473 0.01563 *
36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |

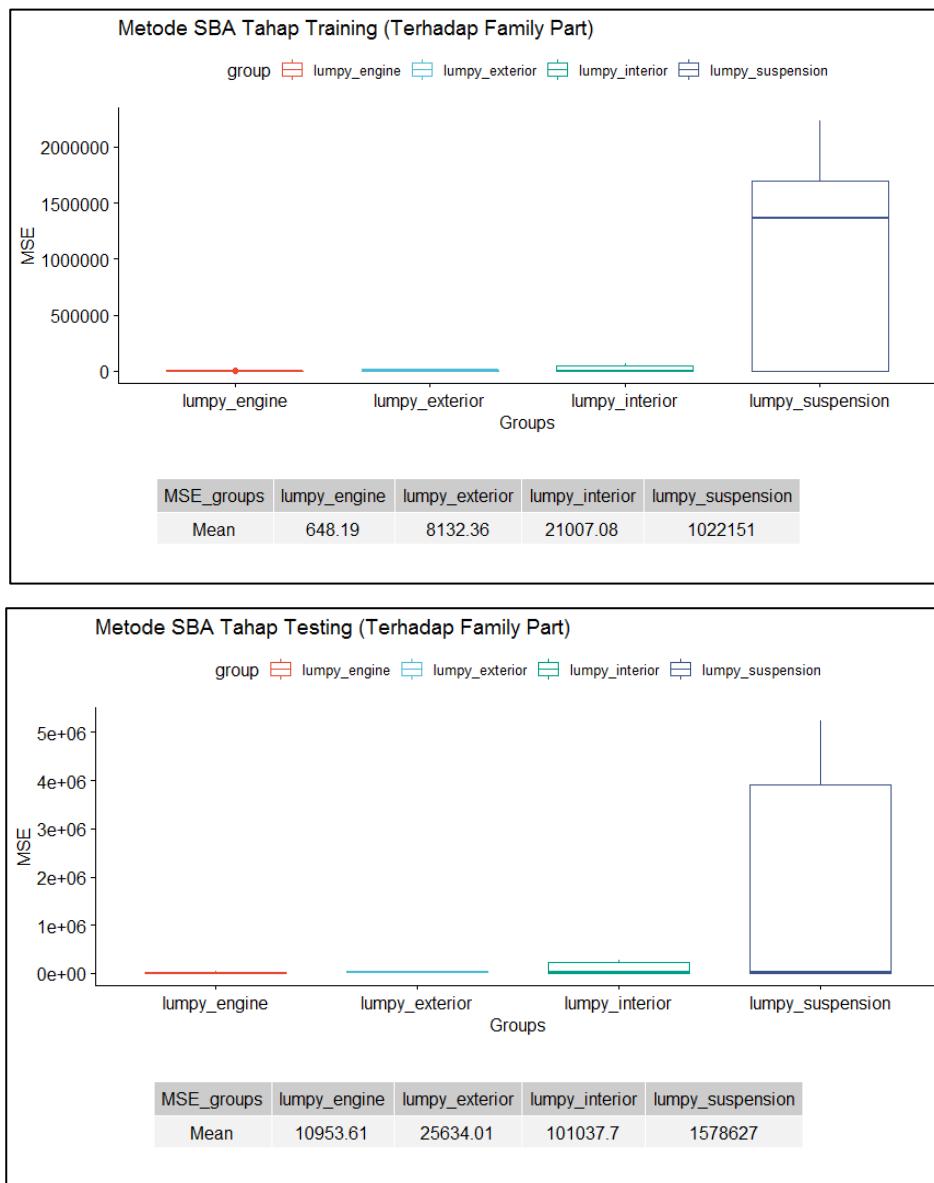
```

Lampiran 34 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*

Metode Croston Terhadap Berbagai *Family Part*



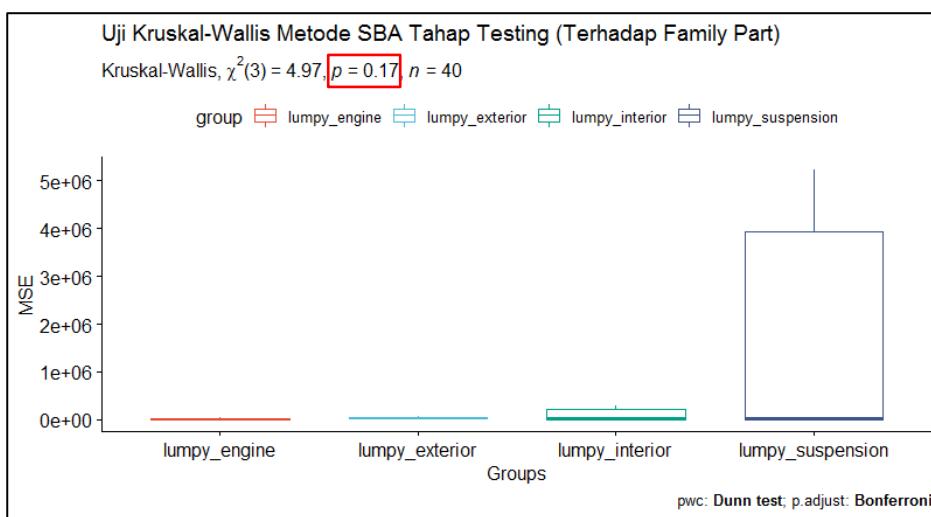
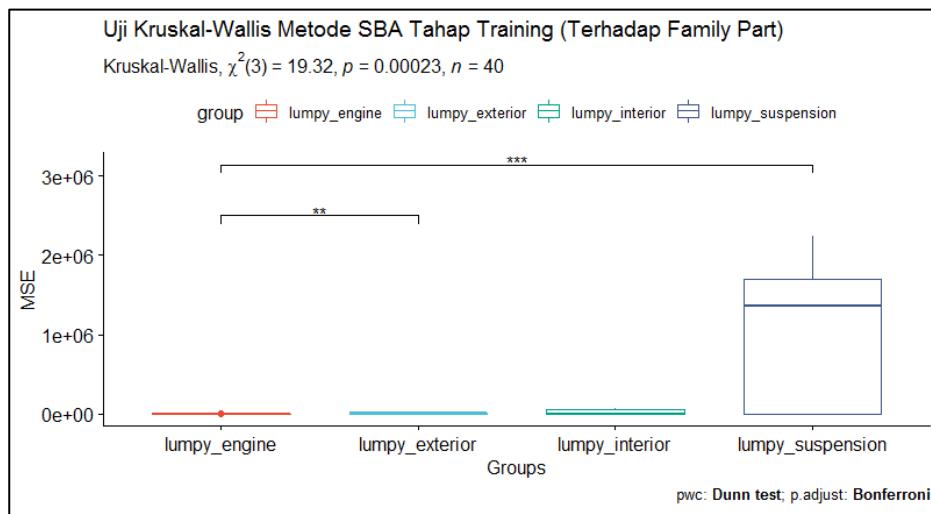
Lampiran 35 Uji Signifikansi Pengaruh *Family Part* Terhadap Metode Croston



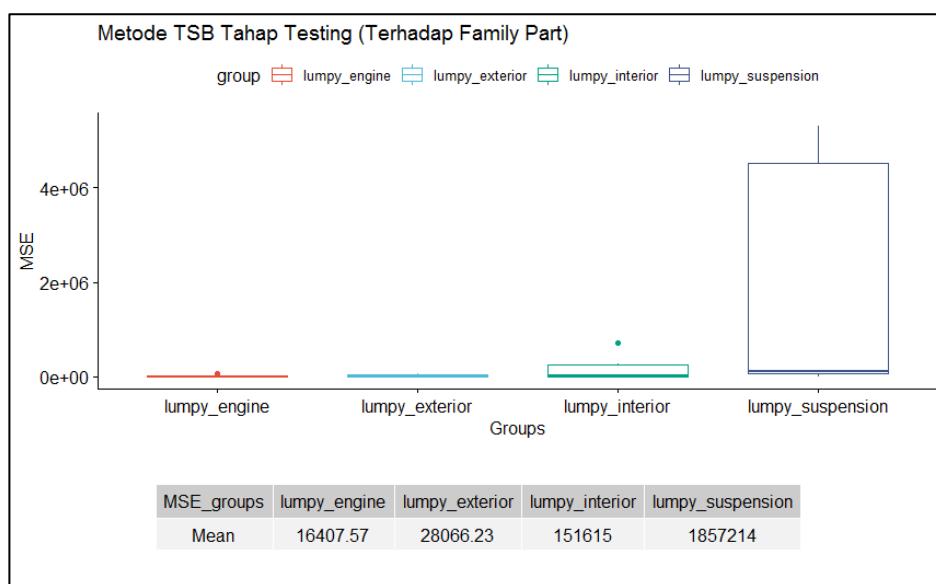
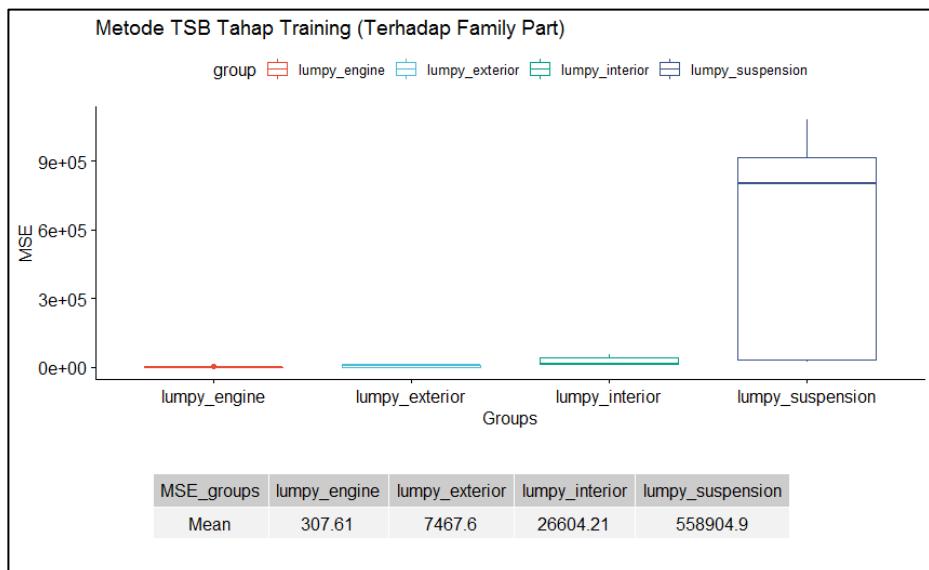
Lampiran 36 Boxplot MSE *Training* dan *Testing* Metode SBA Terhadap Berbagai *Family Part*

```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = Varian Tidak Homogen)
> # MSE Training (Family Part)
> leveneTest(sba_all_family_train$MSE, sba_all_family_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 17.287 4.048e-07 ***
36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Family Part)
> leveneTest(sba_all_family_test$MSE, sba_all_family_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 3.7414 0.01943 *
36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 37 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*
Metode SBA Terhadap Berbagai *Family Part*



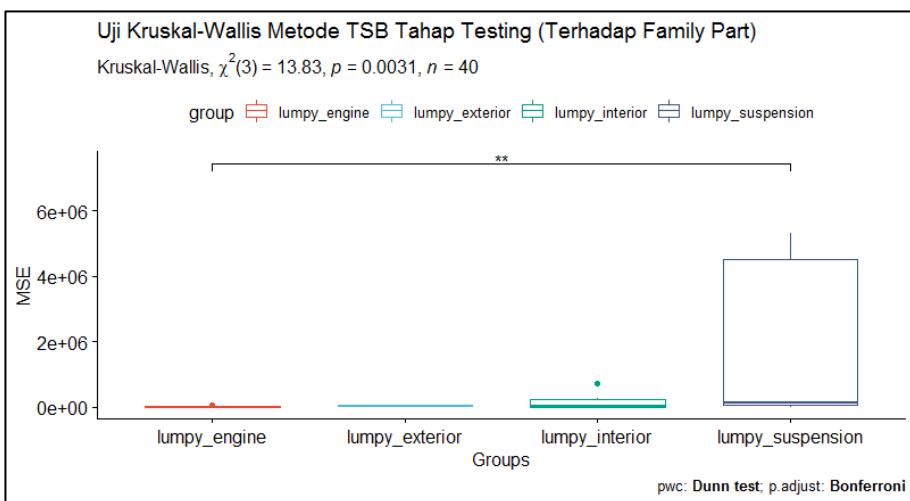
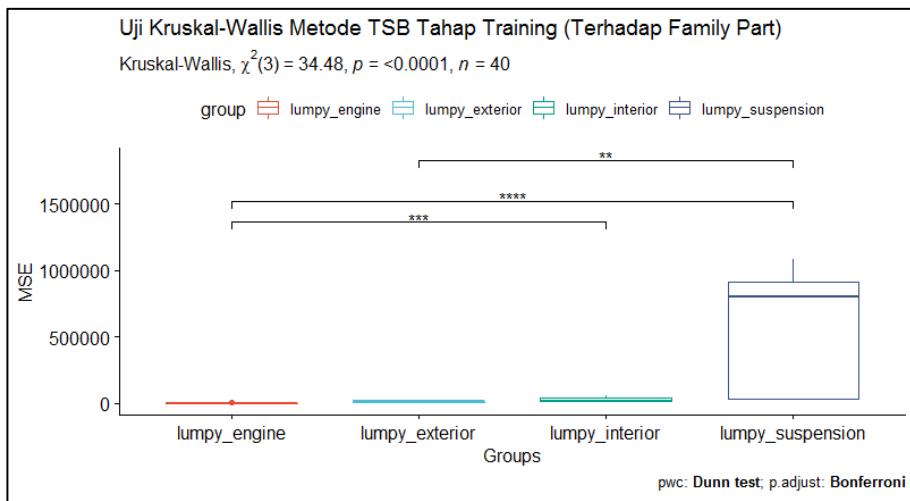
Lampiran 38 Uji Signifikansi Pengaruh *Family Part* Terhadap Metode SBA



Lampiran 39 Boxplot MSE *Training* dan *Testing* Metode TSB Terhadap Berbagai *Family Part*

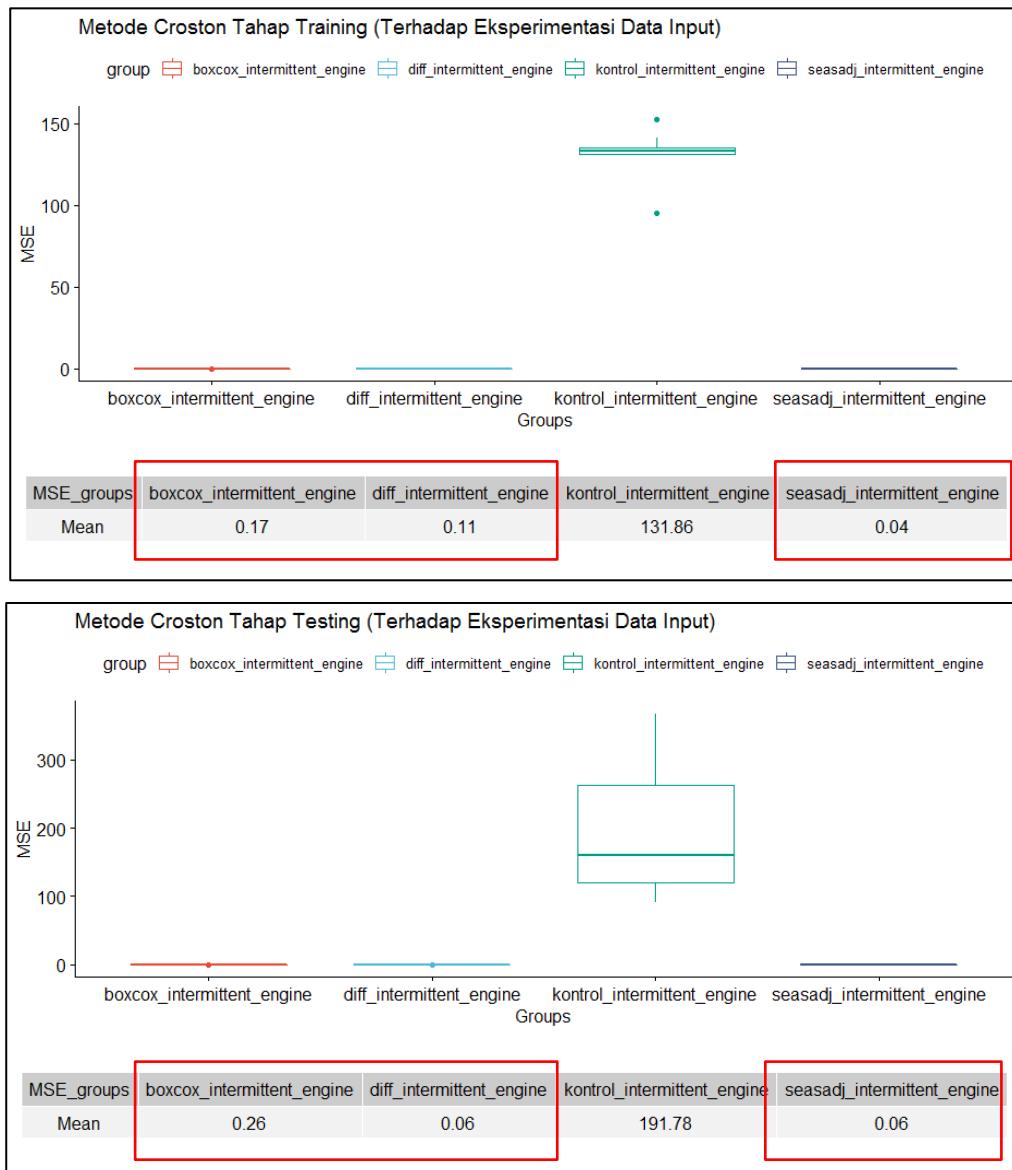
```
Console | Terminal x | Jobs x
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Family Part)
> leveneTest(tsb_all_family_train$MSE, tsb_all_family_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3 11.493 1.966e-05 ***
               36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Family Part)
> leveneTest(tsb_all_family_test$MSE, tsb_all_family_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3 5.1508 0.004576 **
               36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 40 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*
Metode TSB Terhadap Berbagai *Family Part*



Lampiran 41 Uji Signifikansi Pengaruh *Family Part* Terhadap Metode TSB

- **Hasil uji pengaruh perubahan-perubahan pada *input* data permintaan yang sama terhadap kinerja model peramalan Croston, SBA, dan TSB**



Lampiran 42 Boxplot MSE *Training* dan *Testing* Metode Croston Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

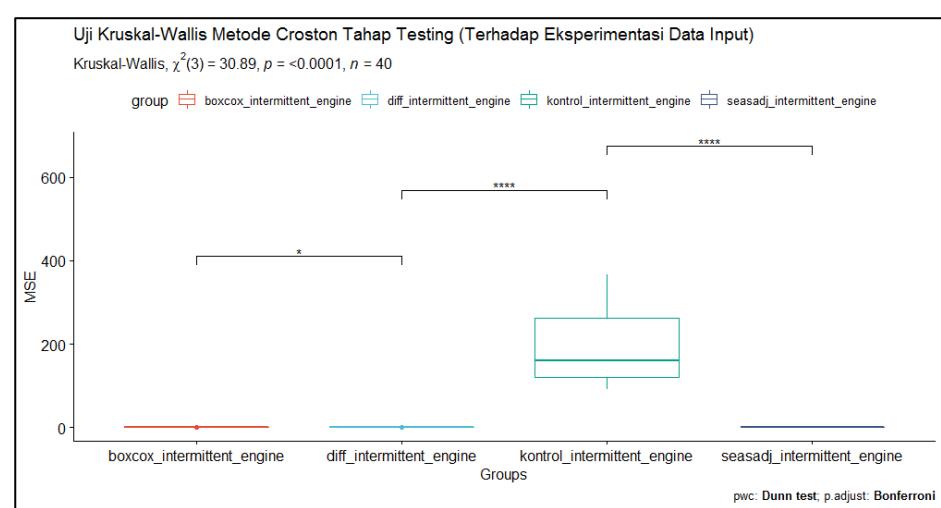
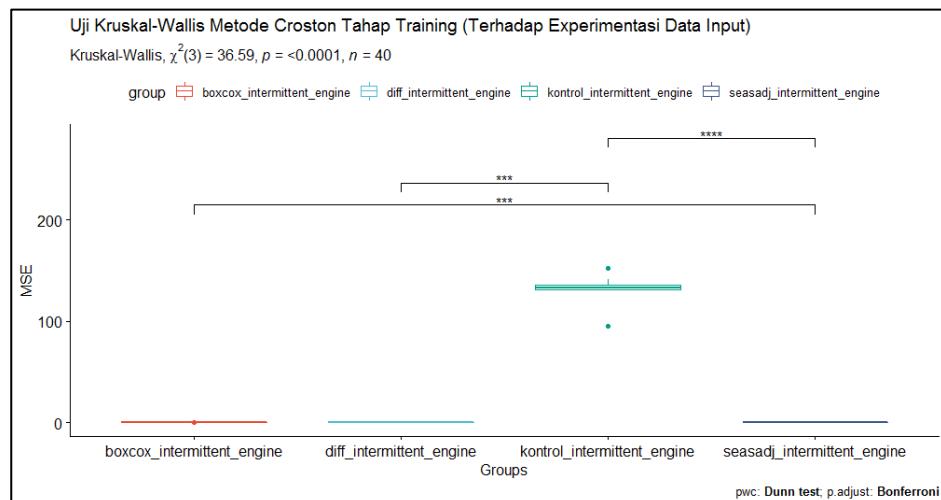
```

