



DISERTASI - IF186601

**MODEL PENDUGA CURAH HUJAN ZONA MUSIM
MENGUNAKAN PREDIKTOR SIMULTAN
DAN HYBRID ARTIFICIAL NEURAL NETWORKS
(Studi Kasus: Wilayah Propinsi Jawa Timur)**

Ahmad Saikhu
NRP. 05111560010007

DOSEN PEMBIMBING
Dr. Agus Zainal Arifin, S.Kom, M.Kom
Dr. Eng. Chastine Fatichah, S.Kom, M.Kom

PROGRAM DOKTOR ILMU KOMPUTER
DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2019

PENGESAHAN

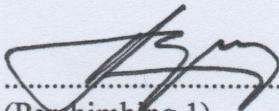
Disertasi disusun untuk memenuhi salah satu syarat memperoleh gelar
Doktor Ilmu Komputer (Dr.)
di
Institut Teknologi Sepuluh Nopember

Oleh:
Ahmad Saikhu
NRP. 05111560010007

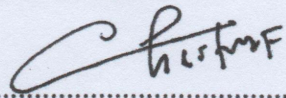
Tanggal Ujian: 19 Juni 2019
Periode Wisuda: September 2019

Disetujui oleh:

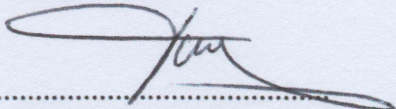
1. Dr. Agus Zainal Arifin, S.Kom, M.Kom
NIP. 19720809 199512 1 001


.....
(Pembimbing 1)

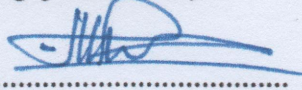
2. Dr. Eng. Chastine Faticah, S.Kom, M.Kom
NIP. 19751220 2001 12 2 002


.....
(Pembimbing 2)


3. Drs. Edi Winarko, M.Sc, Ph.D
NIP. 19630223 1987 03 1 002


.....
(Penguji Eksternal)

4. Dr. Suhartono, M.Sc.
NIP. 19710929 1995 12 1 001



.....
(Penguji 1)

5. Dr. Eng. Darlis Herumurti, S.Kom, M. Kom
NIP. 19771217 2003 12 1 001


.....
(Penguji 2)



Kepala Departemen Informatika
Fakultas Teknologi Informasi dan Komunikasi


Dr. Eng. Darlis Herumurti, S.Kom, M.Kom
NIP. 19771217 2003 12 1 001

Halaman ini sengaja dikosongkan

PERNYATAAN KEASLIAN

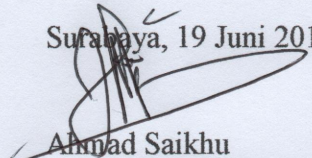
Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Disertasi saya dengan judul:

**MODEL PENDUGA CURAH HUJAN ZONA MUSIM
MENGUNAKAN PREDIKTOR SIMULTAN
DAN HYBRID ARTIFICIAL NEURAL NETWORKS
(Studi Kasus: Wilayah Propinsi Jawa Timur)**

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Surabaya, 19 Juni 2019


Ahmad Saikhu
NRP. 05111560010007

Halaman ini sengaja dikosongkan

**MODEL PENDUGA CURAH HUJAN ZONA MUSIM
MENGUNAKAN PREDIKTOR SIMULTAN
DAN HYBRID ARTIFICIAL NEURAL NETWORKS
(Studi Kasus: Wilayah Propinsi Jawa Timur)**

ABSTRAK

Pemodelan curah hujan cukup penting untuk perencanaan sumber daya air, pertanian, infrastruktur, pertambangan dan merupakan penelitian penting di bidang hidrologi. Penelitian pemodelan curah hujan, diantaranya adalah model konseptual yang bertumpu pada interaksi dan proses antara dinamika atmosfer, suhu permukaan laut, dan posisi matahari sehingga memiliki kompleksitas tinggi dan berbiaya mahal. Kedua adalah model empiris, dipilih untuk menemukan hubungan data input dan output ketika pengguna tidak memahami secara lengkap hubungan faktor-faktor fisik yang terjadi dan berfokus pada akurasi.

Curah hujan di Indonesia dipengaruhi faktor lokal (cuaca) dan iklim regional (global) sehingga pemilihan prediktor yang relevan adalah faktor yang penting untuk efisiensi. Model curah hujan di suatu wilayah yang memiliki akurasi tinggi belum tentu sesuai untuk wilayah lainnya karena model tersebut bersifat khas akibat dari sejumlah faktor. Model curah hujan wilayah tropis yang mampu mensimulasikan dan memprediksi curah hujan di Indonesia secara efektif sangat dibutuhkan.

Model empiris ARIMA memiliki kelemahan kurang sesuai untuk data non-linier sehingga akurasinya rendah. Untuk mengatasi kekurangan ini, pilihannya adalah metode ANN yang juga memiliki kelemahan tidak menjamin akurasi yang optimal karena bergantung pada karakteristik data. Untuk mengatasi kekurangan ARIMA maupun ANN adalah dengan mengidentifikasi sebanyak mungkin variabel prediktor dengan mempertimbangkan faktor waktu dan lokasi (*spatio-temporal*), pola hubungan linier/non-linier dan fitur lokal/global.

Karenanya dalam penelitian ini dibangun pendekatan baru model penduga curah hujan wilayah Zona Musim di Indonesia melalui kombinasi proses ekstraksi, seleksi variabel prediktor dan penggabungan model GSTAR dan RNN.