Console Terminal x Jobs x
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Eksperimentasi Data Input)
> leveneTest(crost_all_experiment_train$MSE, crost_all_experiment_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  3.9064 0.01632 *
  36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Eksperimentasi Data Input)
> leveneTest(crost_all_experiment_test$MSE, crost_all_experiment_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 11.151 2.542e-05 ***
  36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |

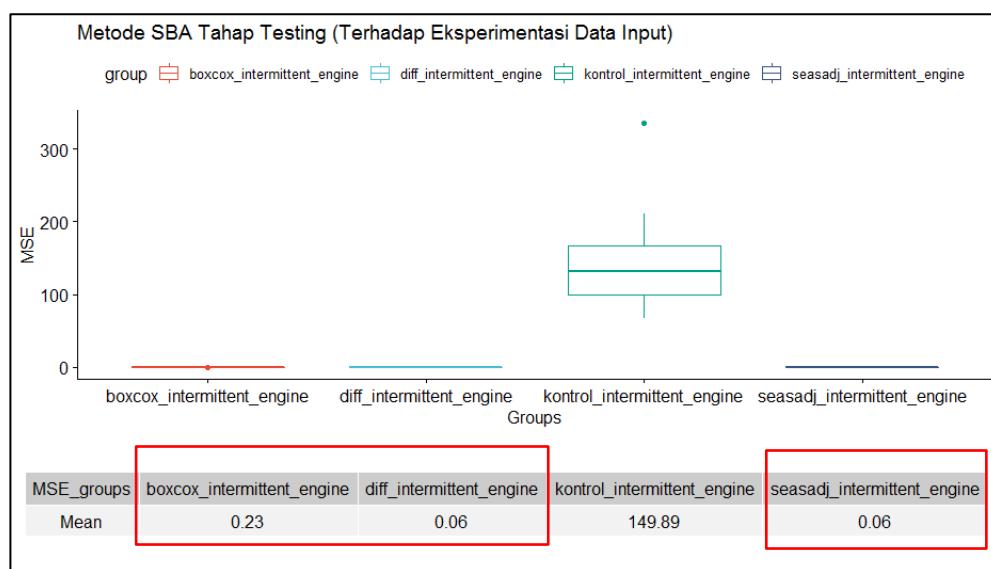
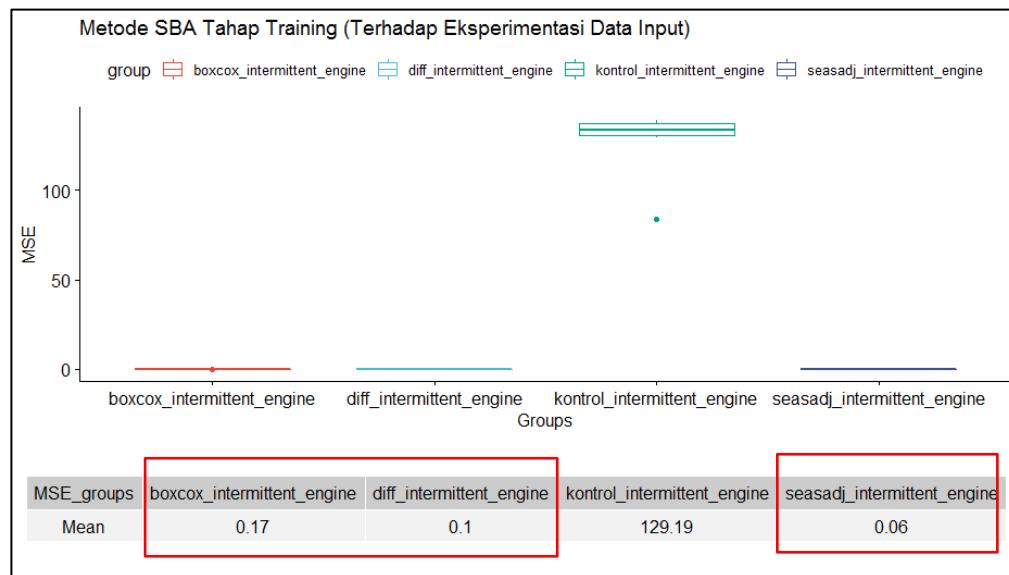
```

Lampiran 43 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*

Metode Croston Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama



Lampiran 44 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode Croston



Lampiran 45 Boxplot MSE *Training* dan *Testing* Metode SBA Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

```

Console Terminal Jobs
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Eksperimentasi Data Input)
> LeveneTest(sba_all_experiment_train$MSE, sba_all_experiment_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3  2.7298 0.05812 .
  36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Eksperimentasi Data Input)
> LeveneTest(sba_all_experiment_test$MSE, sba_all_experiment_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3  7.1338 0.000701 ***
  36
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |

```

Lampiran 46 Uji Asumsi Anova Homogenitas Varian MSE *Training* dan *Testing*
Metode SBA Terhadap Perubahan-perubahan akibat eksperimentasi
pada input peramalan yang sama

```

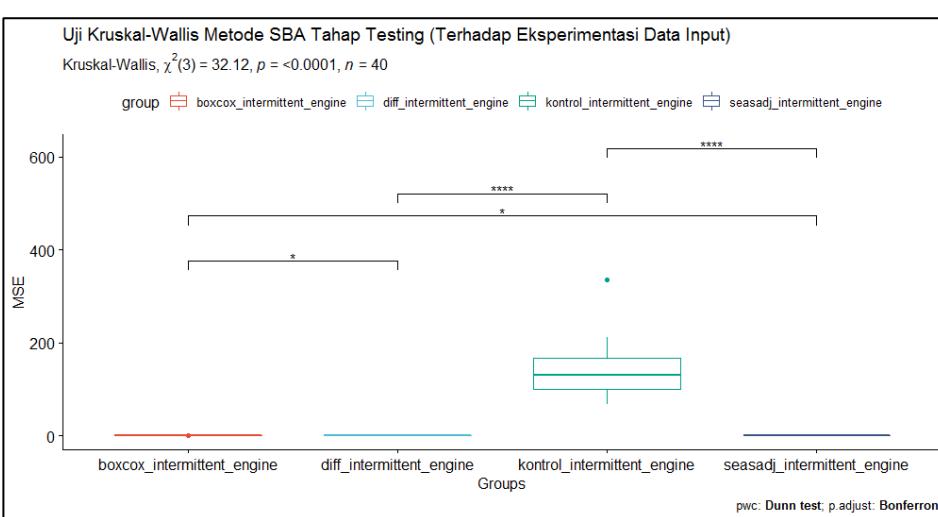
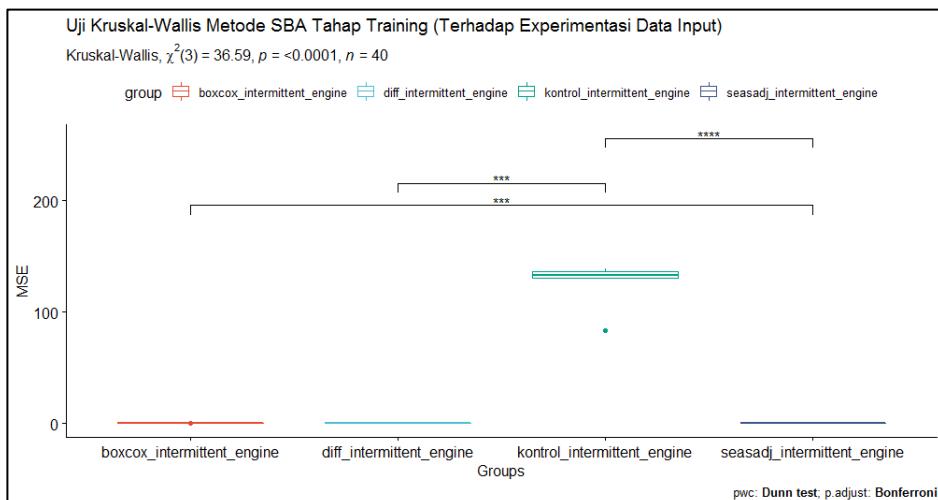
Console Terminal Jobs
C:/Users/X302U/Desktop/ ↵
> # Uji Normalitas Residual Shapiro-wilk (p value < 0.05 = Residual Tidak
> # Berdistribusi Normal)
> # MSE Training (Eksperimentasi Data Input)
> linmod<-lm(MSE~group,data=sba_all_experiment_train)
> shapiro.test(linmod$residuals)

  shapiro-wilk normality test

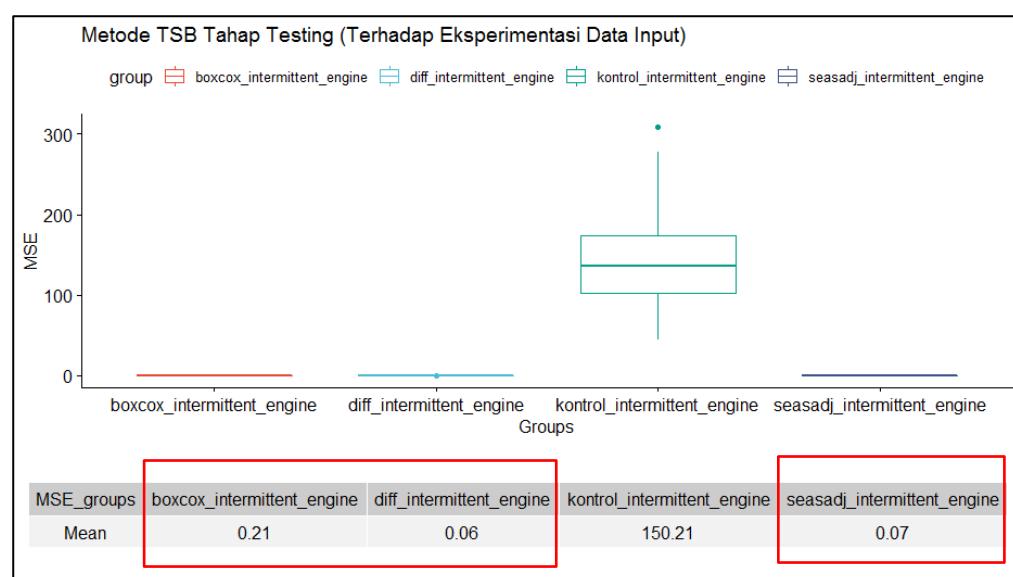
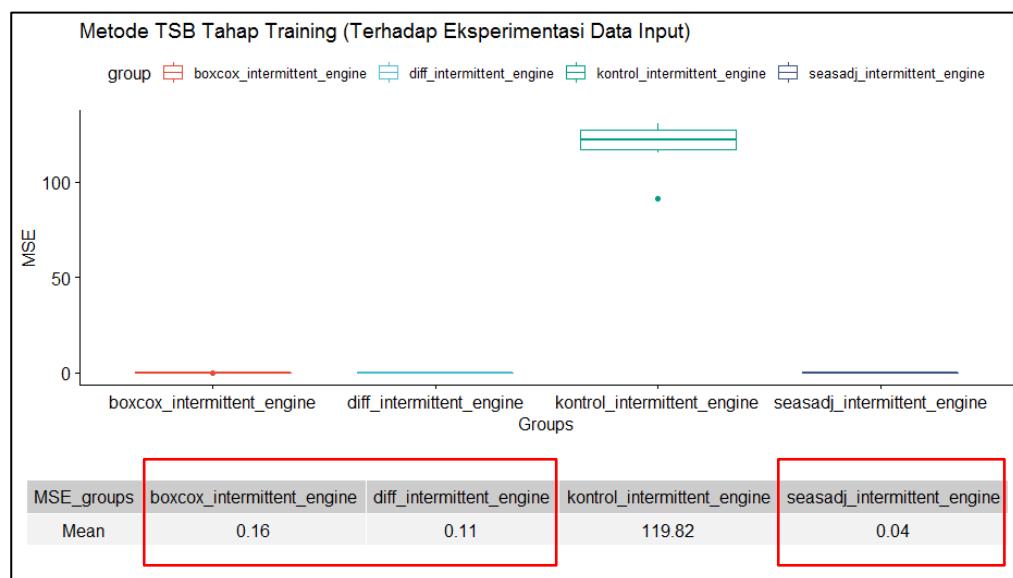
data: linmod$residuals
W = 0.349, p-value = 4.109e-12

```

Lampiran 47 Uji Asumsi Anova Normalitas Residual MSE *Training* Metode SBA
Terhadap Perubahan-perubahan akibat eksperimentasi pada input
peramalan yang sama



Lampiran 48 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode SBA

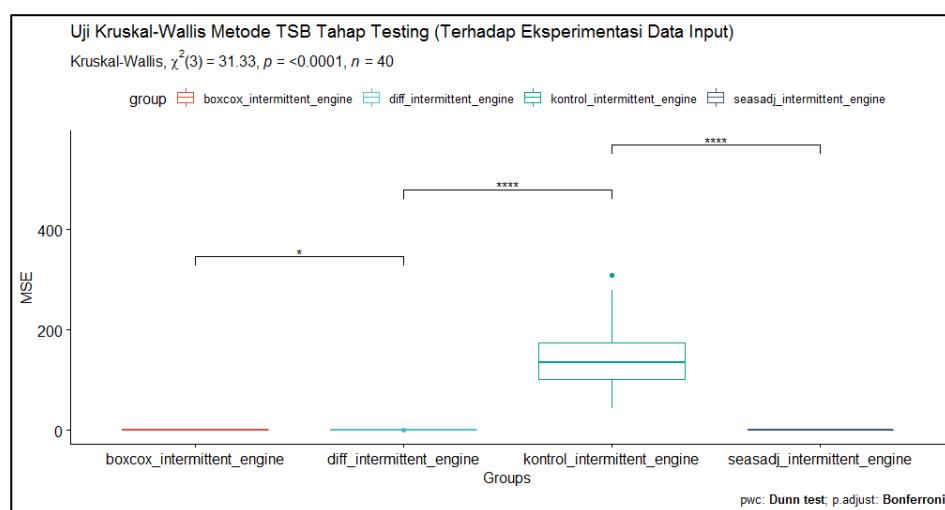
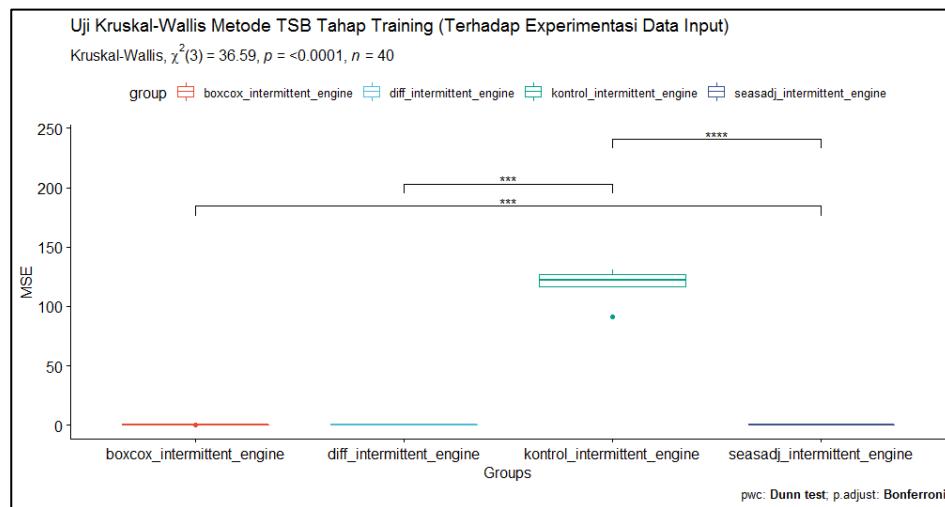


Lampiran 49 Boxplot MSE *Training* dan *Testing* Metode TSB Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Eksperimentasi Data Input)
> leveneTest(tsb_all_experiment_train$MSE, tsb_all_experiment_train$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  3  7.2177 0.0006499 ***
               36
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Eksperimentasi Data Input)
> leveneTest(tsb_all_experiment_test$MSE, tsb_all_experiment_test$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  3 10.426 4.435e-05 ***
               36
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
```

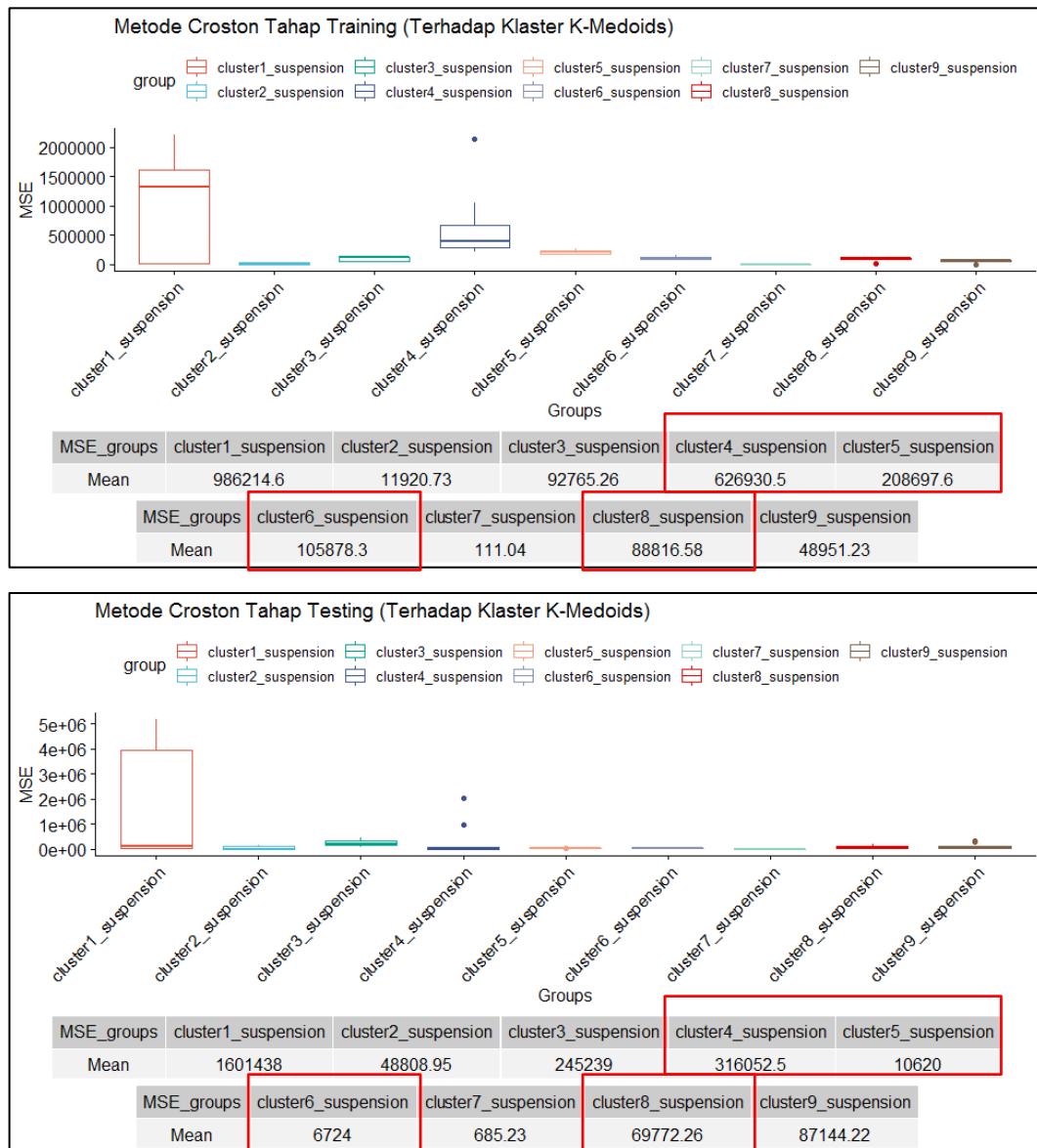
Lampiran 50 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode TSB

Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama



Lampiran 51 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode TSB

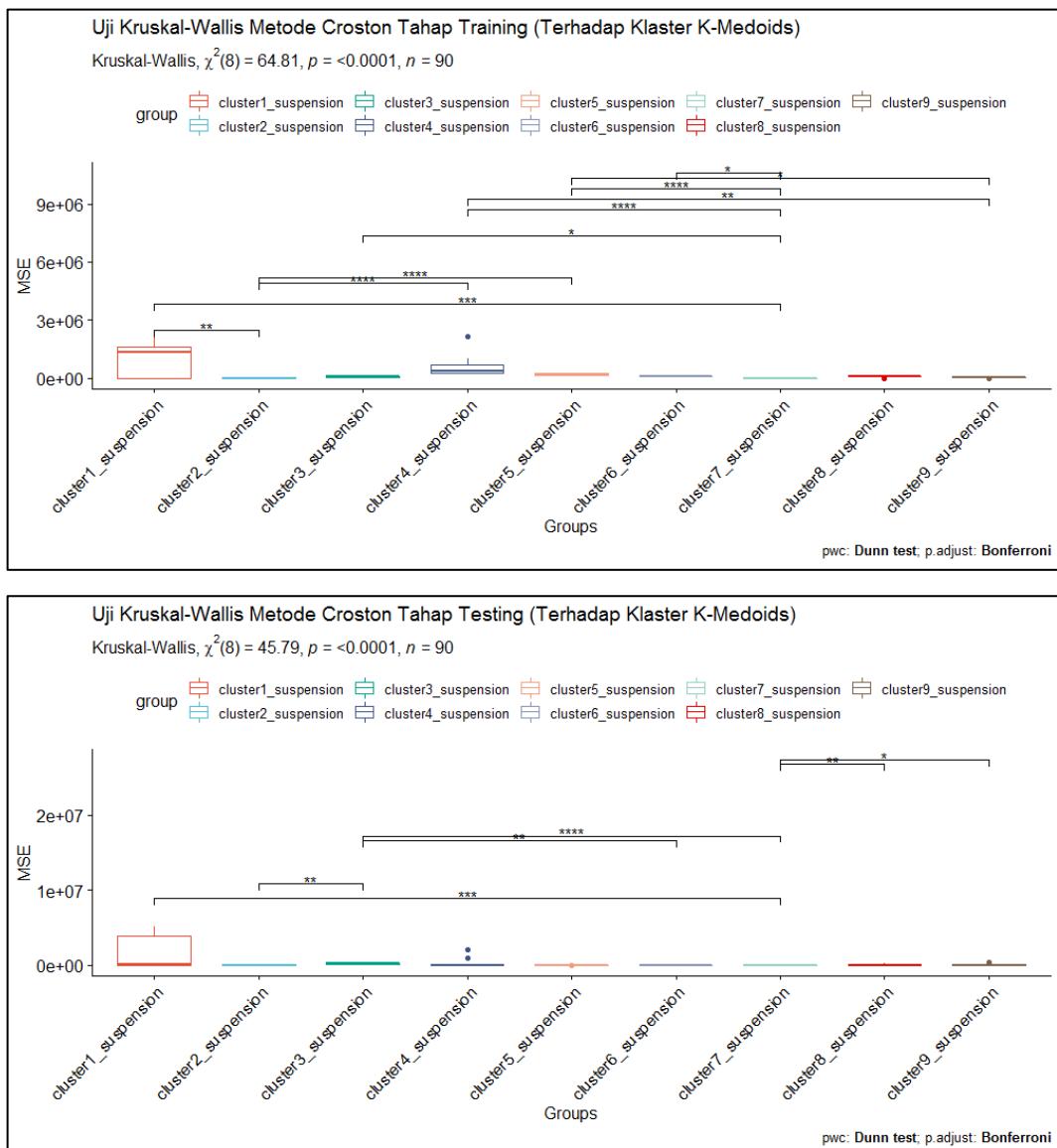
- Hasil uji pengaruh perbedaan pengelompokan *part* hasil K-Medoids *Clustering* terhadap kinerja model peramalan Croston, SBA, dan TSB



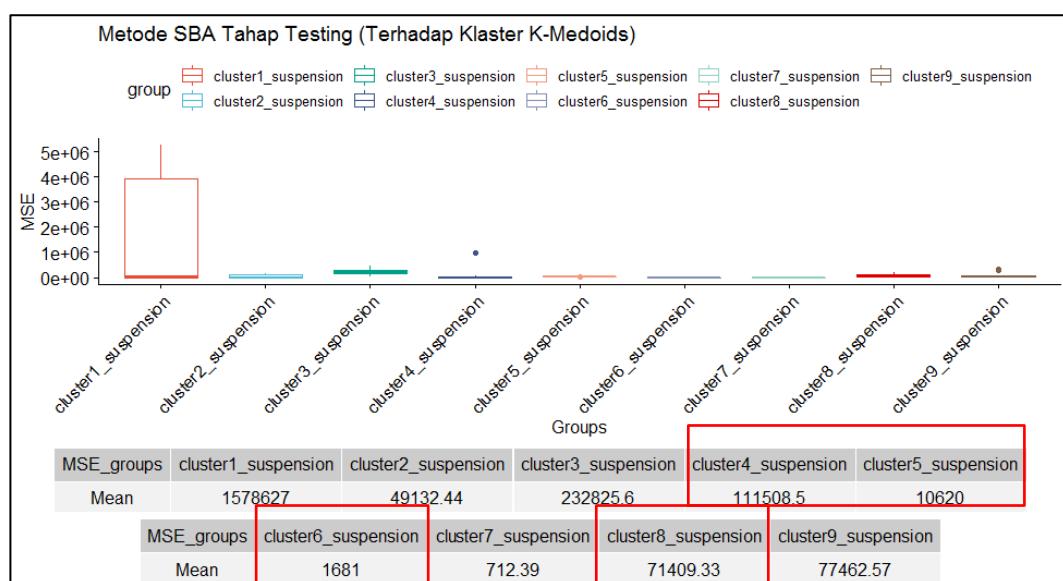
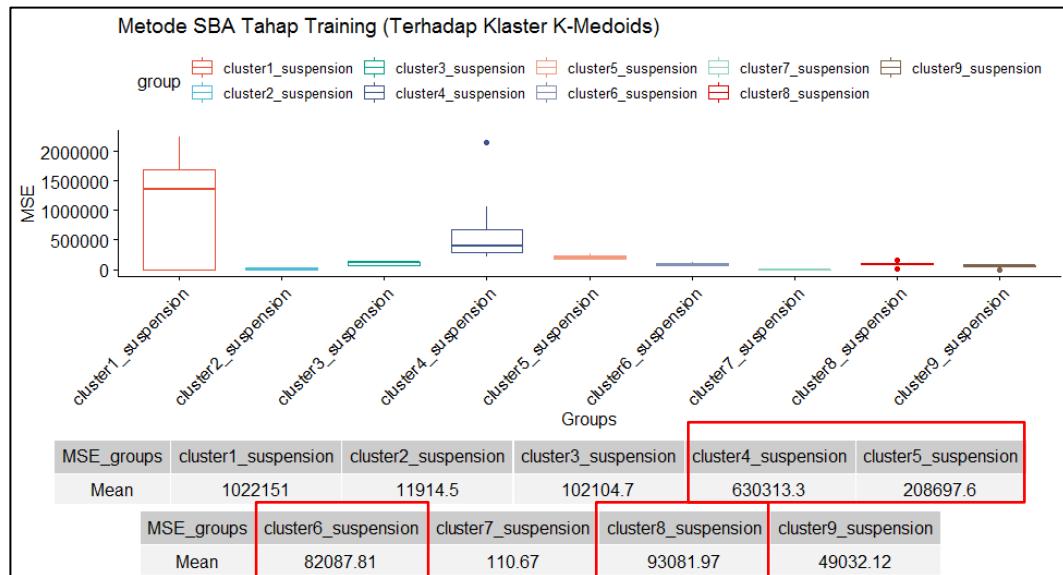
Lampiran 52 Boxplot MSE *Training* dan *Testing* Metode Croston Terhadap Klaster K-Medoids

```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Klaster K-Medoids)
> leveneTest(crost_all_cluster_train$MSE, crost_all_cluster_train$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
    Df F value    Pr(>F)
group  8  9.0506 8.536e-09 ***
               81
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klaster K-Medoids)
> leveneTest(crost_all_cluster_test$MSE, crost_all_cluster_test$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
    Df F value    Pr(>F)
group  8  3.7197 0.0009519 ***
               81
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 53 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode Croston Terhadap Klaster K-Medoids



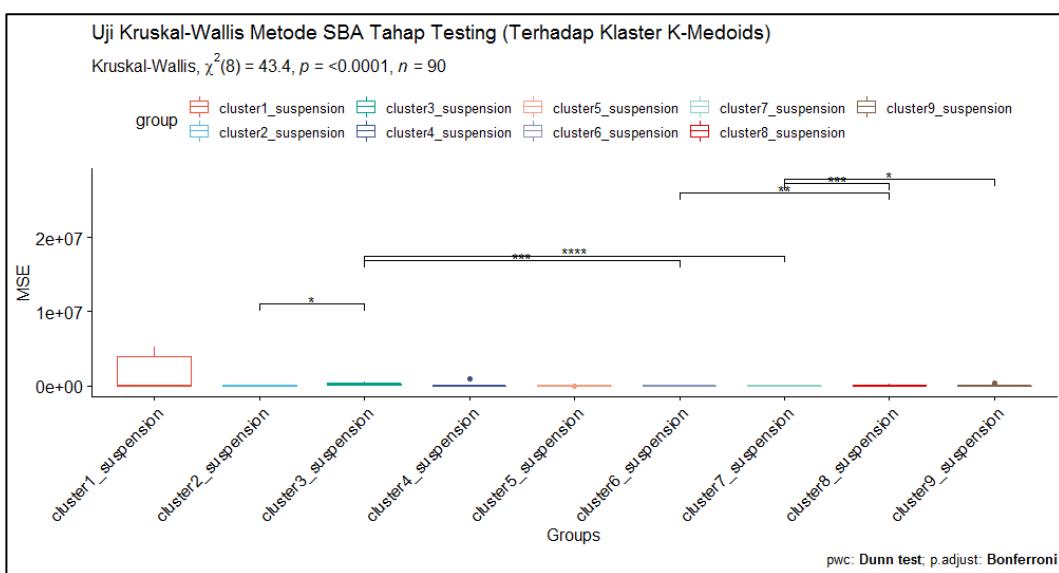
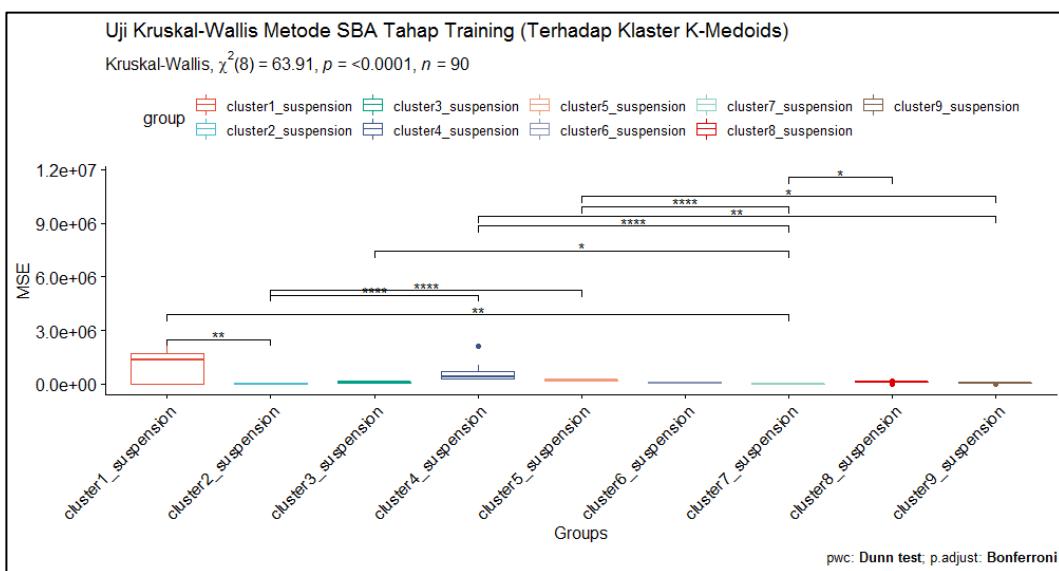
Lampiran 54 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode Croston



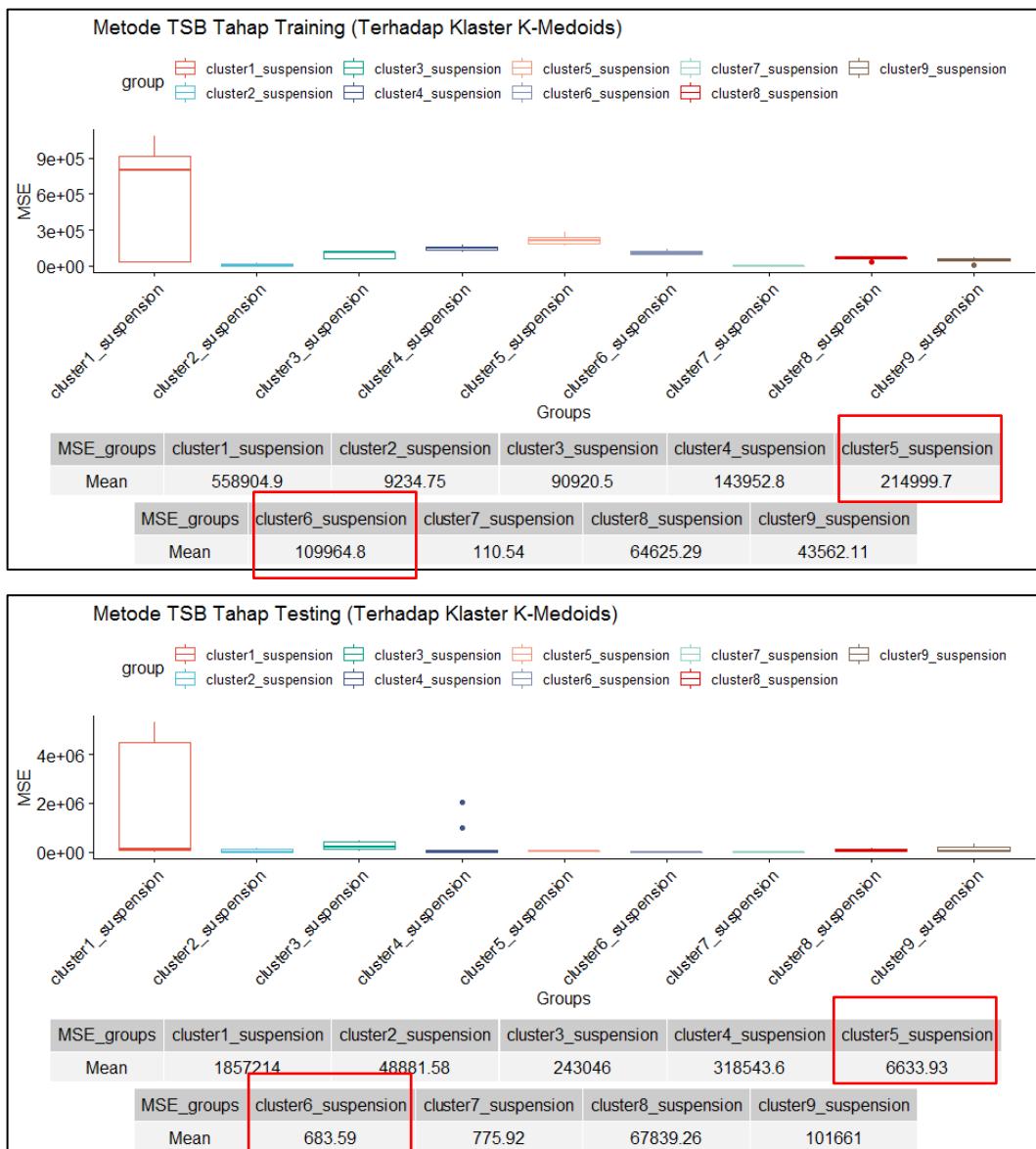
Lampiran 55 Boxplot MSE *Training* dan *Testing* Metode SBA Terhadap Klaster K-Medoids

```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = Varian Tidak Homogen)
> # MSE Training (Klaster K-Medoids)
> leveneTest(sba_all_cluster_train$MSE, sba_all_cluster_train$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  8  9.9112 1.648e-09 ***
81
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klaster K-Medoids)
> leveneTest(sba_all_cluster_test$MSE, sba_all_cluster_test$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  8  3.6582 0.001103 **
81
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 56 Uji Asumsi Anova Homogenitas Varian MSE Training Metode SBA
Terhadap Klaster K-Medoids



Lampiran 57 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode SBA



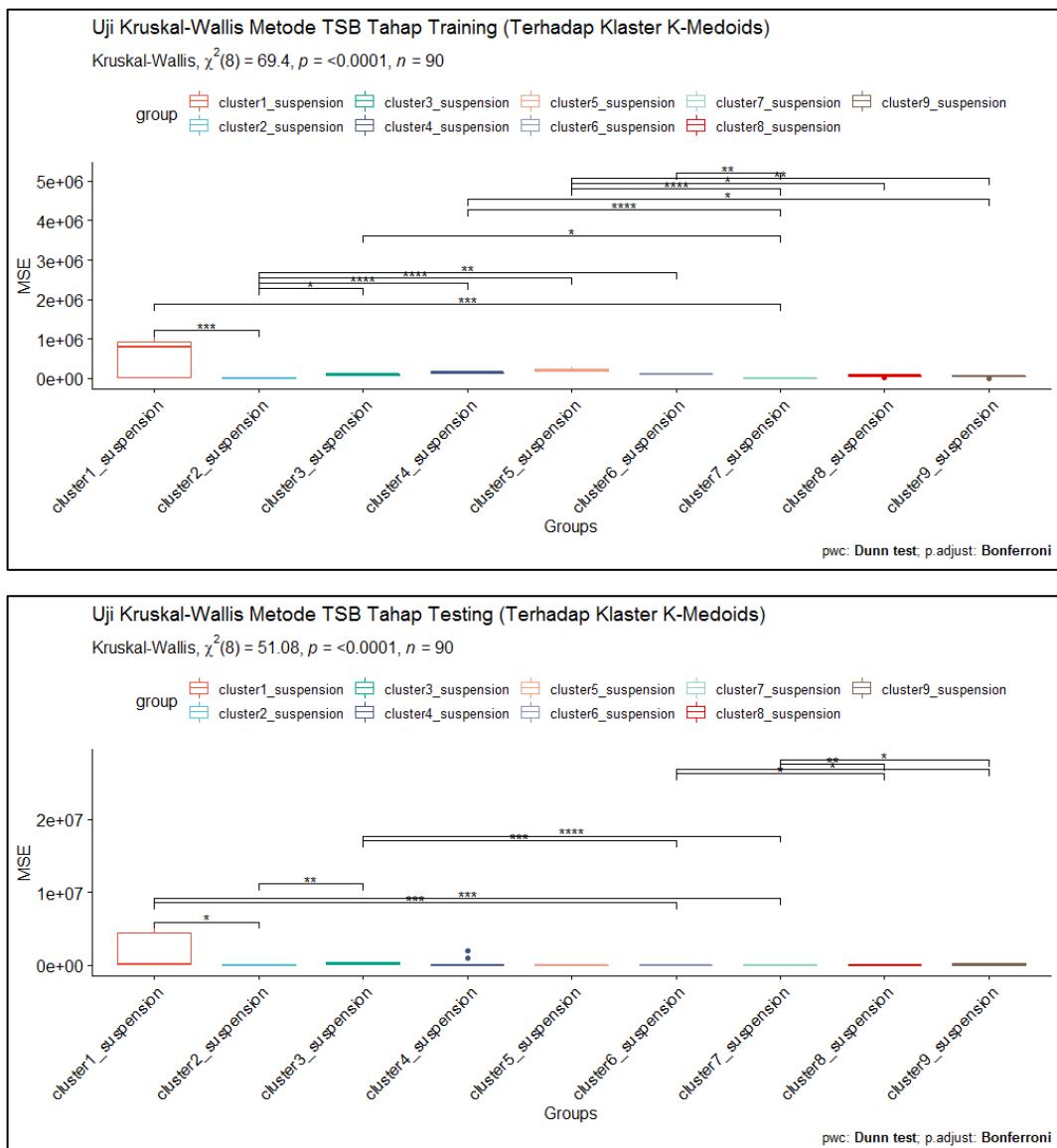
Lampiran 58 Boxplot MSE *Training* dan *Testing* Metode TSB Terhadap Klaster K-Medoids

```

Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Klaster K-Medoids)
> leveneTest(tsb_all_cluster_train$MSE, tsb_all_cluster_train$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  8 10.857 2.892e-10 ***
   81
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klaster K-Medoids)
> leveneTest(tsb_all_cluster_test$MSE, tsb_all_cluster_test$group,
+ alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  8 4.8031 7.381e-05 ***
   81
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |

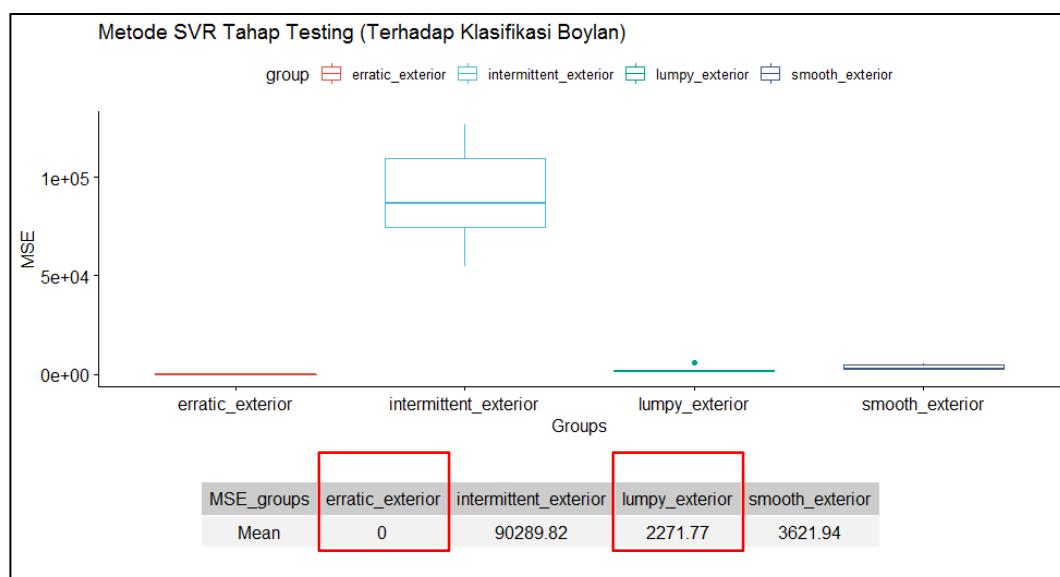
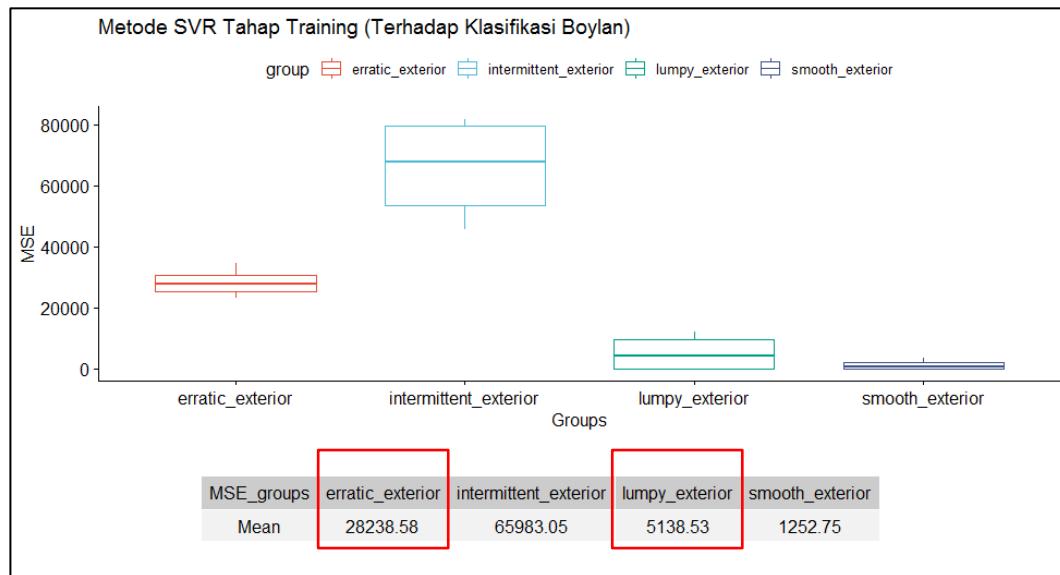
```

Lampiran 59 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode TSB
Terhadap Klaster K-Medoids



Lampiran 60 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode TSB

- **Hasil uji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja model peramalan SVR**

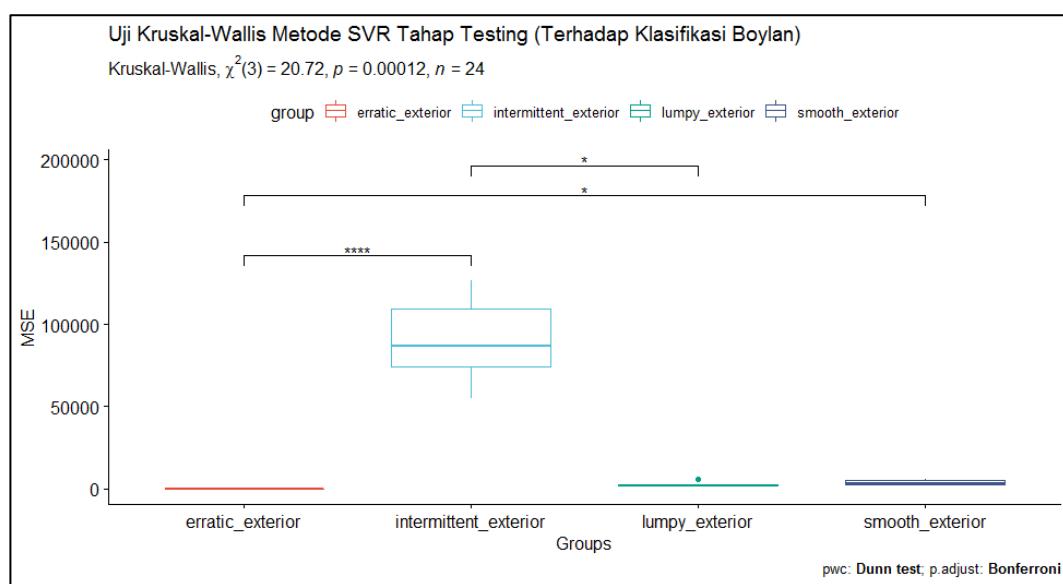
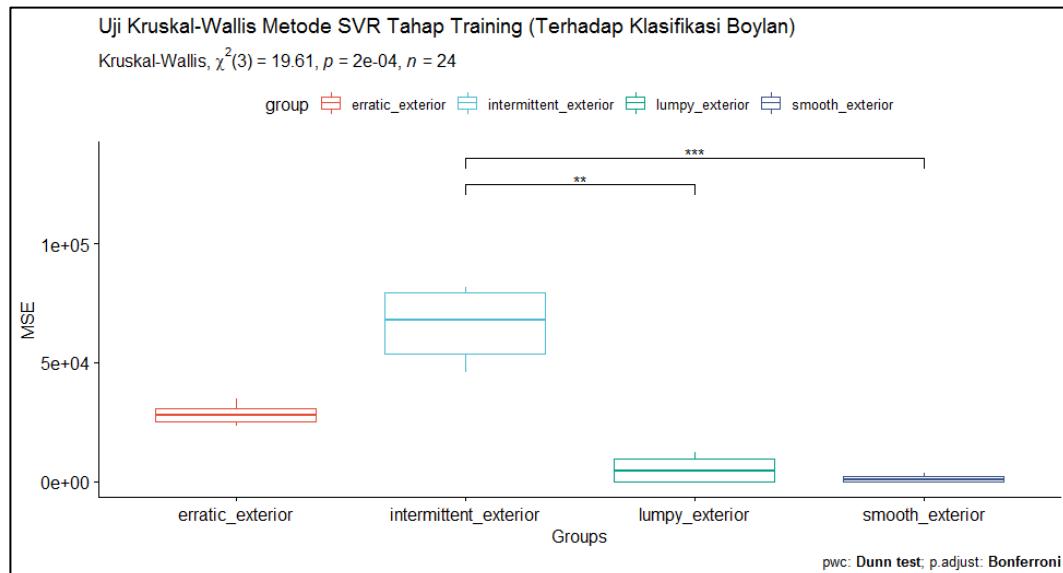


Lampiran 61 Boxplot MSE *Training* dan *Testing* Metode SVR Terhadap Berbagai Pola Permintaan Boylan

Console Terminal × Jobs ×

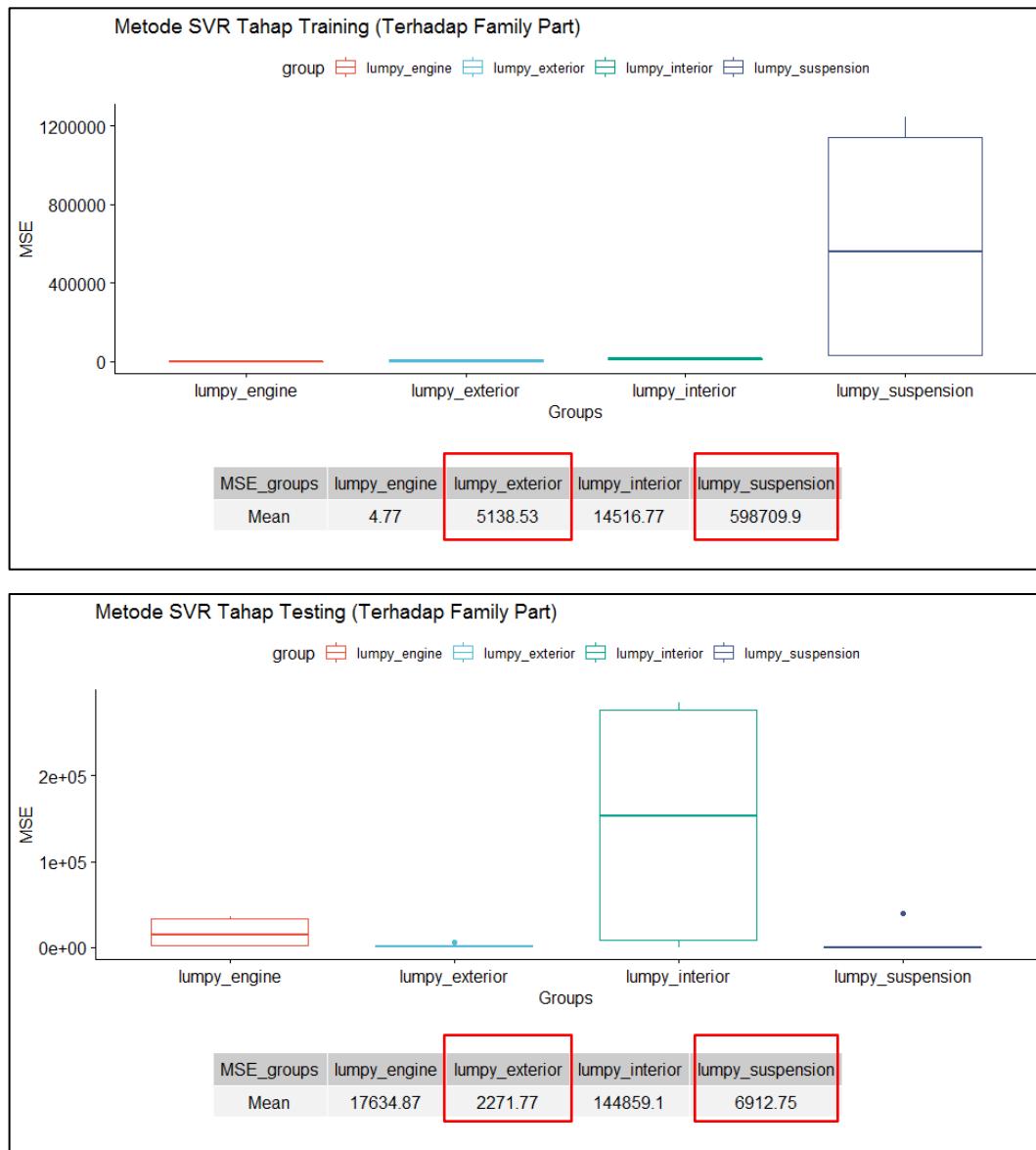
```
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas Varian (p value < 0.05 = Varian Tidak Homogen)
> # MSE Training (Klasifikasi Boylan)
> leveneTest(svr_all_boylan_train$MSE, svr_all_boylan_train
+             $group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 13.299 5.318e-05 ***
20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(svr_all_boylan_test$MSE, svr_all_boylan_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3 18.582 5.313e-06 ***
20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 62 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode SVR
Terhadap Berbagai Pola Permintaan Boylan



Lampiran 63 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode SVR

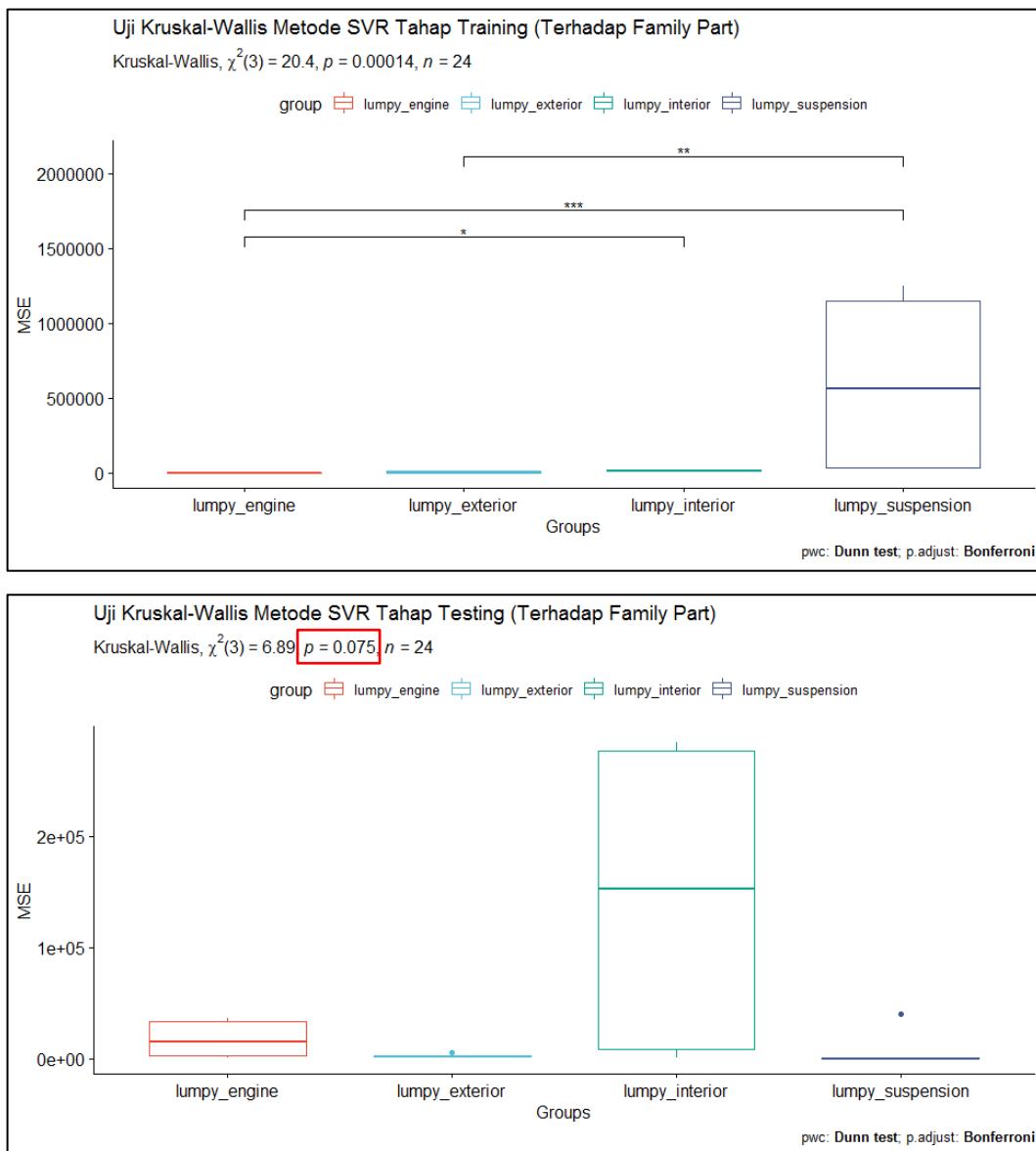
- Hasil uji pengaruh perbedaan *family part* terhadap kinerja model peramalan SVR



Lampiran 64 Boxplot MSE *Training* dan *Testing* Metode SVR Terhadap Berbagai *Family Part*

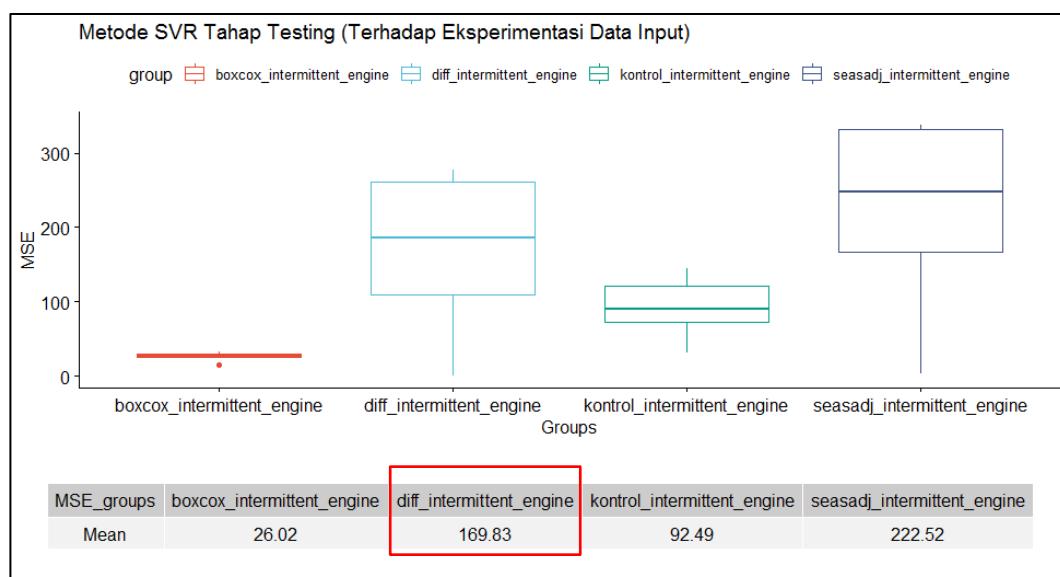
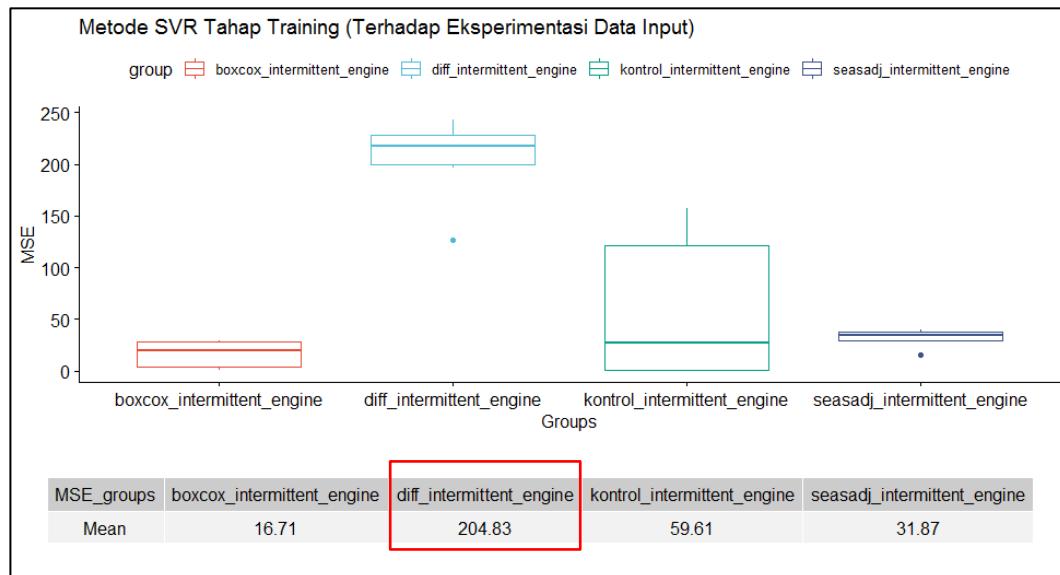
```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (klasifikasi Boylan)
> leveneTest(svr_all_family_train$MSE, svr_all_family_train
+             $group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3  455.75 < 2.2e-16 ***
20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(svr_all_family_test$MSE, svr_all_family_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of variance (center = median: c("two.sided"))
  Df F value    Pr(>F)
group  3  183.97 9.959e-15 ***
20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 65 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode SVR
Terhadap Berbagai *Family Part*



Lampiran 66 Uji Signifikansi Pengaruh *Family Part* Terhadap Metode SVR

- **Hasil uji pengaruh perubahan-perubahan pada *input* data permintaan yang sama terhadap kinerja model peramalan SVR**

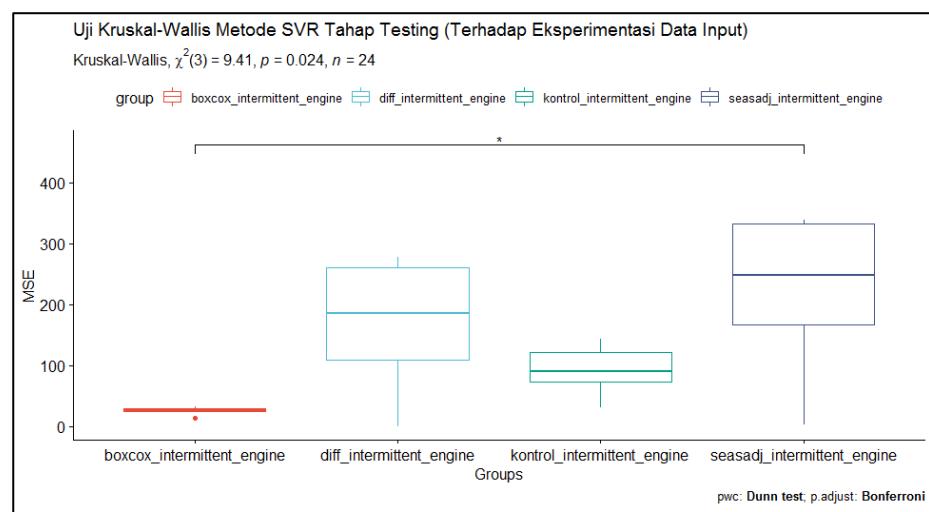
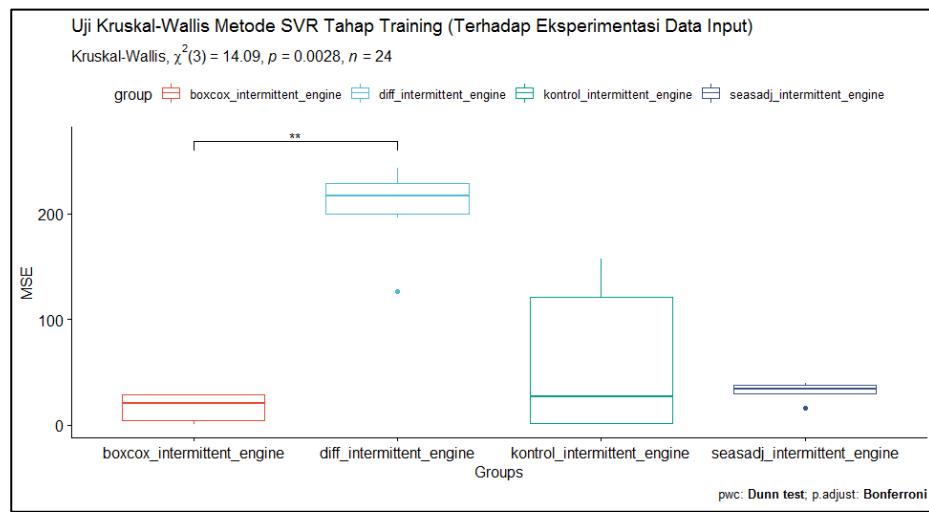


Lampiran 67 Boxplot MSE *Training* dan *Testing* Metode SVR Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (klasifikasi Boylan)
> leveneTest(svr_all_experiment_train$MSE, svr_all_experiment_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  3.6667 0.02967 *
  20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (klasifikasi Boylan)
> leveneTest(svr_all_experiment_test$MSE, svr_all_experiment_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  6.5767 0.002846 **
  20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

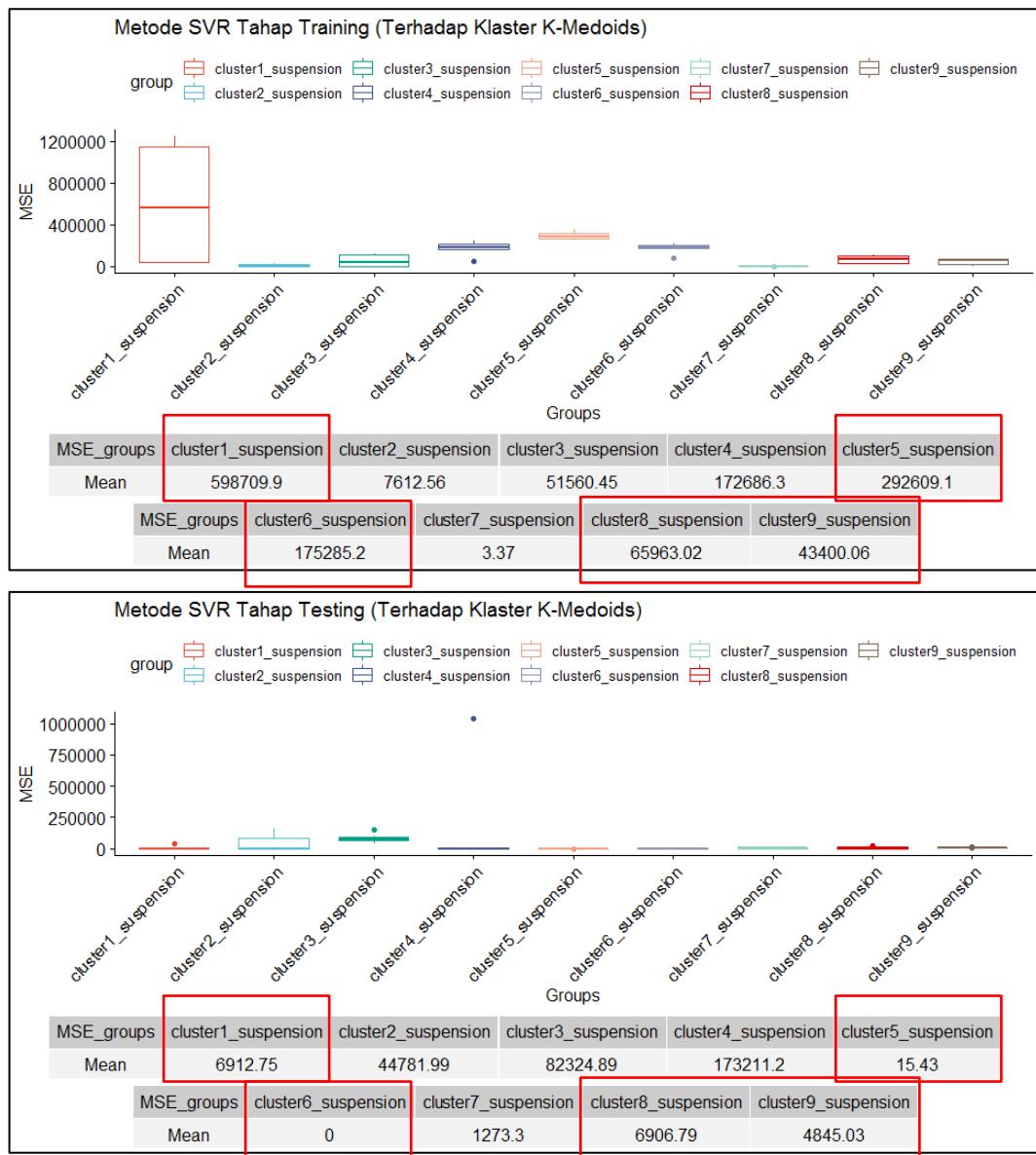
Lampiran 68 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode SVR

Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama



Lampiran 69 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode SVR

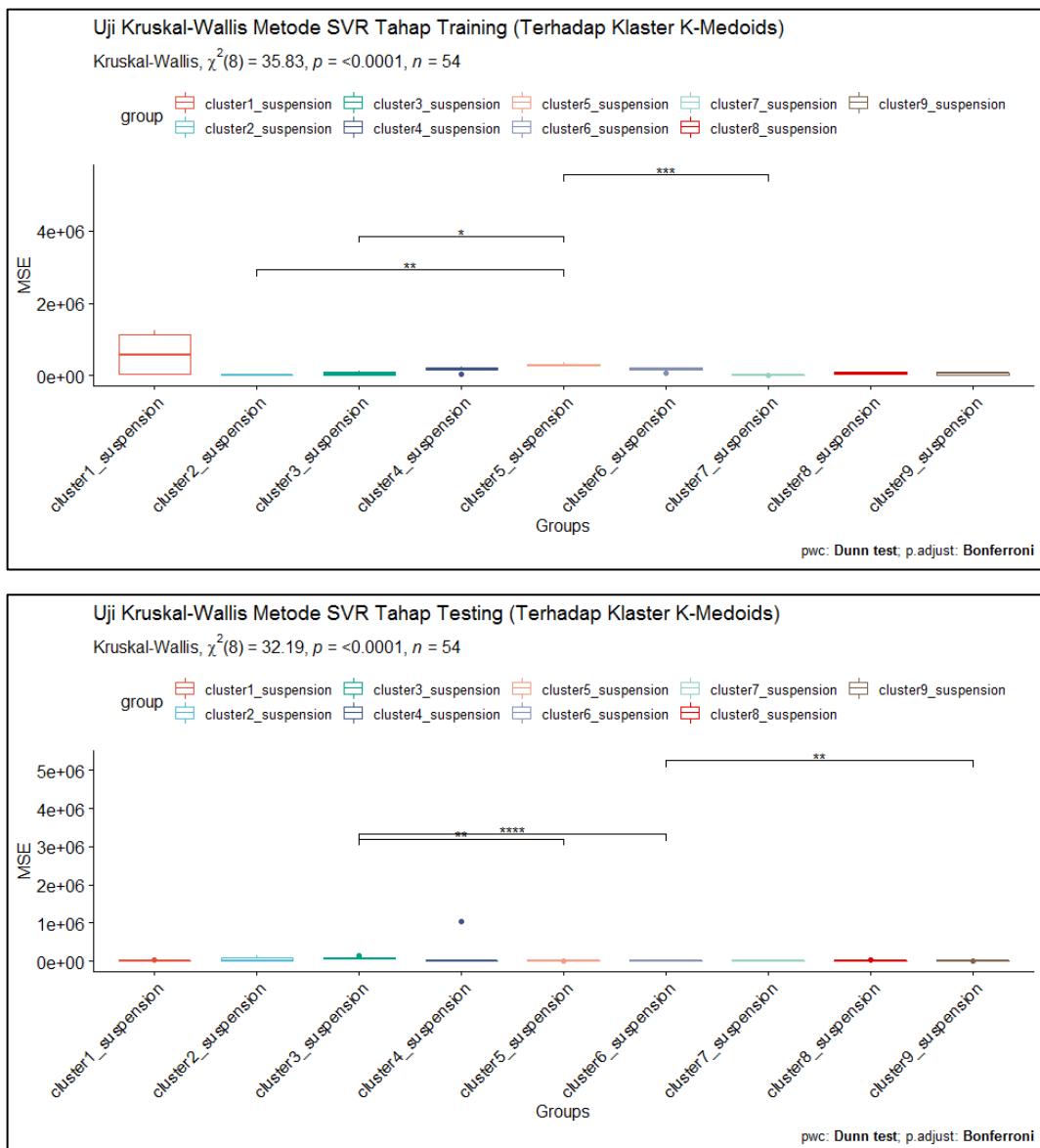
- Hasil uji pengaruh perbedaan pengelompokan *part* hasil K-Medoids *Clustering* terhadap kinerja model peramalan SVR



Lampiran 70 Boxplot MSE *Training* dan *Testing* Metode SVR Terhadap Klaster K-Medoids

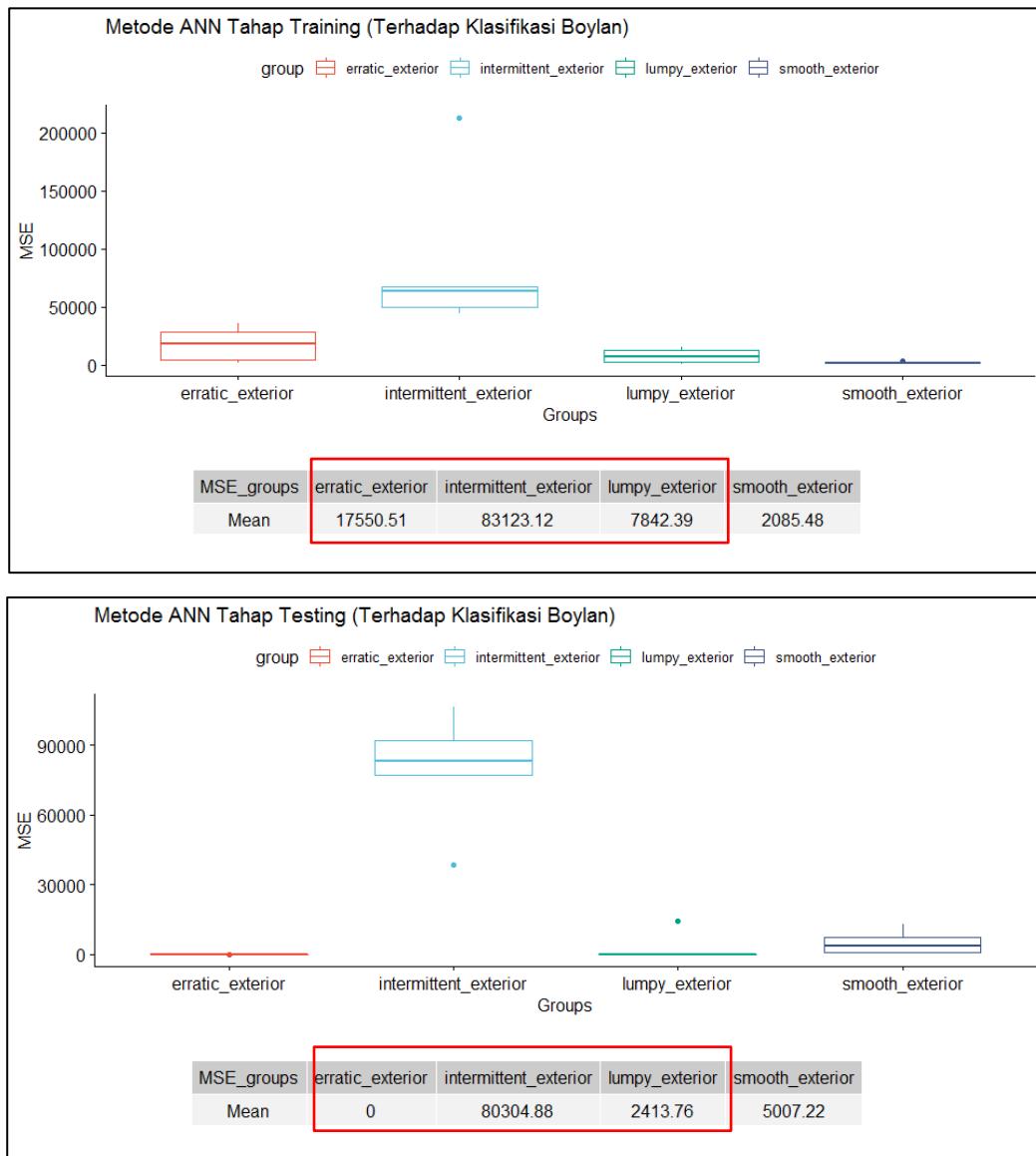
```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas varian (p value < 0.05 = Varian Tidak Homogen)
> # MSE Training (Klasifikasi Boylan)
> leveneTest(svr_all_cluster_train$MSE, svr_all_cluster_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value Pr(>F)
group  8 175.4 < 2.2e-16 ***
               45
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(svr_all_cluster_test$MSE, svr_all_cluster_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value Pr(>F)
group  8  0.9213 0.5081
               45
> |
```

Lampiran 71 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode SVR
Terhadap Klaster K-Medoids



Lampiran 72 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode SVR

- **Hasil uji pengaruh perbedaan pola permintaan hasil klasifikasi Boylan terhadap kinerja model peramalan ANN**



Lampiran 73 Boxplot MSE *Training* dan *Testing* Metode ANN Terhadap Berbagai Pola Permintaan Boylan

```

Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Klasifikasi Boylan)
> leveneTest(ann_all_boyln_train$MSE, ann_all_boyln_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  1.3916 0.2744
20
> # MSE Testing (Klasifikasi Boylan)
> leveneTest(ann_all_boyln_test$MSE, ann_all_boyln_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  3.9749 0.02257 *
20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>

```

Lampiran 74 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode ANN
 Terhadap Berbagai Pola Permintaan Boylan

```

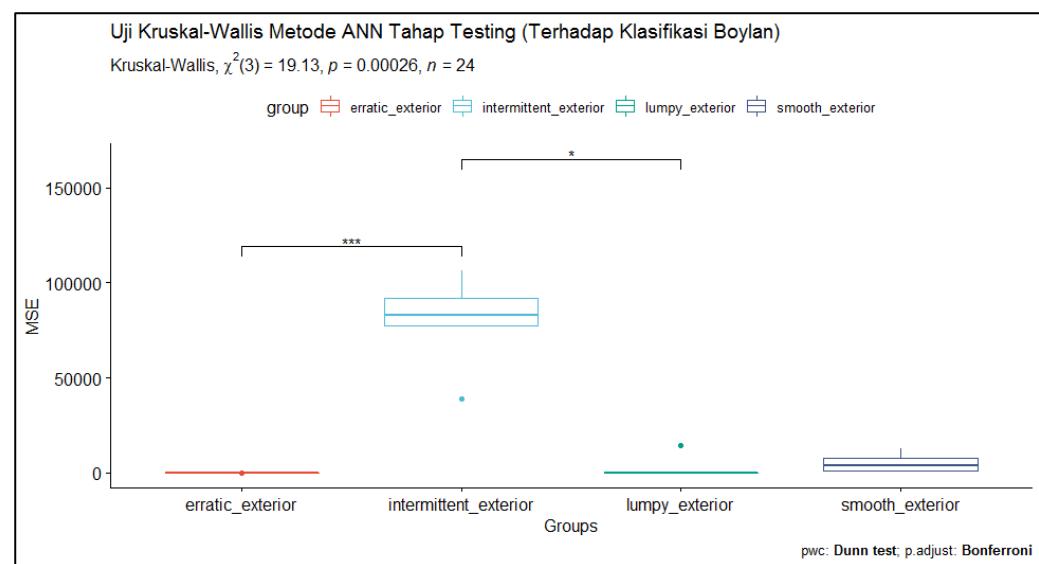
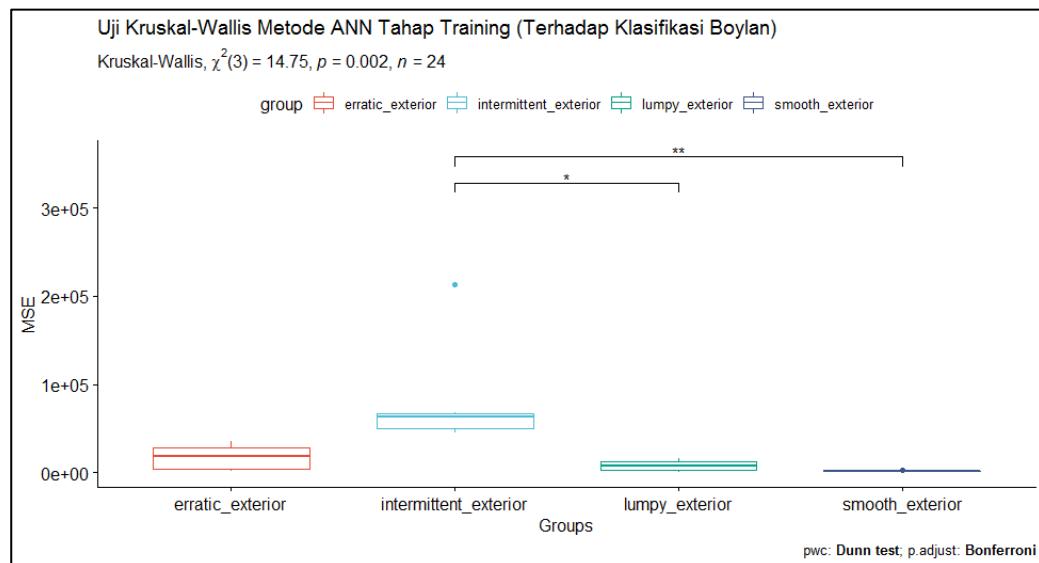
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↗
> # Uji Normalitas Residual Shapiro-wilk (p value < 0.05 = Residual Tidak
> # Berdistribusi Normal)
> # MSE Training (Klasifikasi Boylan)
> linmod <- lm(MSE~group,data=ann_all_boyln_train)
> shapiro.test(linmod$residuals)

  Shapiro-wilk normality test

data: linmod$residuals
W = 0.63447, p-value = 1.535e-06

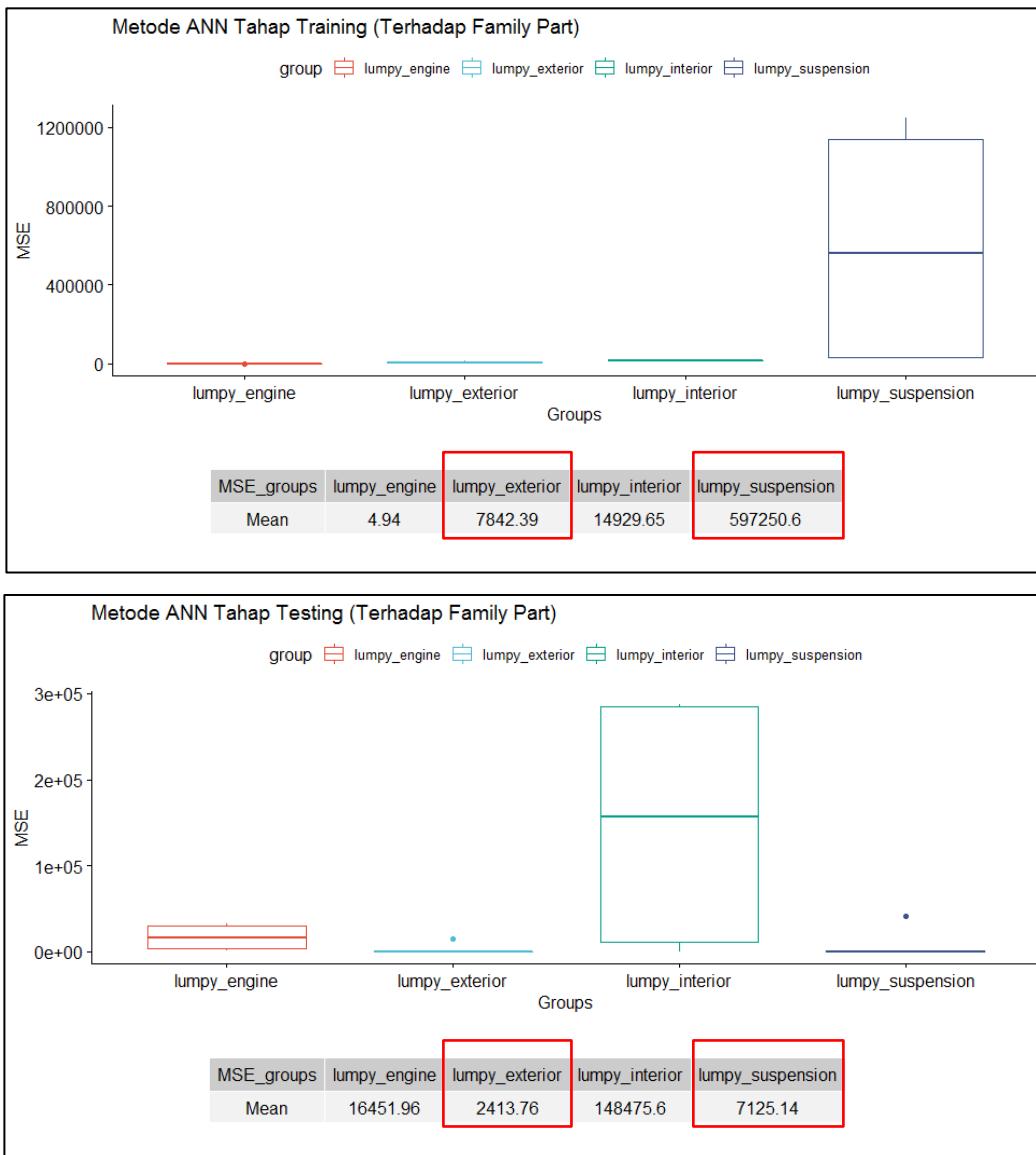
```

Lampiran 75 Uji Asumsi Anova Normalitas Residual MSE *Training* Metode ANN
 Terhadap Berbagai Pola Permintaan Boylan



Lampiran 76 Uji Signifikansi Pengaruh Klasifikasi Boylan Terhadap Metode ANN

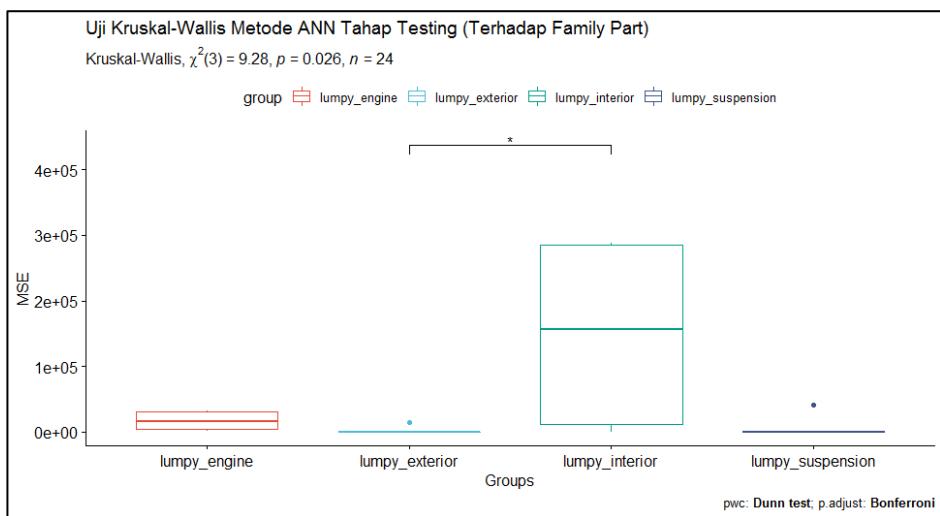
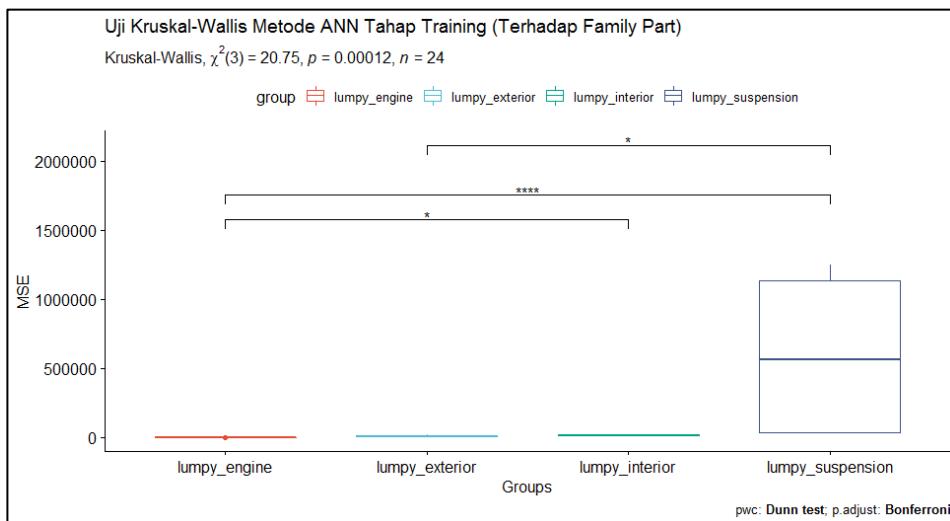
- Hasil uji pengaruh perbedaan *family part* terhadap kinerja model peramalan ANN



Lampiran 77 Boxplot MSE *Training* dan *Testing* Metode ANN Terhadap Berbagai *Family Part*

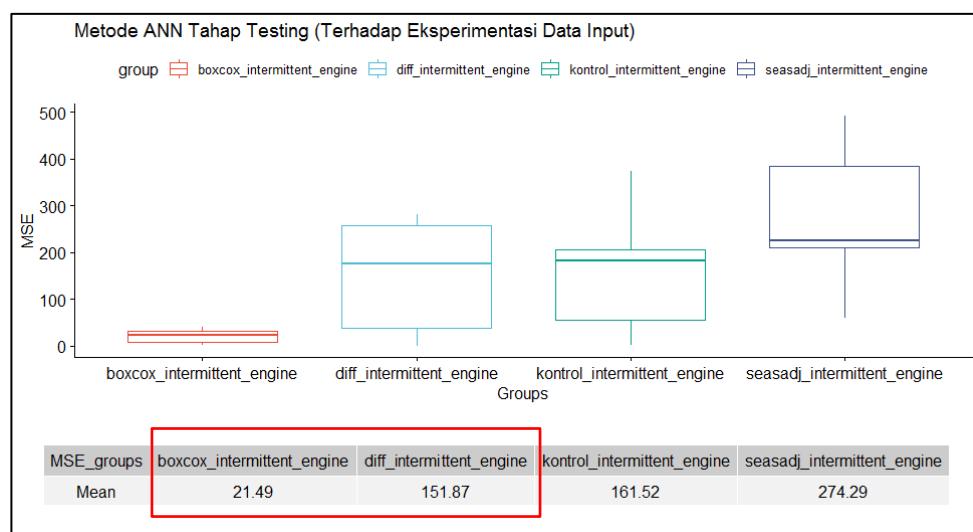
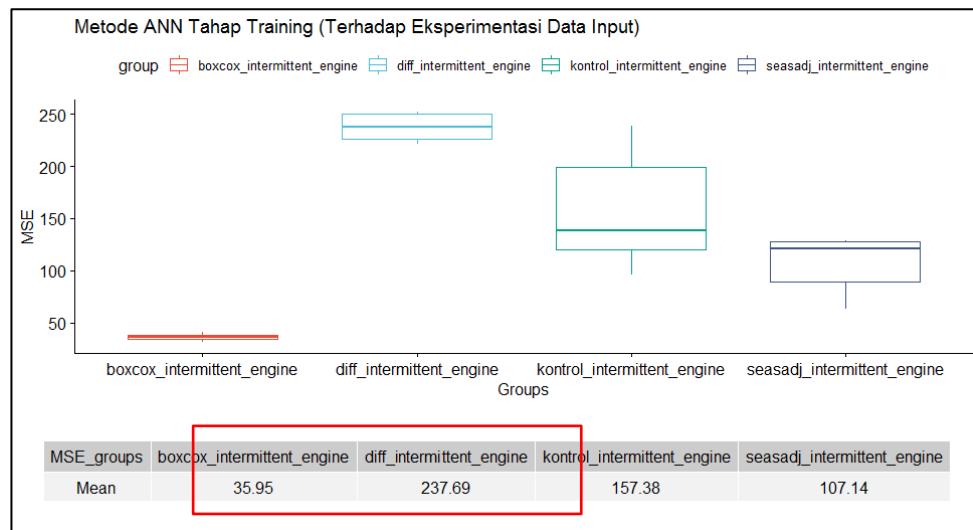
```
Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas Varian (p value < 0.05 = Varian Tidak Homogen)
> # MSE Training (Family Part)
> leveneTest(ann_all_family_train$MSE, ann_all_family_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  3  469.41 < 2.2e-16 ***
   20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Family Part)
> leveneTest(ann_all_family_test$MSE, ann_all_family_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
   Df F value    Pr(>F)
group  3  187.52 8.283e-15 ***
   20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Lampiran 78 Uji Asumsi Anova Homogenitas Varian MSE *Training* Metode ANN
Terhadap Berbagai *Family Part*



Lampiran 79 Uji Signifikansi Pengaruh *Family Part* Terhadap Metode ANN

- **Hasil uji pengaruh perubahan-perubahan pada *input* data permintaan yang sama terhadap kinerja model peramalan ANN**



Lampiran 80 Boxplot MSE *Training* dan *Testing* Metode ANN Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

```

Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Eksperimentasi Data Input)
> leveneTest(ann_all_experiment_train$MSE, ann_all_experiment_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  3.1527 0.04752 *
20
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> # MSE Testing (Eksperimentasi Data Input)
> leveneTest(ann_all_experiment_test$MSE, ann_all_experiment_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  Df F value Pr(>F)
group  3  1.9461 0.1547
20
> |

```

Lampiran 81 Uji Asumsi Anova Homogenitas Varian MSE Training Metode ANN

Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

```

Console Terminal × Jobs ×
C:/Users/X302U/Desktop/ ↵
> # Uji Normalitas Residual Shapiro-Wilk (p value < 0.05 = Residual Tidak
> # Berdistribusi Normal)
> # MSE Testing (Eksperimentasi Data Input)
> linmod <- lm(MSE~group,data=ann_all_experiment_test)
> shapiro.test(linmod$residuals)

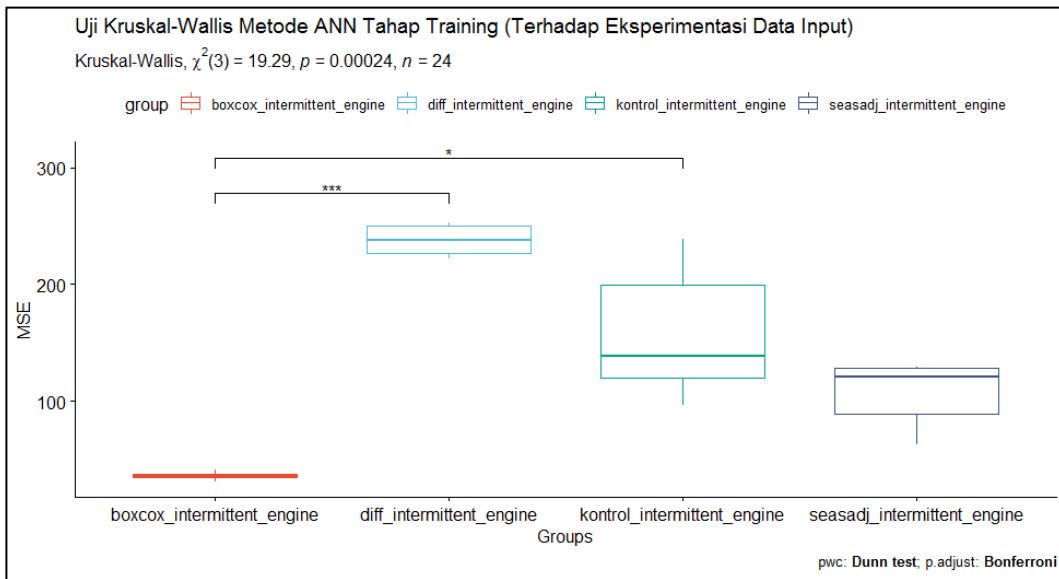
Shapiro-Wilk normality test

data: linmod$residuals
W = 0.95787, p-value = 0.3972

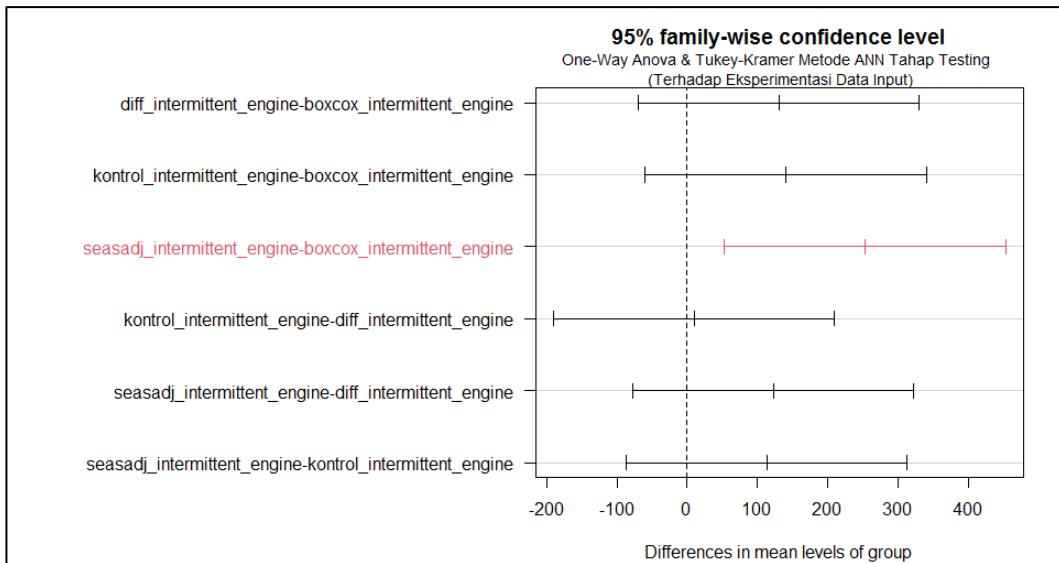
```

Lampiran 82 Uji Asumsi Anova Normalitas Residual MSE Testing Metode ANN

Terhadap Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama

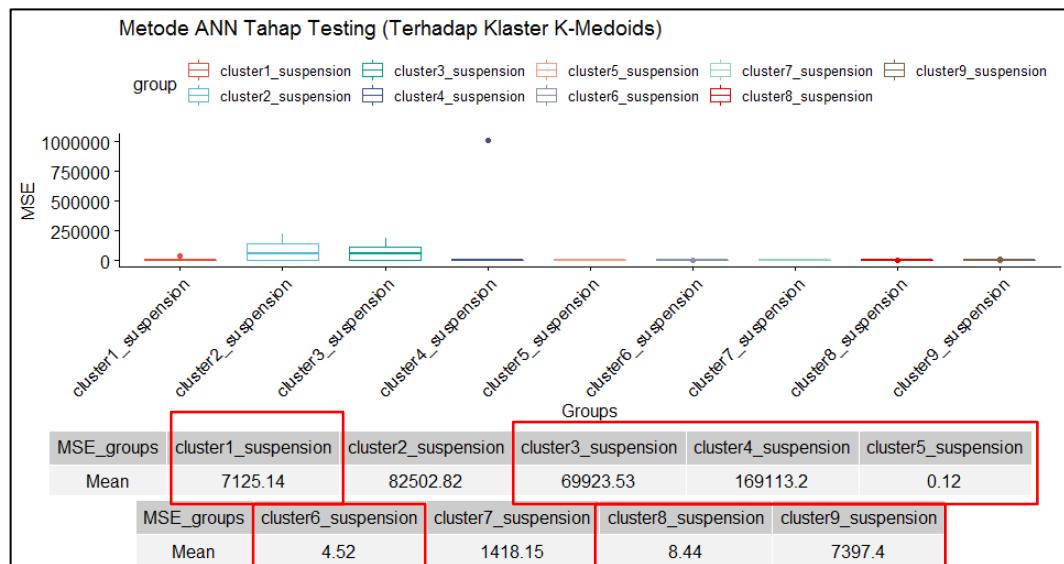
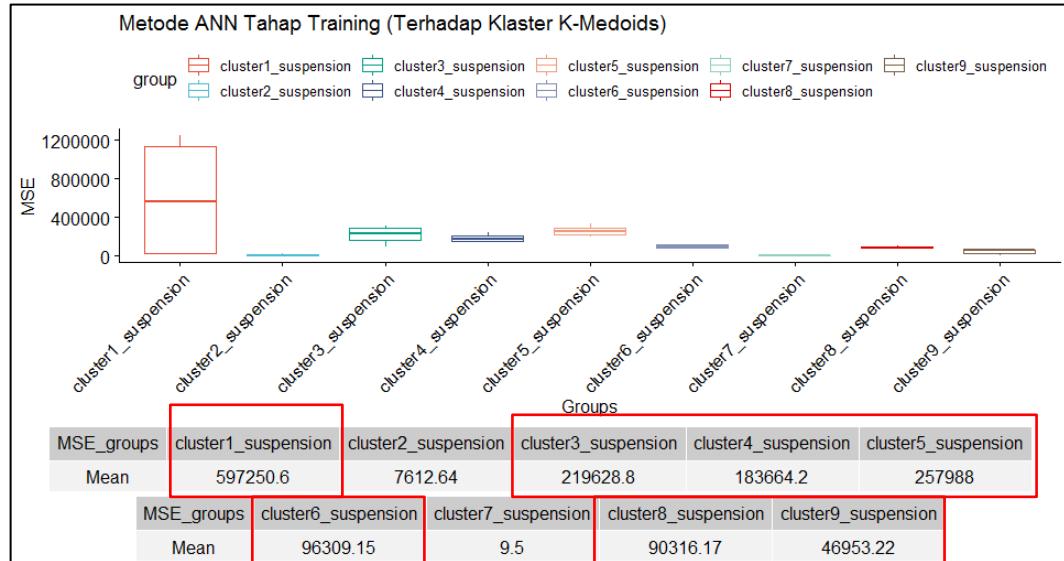


Lampiran 83 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode ANN Di Tahap *Training*



Lampiran 84 Uji Signifikansi Pengaruh Perubahan-perubahan akibat eksperimentasi pada input peramalan yang sama Terhadap Metode ANN Di Tahap *Testing*

- Hasil uji pengaruh perbedaan pengelompokan *part* hasil K-Medoids
Clustering terhadap kinerja model peramalan ANN



Lampiran 85 Boxplot MSE Training dan Testing Metode ANN Terhadap Klaster K-Medoids

Console Terminal × Jobs ×

```
C:/Users/X302U/Desktop/ ↵
> # Uji Homogenitas Varian (p value < 0.05 = varian Tidak Homogen)
> # MSE Training (Klaster K-Medoids)
> leveneTest(ann_all_cluster_train$MSE, ann_all_cluster_train$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  DF F value    Pr(>F)
group  8  213.65 < 2.2e-16 ***
               45
---
signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> # MSE Testing (Klaster K-Medoids)
> leveneTest(ann_all_cluster_test$MSE, ann_all_cluster_test$group,
+             alternative = c("two.sided"), conf.level = 0.95)
Levene's Test for Homogeneity of Variance (center = median: c("two.sided"))
  DF F value    Pr(>F)
group  8   1.0804 0.3939
               45
> |
```

Lampiran 86 Uji Asumsi Anova Homogenitas Varian MSE Training Metode ANN
Terhadap Klaster K-Medoids

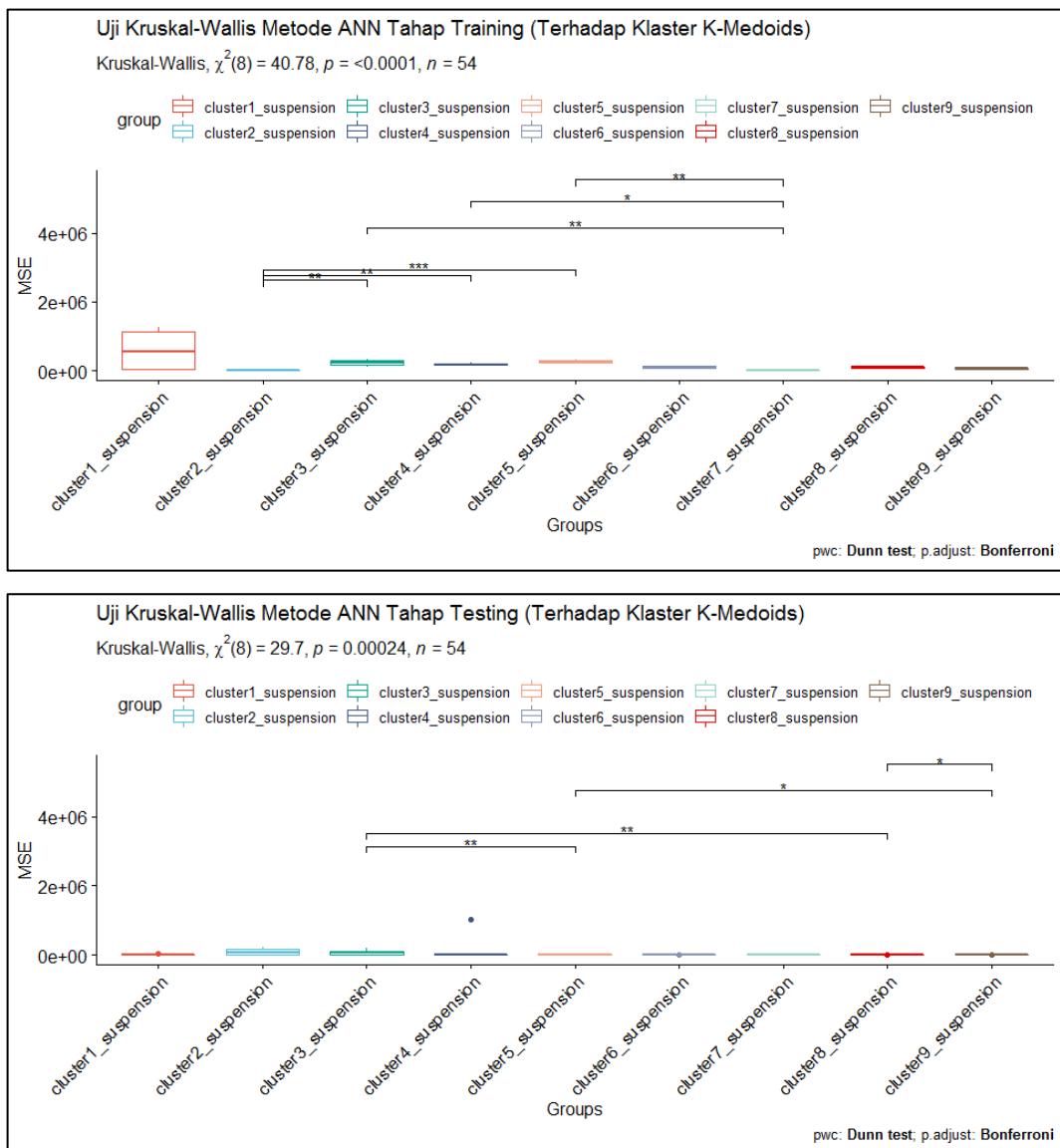
Console Terminal × Jobs ×

```
C:/Users/X302U/Desktop/ ↵
> # Uji Normalitas Residual Shapiro-Wilk (p value < 0.05 = Residual Tidak
> # Berdistribusi Normal)
> # MSE Testing (Klaster K-Medoids)
> linmod <- lm(MSE~group,data=ann_all_cluster_test)
> shapiro.test(linmod$residuals)

  Shapiro-Wilk normality test

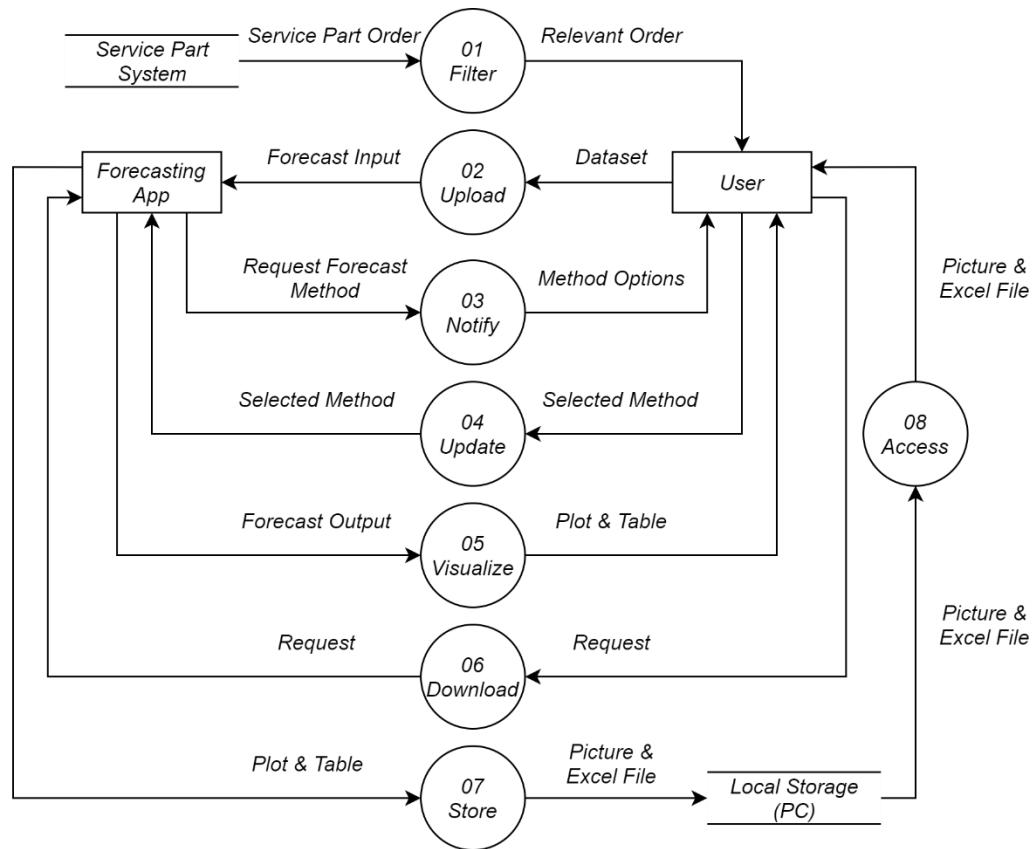
data: linmod$residuals
W = 0.47816, p-value = 1.41e-12
```

Lampiran 87 Uji Asumsi Anova Normalitas Residual MSE Testing Metode ANN
Terhadap Klaster K-Medoids



Lampiran 88 Uji Signifikansi Pengaruh Klaster K-Medoids Terhadap Metode ANN

- Skema aplikasi peramalan sebagai usulan perbaikan sistem peramalan permintaan *part* di PT. X



Lampiran 89 *Data Flow Diagram* Skema Aplikasi Peramalan Implementasi Dari Usulan Perbaikan Pada Tugas Akhir Ini

- Model peramalan Croston, SBA, dan TSB pada RStudio

```
Function: cr_sba_tsb_frc (.GlobalEnv)
1 ✓ function(dataset) {
2
3   # In-sample Data
4   a <- length(dataset) - 19 + 1
5   b <- length(dataset) - 20 + 1
6   c <- length(dataset) - 21 + 1
7   ind <- c(a,b,c)
8   cr_in_sample <- dataset[ind[2]:length(dataset)]
9   sba_in_sample <- dataset[ind[1]:length(dataset)]
10  tsb_in_sample <- dataset[ind[3]:length(dataset)]
11
12  # Forecasting with Croston
13  cr_frc<-crost(
14    cr_in_sample,
15    h=3,
16    w=NULL,
17    init="mean",
18    nop=1,
19    type="croston",
20    cost=c("mae","mse"),
21    init.opt = F,
22    outplot = F,
23    opt.on = F,
24    na.rm = F
25  )
```

Lampiran 90 Model Peramalan Croston

```
27  # Forecasting with SBA
28  sba_frc<-crost(
29    sba_in_sample,
30    h=3,
31    w=NULL,
32    init="mean",
33    nop=1,
34    type="sba",
35    cost=c("mae","mse"),
36    init.opt = F,
37    outplot = F,
38    opt.on = F,
39    na.rm = F
40  )
41
42  # Forecasting with TSB
43  tsb_frc<-tsb(
44    tsb_in_sample,
45    h=3,
46    w=NULL,
47    init="mean",
48    cost=c("mae","mse"),
49    init.opt = F,
50    outplot = F,
51    opt.on = F,
52    na.rm = F
53  )
```

Lampiran 91 Model Peramalan SBA dan TSB

- Model Peramalan SVR dan ANN pada RStudio

```
Function: svr_frc (.GlobalEnv)
1 ~ function(dataset) {
2
3     # In-sample Data
4     ind <- 13
5     z1 <- ind
6     a2 <- z1 + 1
7     z2 <- z1 + 3
8     a3 <- z2 + 1
9     z3 <- z2 + 3
10    xtrain <- dataset[1:z1,1:3]
11    ytrain <- dataset[1:z1,4]
12    xval <- dataset[a2:z2,1:3]
13    yval <- dataset[a2:z2,4]
14
15    # Forecasting with SVR
16    tunecont<-tune.control(
17        random=F,
18        sampling="fix",
19        fix=1
20    )
21
22    tuneresult<-tune(
23        SVM,
24        xtrain,
25        ytrain,
26        scale=F,
27        type="eps-regression",
28        kernel="radial",
29        ranges=list(
30            epsilon=seq(0,1,0.1),
31            cost=2^(seq(2,10,0.66)),
32            gamma=10^(seq(-10,0,0.66))),
33            tunecontrol = tunecont,
34            validation.x = xval,
35            validation.y = yval
36        )
37    tunedmodel<-tuneresult$best.model
```

Lampiran 92 Model Peramalan SVR

```

Function: ann_frc (.GlobalEnv)
1 ~ function(dataset) {
2
3   # In-sample Data
4   ind <- 13
5   z1 <- ind
6   a2 <- z1 + 1
7   z2 <- z1 + 3
8   a3 <- z2 + 1
9   z3 <- z2 + 3
10  xtrain <- as.matrix(dataset[1:z1,1:3])
11  ytrain <- as.matrix(dataset[1:z1,4])
12  xval <- as.matrix(dataset[a2:z2,1:3])
13  yval <- as.matrix(dataset[a2:z2,4])
14
15  # Forecasting with ANN
16  par <- list(
17    dropout1 = seq(0.166,0.5,0.066),
18    dropout2 = seq(0.166,0.5,0.066),
19    neurons1 = c(16,32,64),
20    neurons2 = c(16,32,64),
21    neurons3 = c(16,32,64),
22    l2 = c(0.0001,0.001,0.01),
23    lr = c(0.0001,0.001,0.01)
24  )
25
26  runs <- tuning_run('annmlp4.R', runs_dir='_tuning4',
27                      sample=0.002, flags=par)
28
29  ls_runs(order=metric_val_mean_absolute_error, decreasing=F,
30          runs_dir='_tuning4')
31
32  best_run <- ls_runs(order=metric_val_mean_absolute_error, decreasing=F,
33                        runs_dir='_tuning4')[1,]

```

Lampiran 93 Model Peramalan ANN (1)

```

30 ~ =====
31 # Build Sequential Model
32 ~ =====
33 #(4) Build Base Model
34 ~ build_model <- function() {
35   model<-keras_model_sequential()
36   model %>%
37     layer_dense(units=FLAGS$neurons1,
38                 input_shape=dim(xtrain)[2],
39                 activation='relu',
40                 kernel_initializer = initializer_glorot_uniform(seed=104),
41                 kernel_regularizer = regularizer_l2(l=FLAGS$l2)) %>%
42     layer_dense(units=FLAGS$neurons2,
43                 activation='relu') %>%
44     layer_dropout(FLAGS$dropout1) %>%
45     layer_dense(units=FLAGS$neurons3,
46                 activation='relu') %>%
47     layer_dropout(FLAGS$dropout2) %>%
48     layer_dense(units=1,activation='relu')
49
50   model %>% compile(
51     loss="mse",
52     optimizer=optimizer_rmsprop(lr=FLAGS$lr),
53     metrics=list('mean_absolute_error')
54   )

```

Lampiran 94 Model Peramalan ANN (2)

```

60 #=====
61 # Train the Base Model
62 #=====
63 #(5) Define Stopping Condition
64 early_stop<-callback_early_stopping(monitor="val_loss",
65                                         patience=20)
66 epochs<-100
67
68 #(6) Define Training conditions
69 history<-model %>% fit(
70   xtrain, ytrain,
71   epochs = epochs,
72   verbose=0,
73   callbacks=list(early_stop),
74   validation_data=list(xval,yval),
75   shuffle=F
76 )
77 plot(history)
78
79 #(7) Return Training Performance
80 score<-model %>% evaluate(
81   xtest, ytest,
82   verbose=0
83 )
84
85 #(8) Save Base Model to External Source
86 save_model_hdf5(model,'model4.h5')
87 cat('Test Loss:',score$loss,'\n')
88 cat('Test accuracy',score$mean_absolute_error,'\n')

```

Lampiran 95 Model Peramalan ANN (3)

- **Visualisasi peramalan dan perhitungan interval peramalan pada RStudio**

```

1 ggplot(data = all_data, aes(x = time)) +
2   geom_line(aes(y = actual_data, color = "actual_data"), size = 1) +
3   geom_line(aes(y = point_forecast, color = "point_forecast"), size = 1) +
4   geom_line(aes(y = forecast_upper_interval, color = "forecast_interval"),
5             size = 1) +
6   geom_line(aes(y = forecast_lower_interval, color = "forecast_interval"),
7             size = 1) +
8   scale_colour_manual(
9     "",
10    breaks = c("actual_data", "point_forecast", "forecast_interval"),
11    values = c("black", "red", "blue")
12  ) +
13  geom_vline(xintercept = as.numeric(as.Date("2019-12-1"))) +
14  scale_x_date(date_labels = "%b %Y") +
15  labs(
16    title = "...",
17    subtitle = "...",
18    y = "Demand",
19    x = "Period"
20  ) +
21  facet_grid(group ~ method, scales = "free_y")

```

Lampiran 96 Visualisasi Peramalan Berdasarkan Metode dan Kelompok (Grup)
Part

```
Method: confint (GlobalEnv) ▾
1 ▾ function(forecastdata,fitdata,actualdata) {
2   upconfint <- matrix(rep(0,39),nrow=3)
3   lowconfint <- matrix(rep(0,39),nrow=3)
4 ▾   for (j in 1:13) {
5 ▾     for (i in 1:3) {
6       upconfint[i,j] <-
7         forecastdata[24+i,j] + 1.96*sd(
8           actualdata[,j]-na.omit(fitdata[,j]))
9       lowconfint[i,j] <- max(0,forecastdata[24+i,j] - 1.96*sd(
10          actualdata[,j]-na.omit(fitdata[,j])))
11     }
12   }
13   result <- list(upper_limit = upconfint, lower_limit = lowconfint)
14   return(result)
15 }
16 }
```

Lampiran 97 Perhitungan Interval Peramalan

- Dataset yang digunakan untuk peramalan**

Lampiran 98 Permintaan Ekspor Bulanan *Spare Part* Ke Arab Saudi Selama 2018-2019

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
534200K430	HINGE ASSY, HOOD, LH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	0	148	94	48	151	0
531120D360	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	1	160	166	98	239	0
531120K240	GRILLE, RADIATOR, LWR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	4	110	60	49	81	10
525360D190	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	18	109	147	91	126	16
526110D080	ABSORBER, FR BUMPER ENERGY	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	76	88	43	106	5
646070D250	CABLE SUB- ASSY, LUGGAGE DOOR LOCK CONT	MEDIUM	Vios	Engine	Abdul Latif Jameel	0	2	20	15	24	1
12361BZ162	INSULATOR, ENGINE MOUNTING, FR RH	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	3	5	13	0	13	3
521190U933	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	2

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
521590U933	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
525360D290	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
531120D780	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
525350D290	RETAINER, FR BUMPER SIDE, RH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
743200K450E0	VISOR A/S LH	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	16	7	18	0
83800FAC20	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	9	5	3	0	0
83800FAC21	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	14	0
636500K570B0	BOX A/S ROOF CONSOLE	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	0	0	5	0
16620BZ010	TENSIONER ASSY, V-RIBBED BELT	MEDIUM	Daihatsu Model (Avanza/Rush)	Interior	Abdul Latif Jameel	0	1	5	3	0	2
13101BZ100	PISTON SUB-ASSY, W/PIN	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	21	9	0	13	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
123720Y280	INSULATOR, ENGINE MOUNTING, LH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	18	15	0	11	0
132110Y011	PISTON	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	23	8	8	0
11101B9622	HEAD SUB- ASSY, CYLINDER	BIG	#N/A	Engine	Abdul Latif Jameel	0	0	8	0	0	4
123640Y010	ROD, ENGINE MOVING CONTROL, NO.2	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	0	0
312500K310	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	0	0	893	343	734
485100DD80	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
485100DC40	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	0	123	169	206	244	51
43512BZ280	DISC, FR	BIG	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	0	0	41	50	21	0
48815BZ290	BUSH, STABILIZER	MEDIUM	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
521190U920	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	22	529	503	202	557	83

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
521590U908	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	14	416	334	120	356	48
521590M911	COVER, RR BUMPER L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	3	87	94	44	126	33
534200D630	HINGE ASSY, HOOD, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	16	62	79	40	139	12
483040D170	BUMPER SUB-ASSY, FR SPRING	MEDIUM	Vios	Exterior	Abdul Latif Jameel	4	194	160	75	87	8
885900K050	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	1	18	14	4	9
871030D370	MOTOR SUB-ASSY, BLOWER W/FAN	MEDIUM	Vios	Interior	Abdul Latif Jameel	0	5	0	1	10	2
90916T2039	BELT, V-RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	2	25	32	0	0	0
15601BZ030	ELEMENT SUB-ASSY, OIL FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	10	136	79	216	164	5
111150Y030	GASKET, CYLINDER HEAD	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	28	127	125	145	176
17801BZ150	ELEMENT SUB-ASSY, AIR CLEANER FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	9	35	48	21	23	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
123050Y060	INSULATOR SUB-ASSY, ENGINE MOUNTING, RH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	21	19	8	13	0
514420D210	COVER, ENGINE UNDER, LH	MEDIUM	Vios	Engine	Abdul Latif Jameel	10	37	81	50	95	12
312100K310	COVER ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	0	0	393	361	262
488150K130	BUSH, STABILIZER	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	137	55	140	115	62
485208Z125	ABSORBER SET, SHOCK, FR LH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	4	20	29	38	28	6
485108Z265	ABSORBER SET, SHOCK, FR RH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	6	36	14	17	24	11
049470K020	FITTING KIT, DISC BRAKE	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	2	56	32	27	19
521190M913	COVER,FR BUMPER L/C	BIG	#N/A	Exterior	Abdul Latif Jameel	0	208	112	39	65	38
811500DG30	HEADLAMP ASSY, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
534100K430	HINGE ASSY, HOOD, RH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	4	152	120	63	118	34

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
531110K700	GRILLE, RADIATOR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	7	196	128	60	123	50
531120DA40	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
90916T2029	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
90916T2046	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
871030K440	MOTOR SUB- ASSY, BLOWER W/FAN	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	0	0	0	0
885900K030	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	0	0	0	0
556620D160	BEZEL, INSTRUMENT PANEL REGISTER, NO.2	MEDIUM	Vios	Interior	Abdul Latif Jameel	0	0	0	0	0	0
232200P210	PUMP ASSY, FUEL W/FILTER	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	0	0
12361BZ200	INSULATOR, ENGINE MOUNTING, FR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	1	3	3	0	3	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.01	2018.02	2018.03	2018.04	2018.05	2018.06
514410D330	COVER, ENGINE UNDER, RH	MEDIUM	Vios	Engine	Abdul Latif Jameel	2	40	74	38	70	0
23291BZ010	INSULATOR, INJECTOR VIBRATION	BIG	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	1	5	0	4	0	0
12371BZ170	INSULATOR, ENGINE MOUNTING, RR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	60	0	0	24	0
130410Y04002	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
312500K311	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
117010Y03003	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
130410Y04003	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
117010C01101	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	168	95	0	681	0	551

Lampiran 97 Permintaan Ekspor Bulanan Spare Part Ke Arab Saudi Selama 2018-2019 (Lanjutan)

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
534200K430	HINGE ASSY, HOOD, LH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	0	14	70	48	59	89
531120D360	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
531120K240	GRILLE, RADIATOR, LWR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	0	15	67	64	0	57
525360D190	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
526110D080	ABSORBER, FR BUMPER ENERGY	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
646070D250	CABLE SUB- ASSY, LUGGAGE DOOR LOCK CONT	MEDIUM	Vios	Engine	Abdul Latif Jameel	5	2	20	23	10	17
12361BZ162	INSULATOR, ENGINE MOUNTING, FR RH	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	0	0	3	10	14
521190U933	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	120	351	740	0	394	669
521590U933	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	70	260	448	0	114	372

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
525360D290	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	6	113	298	231	102	131
531120D780	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	23	80	342	0	163	0
525350D290	RETAINER, FR BUMPER SIDE, RH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	141	308	235	130	38
743200K450E0	VISOR A/S LH	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	0	0	0	0
83800FAC20	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	7	14
83800FAC21	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
636500K570B0	BOX A/S ROOF CONSOLE	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	0	0	0	0
16620BZ010	TENSIONER ASSY, V-RIBBED BELT	MEDIUM	Daihatsu Model (Avanza/Rush)	Interior	Abdul Latif Jameel	0	0	0	0	0	3
13101BZ100	PISTON SUB-ASSY, W/PIN	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	28	0	0	0	4	4
123720Y280	INSULATOR, ENGINE MOUNTING, LH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	35	12	0	0	5

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
132110Y011	PISTON	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	0	5
11101B9622	HEAD SUB-ASSY, CYLINDER	BIG	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	0	0
123640Y010	ROD, ENGINE MOVING CONTROL, NO.2	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	0	1	2	0
312500K310	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	1035	82	0	0	0	0
485100DD80	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	0	0	0	249	124	0
485100DC40	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	0	115	139	0	0	67
43512BZ280	DISC, FR	BIG	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	4	44	6	62	8	45
48815BZ290	BUSH, STABILIZER	MEDIUM	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
521190U920	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
521590U908	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
521590M911	COVER, RR BUMPER L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	51	46	150	0	45	97
534200D630	HINGE ASSY, HOOD, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	3	25	147	123	24	0
483040D170	BUMPER SUB-ASSY, FR SPRING	MEDIUM	Vios	Exterior	Abdul Latif Jameel	6	239	27	14	27	1
885900K050	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	3	2	35	14	69
871030D370	MOTOR SUB-ASSY, BLOWER W/FAN	MEDIUM	Vios	Interior	Abdul Latif Jameel	13	9	17	22	1	1
90916T2039	BELT, V-RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
15601BZ030	ELEMENT SUB-ASSY, OIL FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	118	0	310	145	793
111150Y030	GASKET, CYLINDER HEAD	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	125	3	128	498	486
17801BZ150	ELEMENT SUB-ASSY, AIR CLEANER FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	30	93	0	110	59	72
123050Y060	INSULATOR SUB-ASSY, ENGINE	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	11	4	70	15	34

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
	MOUNTING, RH										
514420D210	COVER, ENGINE UNDER, LH	MEDIUM	Vios	Engine	Abdul Latif Jameel	0	24	150	107	0	0
312100K310	COVER ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	237	149	181	776	383	150
488150K130	BUSH, STABILIZER	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	29	24	28	17	0	0
485208Z125	ABSORBER SET, SHOCK, FR LH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	0	0	1	172	19	99
485108Z265	ABSORBER SET, SHOCK, FR RH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	0	0	5	143	31	85
049470K020	FITTING KIT, DISC BRAKE	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	10	26	43	97	5
521190M913	COVER,FR BUMPER L/C	BIG	#N/A	Exterior	Abdul Latif Jameel	17	54	105	0	66	24
811500DG30	HEADLAMP ASSY, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	1	37	247	220	106	102
534100K430	HINGE ASSY, HOOD, RH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	0	0	0	57	68	88
531110K700	GRILLE, RADIATOR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	19	34	170	229	52	7

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
531120DA40	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	305	0	29
90916T2029	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	417	51	0	12	0
90916T2046	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	1	102	458	75
871030K440	MOTOR SUB- ASSY, BLOWER W/FAN	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	2	0	29	26	26
885900K030	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	1	0	62	0	0	0
556620D160	BEZEL, INSTRUMENT PANEL REGISTER, NO.2	MEDIUM	Vios	Interior	Abdul Latif Jameel	0	60	13	1	0	0
232200P210	PUMP ASSY, FUEL W/FILTER	MEDIUM	#N/A	Engine	Abdul Latif Jameel	1	2	9	4	5	4
12361BZ200	INSULATOR, ENGINE MOUNTING, FR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	1	1	0	2	7
514410D330	COVER, ENGINE UNDER, RH	MEDIUM	Vios	Engine	Abdul Latif Jameel	0	47	156	92	0	44

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2018.07	2018.08	2018.09	2018.10	2018.11	2018.12
23291BZ010	INSULATOR, INJECTOR VIBRATION	BIG	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	105	0	0	1	0	10
12371BZ170	INSULATOR, ENGINE MOUNTING, RR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	0	0	17	3	42
130410Y04002	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	585	61	0	0
312500K311	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	0	0	67	232	422
117010Y03003	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
130410Y04003	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	0	261	1508
117010C01101	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	27	636	1591	0	0	0

Lampiran 97 Permintaan Ekspor Bulanan Spare Part Ke Arab Saudi Selama 2018-2019 (Lanjutan)

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
534200K430	HINGE ASSY, HOOD, LH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	103	85	72	249	70	46

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
531120D360	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
531120K240	GRILLE, RADIATOR, LWR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	17	44	54	138	77	45
525360D190	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
526110D080	ABSORBER, FR BUMPER ENERGY	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
646070D250	CABLE SUB- ASSY, LUGGAGE DOOR LOCK CONT	MEDIUM	Vios	Engine	Abdul Latif Jameel	16	3	14	0	27	0
12361BZ162	INSULATOR, ENGINE MOUNTING, FR RH	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	5	11	4	13	16	12
521190U933	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	770	349	742	915	268	296
521590U933	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	221	101	498	337	211	272
525360D290	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	143	117	153	300	439	89

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
531120D780	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	165	250	0	49
525350D290	RETAINER, FR BUMPER SIDE, RH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	141	238	266	217	206	154
743200K450E0	VISOR A/S LH	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	9	0	0	0	0	0
83800FAC20	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	10	0	0	0	0	0
83800FAC21	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	17
636500K570B0	BOX A/S ROOF CONSOLE	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	0	9	8	0	0	0
16620BZ010	TENSIONER ASSY, V- RIBBED BELT	MEDIUM	Daihatsu Model (Avanza/Rush)	Interior	Abdul Latif Jameel	0	7	6	7	4	2
13101BZ100	PISTON SUB- ASSY, W/PIN	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	28	24	8	23	20	0
123720Y280	INSULATOR, ENGINE MOUNTING, LH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	17	0	0	0	0
132110Y011	PISTON	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	12	10	9	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
11101B9622	HEAD SUB-ASSY, CYLINDER	BIG	#N/A	Engine	Abdul Latif Jameel	14	0	0	9	0	0
123640Y010	ROD, ENGINE MOVING CONTROL, NO.2	MEDIUM	#N/A	Engine	Abdul Latif Jameel	7	0	7	7	13	0
312500K310	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
485100DD80	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	0	105	170	285	274	22
485100DC40	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	195	0	0	0	0	1
43512BZ280	DISC, FR	BIG	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	50	0	0	0	0	30
48815BZ290	BUSH, STABILIZER	MEDIUM	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	0	0	0	67	0	0
521190U920	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
521590U908	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
521590M911	COVER, RR BUMPER L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	127	5	174	282	153	55

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
534200D630	HINGE ASSY, HOOD, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	80	117	20	38	269	22
483040D170	BUMPER SUB-ASSY, FR SPRING	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	26	24	205	0
885900K050	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	80	25	8	57	3	4
871030D370	MOTOR SUB-ASSY, BLOWER W/FAN	MEDIUM	Vios	Interior	Abdul Latif Jameel	10	2	1	6	0	18
90916T2039	BELT, V-RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
15601BZ030	ELEMENT SUB-ASSY, OIL FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	900	180	370	570	820	28
111150Y030	GASKET, CYLINDER HEAD	MEDIUM	#N/A	Engine	Abdul Latif Jameel	197	184	179	709	161	39
17801BZ150	ELEMENT SUB-ASSY, AIR CLEANER FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	451	80	50	255	140	0
123050Y060	INSULATOR SUB-ASSY, ENGINE MOUNTING, RH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	125	24	13	85	74	192

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
514420D210	COVER, ENGINE UNDER, LH	MEDIUM	Vios	Engine	Abdul Latif Jameel	0	20	108	27	109	36
312100K310	COVER ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	545	284	1076	1167	447	19
488150K130	BUSH, STABILIZER	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	23	935	152	393	73	158
485208Z125	ABSORBER SET, SHOCK, FR LH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	53	33	56	77	129	61
485108Z265	ABSORBER SET, SHOCK, FR RH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	48	0	66	116	158	70
049470K020	FITTING KIT, DISC BRAKE	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	16	55	52	85	6	0
521190M913	COVER,FR BUMPER L/C	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	69	421	0	0
811500DG30	HEADLAMP ASSY, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	175	68	27	176	234	65
534100K430	HINGE ASSY, HOOD, RH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	100	143	61	223	269	55
531110K700	GRILLE, RADIATOR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	0	0	131	229	0	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
531120DA40	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	133	0	0	0	0	0
90916T2029	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	68	0	82	137
90916T2046	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	251	35	0	714	0	0
871030K440	MOTOR SUB- ASSY, BLOWER W/FAN	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	6	0	43	123	33	40
885900K030	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	0	0	0	0	0
556620D160	BEZEL, INSTRUMENT PANEL REGISTER, NO.2	MEDIUM	Vios	Interior	Abdul Latif Jameel	0	0	0	0	0	16
232200P210	PUMP ASSY, FUEL W/FILTER	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	5	8	9	4
12361BZ200	INSULATOR, ENGINE MOUNTING, FR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	9	3	9	234	2	0
514410D330	COVER, ENGINE UNDER, RH	MEDIUM	Vios	Engine	Abdul Latif Jameel	0	0	47	27	52	55

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.01	2019.02	2019.03	2019.04	2019.05	2019.06
23291BZ010	INSULATOR, INJECTOR VIBRATION	BIG	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	10	0	80	0	0
12371BZ170	INSULATOR, ENGINE MOUNTING, RR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	5	47	26	42	118	14
130410Y04002	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	3984	347	59
312500K311	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	1540	0	0	2993	0	183
117010Y03003	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	1332	2692	82	343
130410Y04003	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	249	0	0	0	0	0
117010C01101	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	10	0	0	0	0

Lampiran 97 Permintaan Ekspor Bulanan Spare Part Ke Arab Saudi Selama 2018-2019 (Lanjutan)

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
534200K430	HINGE ASSY, HOOD, LH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	63	0	43	117	183	62

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
531120D360	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
531120K240	GRILLE, RADIATOR, LWR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	36	0	0	0	0	0
525360D190	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
526110D080	ABSORBER, FR BUMPER ENERGY	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
646070D250	CABLE SUB- ASSY, LUGGAGE DOOR LOCK CONT	MEDIUM	Vios	Engine	Abdul Latif Jameel	0	0	28	14	0	0
12361BZ162	INSULATOR, ENGINE MOUNTING, FR RH	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	21	3	13	18	30	17
521190U933	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	481	0	63	0	429	228
521590U933	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	264	24	99	0	179	161
525360D290	RETAINER, FR BUMPER SIDE, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	96	0	14	157	187	136

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
531120D780	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	77	0	78	124	138	104
525350D290	RETAINER, FR BUMPER SIDE, RH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	102	22	107	195	227	231
743200K450E0	VISOR A/S LH	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	13	0	0	0	0	0
83800FAC20	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
83800FAC21	METER ASSY, COMBINATION	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	2	10	12	0	10
636500K570B0	BOX A/S ROOF CONSOLE	#N/A	Innova/Fortuner	Interior	Abdul Latif Jameel	0	1	0	0	4	3
16620BZ010	TENSIONER ASSY, V- RIBBED BELT	MEDIUM	Daihatsu Model (Avanza/Rush)	Interior	Abdul Latif Jameel	13	6	3	2	7	3
13101BZ100	PISTON SUB- ASSY, W/PIN	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	0	0	0	24	8
123720Y280	INSULATOR, ENGINE MOUNTING, LH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	0	0
132110Y011	PISTON	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	0	0

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
11101B9622	HEAD SUB-ASSY, CYLINDER	BIG	#N/A	Engine	Abdul Latif Jameel	0	0	0	0	10	0
123640Y010	ROD, ENGINE MOVING CONTROL, NO.2	MEDIUM	#N/A	Engine	Abdul Latif Jameel	0	0	7	11	1	10
312500K310	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	0	0	0	0	0	0
485100DD80	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	0	0	32	58	191	89
485100DC40	ABSORBER ASSY, SHOCK, FR	MEDIUM	Vios	Suspension	Abdul Latif Jameel	76	26	0	0	0	0
43512BZ280	DISC, FR	BIG	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	0	0	0	46	0	0
48815BZ290	BUSH, STABILIZER	MEDIUM	Daihatsu Model (Avanza/Rush)	Suspension	Abdul Latif Jameel	0	0	0	0	52	14
521190U920	COVER, FR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
521590U908	COVER, RR BUMPER, L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
521590M911	COVER, RR BUMPER L/COLOR	BIG	#N/A	Exterior	Abdul Latif Jameel	2	0	0	0	0	84

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
534200D630	HINGE ASSY, HOOD, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	170	87
483040D170	BUMPER SUB-ASSY, FR SPRING	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	8	0	0	0	0
885900K050	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	52	0	24	0	0	0
871030D370	MOTOR SUB-ASSY, BLOWER W/FAN	MEDIUM	Vios	Interior	Abdul Latif Jameel	24	15	9	8	21	4
90916T2039	BELT, V-RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	0	0	0	0
15601BZ030	ELEMENT SUB-ASSY, OIL FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	688	220	170	430	740	410
111150Y030	GASKET, CYLINDER HEAD	MEDIUM	#N/A	Engine	Abdul Latif Jameel	321	54	0	16	88	206
17801BZ150	ELEMENT SUB-ASSY, AIR CLEANER FILTER	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	20	0	10	30	140	197
123050Y060	INSULATOR SUB-ASSY, ENGINE MOUNTING, RH	MEDIUM	#N/A	Engine	Abdul Latif Jameel	108	0	4	131	253	283

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
514420D210	COVER, ENGINE UNDER, LH	MEDIUM	Vios	Engine	Abdul Latif Jameel	11	0	8	36	66	49
312100K310	COVER ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	51	13	13	169	538	570
488150K130	BUSH, STABILIZER	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	76	8	29	32	148	27
485208Z125	ABSORBER SET, SHOCK, FR LH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	116	0	31	125	44	191
485108Z265	ABSORBER SET, SHOCK, FR RH	MEDIUM	#N/A	Suspension	Abdul Latif Jameel	36	18	113	101	95	27
049470K020	FITTING KIT, DISC BRAKE	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	2	1	11	145	78	17
521190M913	COVER,FR BUMPER L/C	BIG	#N/A	Exterior	Abdul Latif Jameel	0	0	0	0	0	10
811500DG30	HEADLAMP ASSY, LH	MEDIUM	Vios	Exterior	Abdul Latif Jameel	6	0	0	32	62	63
534100K430	HINGE ASSY, HOOD, RH	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	25	0	0	0	94	79
531110K700	GRILLE, RADIATOR	MEDIUM	Innova/Fortuner	Exterior	Abdul Latif Jameel	0	0	0	0	0	11

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
531120DA40	GRILLE, RADIATOR, LWR	MEDIUM	Vios	Exterior	Abdul Latif Jameel	0	0	0	0	0	0
90916T2029	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	912	109	0	0	0	313
90916T2046	BELT, V- RIBBED	MEDIUM	#N/A	Interior	Abdul Latif Jameel	0	0	87	314	300	0
871030K440	MOTOR SUB- ASSY, BLOWER W/FAN	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	54	33	11	21	0	0
885900K030	BLOWER ASSY, W/SHROUD	MEDIUM	Innova/Fortuner	Interior	Abdul Latif Jameel	0	7	0	18	27	2
556620D160	BEZEL, INSTRUMENT PANEL REGISTER, NO.2	MEDIUM	Vios	Interior	Abdul Latif Jameel	0	0	0	0	0	0
232200P210	PUMP ASSY, FUEL W/FILTER	MEDIUM	#N/A	Engine	Abdul Latif Jameel	52	97	317	13	54	77
12361BZ200	INSULATOR, ENGINE MOUNTING, FR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	7	0	0	0	0	5
514410D330	COVER, ENGINE UNDER, RH	MEDIUM	Vios	Engine	Abdul Latif Jameel	24	0	0	0	4	42

NO PART	NAMA PART	DIMENSI	MODEL	FAMILY	TUJUAN	2019.07	2019.08	2019.09	2019.10	2019.11	2019.12
23291BZ010	INSULATOR, INJECTOR VIBRATION	BIG	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	0	0	0	0	0	0
12371BZ170	INSULATOR, ENGINE MOUNTING, RR	MEDIUM	Daihatsu Model (Avanza/Rush)	Engine	Abdul Latif Jameel	8	0	0	0	37	17
130410Y04002	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	2	17	1	0	0
312500K311	DISC ASSY, CLUTCH	MEDIUM	Innova/Fortuner	Suspension	Abdul Latif Jameel	99	33	12	0	0	0
117010Y03003	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	47	3	0	0	0	0
130410Y04003	BEARING SET, CON ROD	#N/A	#N/A	Suspension	Abdul Latif Jameel	6	0	0	0	92	1762
117010C01101	BEARING SET, C/SHAFT	#N/A	#N/A	Suspension	Abdul Latif Jameel	0	0	0	0	0	0