Metode DPCCA digunakan untuk ekstraksi fitur, sedangkan untuk seleksi prediktor yang relevan, digunakan metode PCSUF. Optimalitas hasil seleksi prediktor dievaluasi dengan *trace merit-value* (MV) dan % jumlah prediktor. Kinerja model GSTAR, *spatio-temporal* RNN dan *hybrid* GSTAR-RNN diukur dengan AIC-corrected, RMSE, *r* dan R^2 .

Pengujian kinerja dari metode seleksi fitur dan model peramalan yang diusulkan, dipilih dua wilayah, yaitu Surabaya dengan tiga stasiun cuaca dan Malang-Pasuruan dengan tiga stasiun cuaca, masing-masing untuk data periode harian dan bulanan dengan 11 variabel prediktor. Studi kasus diselesaikan dengan 12 variasi model untuk memperoleh model terbaik.

Dari pengujian empat studi kasus, diperoleh bahwa metode DPCCA dan PCSUF cukup efektif untuk memperoleh prediktor *spatio-temporal* yang optimal berdasarkan ukuran *trace-MV* dan % jumlah prediktor. Hal ini dibuktikan dengan kinerja pemodelan peramalan curah hujan menggunakan *spatio-temporal* RNN di mana nilai MSE, *r* dan R^2 testing yang lebih baik dibandingkan tanpa PCSUF. Demikian juga, *spatio-temporal* RNN memiliki kinerja lebih baik dibandingkan dengan GSTAR maupun *Hybrid* GSTAR-RNN.

Kata Kunci: curah hujan, *merit-value*, DPCCA, PCSUF, GSTAR, ST-RNN

RAINFALL MODELING IN SEASON ZONE USING SIMULTANEOUS PREDICTORS AND HYBRID ARTIFICIAL NEURAL NETWORKS (Case Study: Region in East Java Province)

ABSTRACT

Rainfall modeling is quite important for water resource planning, agriculture, infrastructure, mining and hydrology research interest. Research in rainfall modeling, such as a conceptual model that relies on interactions and processes between atmospheric dynamics, sea-surface temperature, and sun position so that they have highly complexity and expensive cost. Another is the empirical model, which is find the relationship between input and output data when the user does not fully understand the relationship between physical factors and focused on accuracy.

Rainfall in Indonesia is influenced by weather and global climates thus the selection of relevant predictors is important factor for efficiency. Rainfall models in a certain area that has high accuracy are not necessarily suitable for other regions because that model has unique characteristic due to several factors. Rainfall model in tropical regions that able to effectively simulate and predict rainfall in Indonesia have not been proposed.

The ARIMA empirical model is unsuitable for non-linear data, thus it produces a low accuracy. Meanwhile ANN method also cannot guarantee an optimal accuracy since it depends on the characteristics of data. In order to overcome those weaknesses, we have to identify predictor variables as many as possible by considering time and location factors (spatiotemporal), linear/non-linear relationship patterns and local/global features.

Therefore, in this study, a new approach is developed for the Seasonal Zone rainfall estimator model in Indonesia through a combination of extraction processes, predictor variables selection and a combination of GSTAR and RNN models.

The DPCCA method is used for feature extraction, while for the relevant predictors selection, the PCSUF method is used. Optimality of predictor selection results is evaluated by trace merit-value (MV) and% number of predictors. The performance of the GSTAR, spatiotemporal RNN, and hybrid models is indicated by AIC-corrected, RMSE, r and R2.

Performance evaluation of the proposed feature selection methods and forecasting model, has been acquired from the two selected regions, Surabaya and Malang-Pasuruan with three weather stations for each city. The data is collected daily and monthly basis with 11 predictor variables. Case studies were completed with 12 variations of models to obtain the best model.

As a result from 4 case studies, DPCCA and PCSUF methods are able to effectively obtained an optimal spatio-temporal predictors based on trace-MV measurements. It proved by the performance of rainfall forecasting modeling using spatiotemporal RNN where the MSE, r and R2 testing values are better than without PCSUF. Moreover, spatiotemporal RNN has a better performance compared to GSTAR and hybrid GSTAR-RNN.

Keywords: rainfall, merit-value, DPCCA, PCSUF, GSTAR, ST-RNN

DAFTAR ISI

PENGESAHAN	iii
PERNYATAAN KEASLIAN.....	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI.....	vii
DAFTAR TABEL	xi
DAFTAR GAMBAR	xiii
DAFTAR SIMBOL.....	xvii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	5
1.3 Batasan Masalah	5
1.4 Tujuan dan Manfaat Penelitian	6
1.5 Kontribusi Penelitian	6
BAB 2 KAJIAN PUSTAKA DAN DASAR TEORI.....	7
2.1 Kajian Pustaka	7
2.2 Dasar Teori.....	13
BAB 3 METODE PENELITIAN.....	31
3.1 Tahap Interpolasi dan Agregasi	33
3.2 Tahap Ekstraksi dan Seleksi fitur	34
3.3 Pemodelan GSTAR.....	38
3.4 Pemodelan spatio-temporal RNN	39
3.5 Pemodelan hybrid GSTAR dan RNN (GSTAR-RNN)	41
3.6 Evaluasi Kinerja.....	43
3.7 Studi Kasus dan Uji Model	44
BAB 4 HASIL DAN PEMBAHASAN.....	47
4.1 Karakteristik Studi Kasus	47

4.2 Ekstraksi Fitur menggunakan metode DPCCA	49
4.3 Seleksi Fitur menggunakan metode PCSUF	52
4.4 Model Penduga Curah Hujan dengan GSTAR.....	55
4.5 Pemodelan ST-RNN	60
4.6 Pemodelan Data Curah Hujan dengan <i>Hybrid</i> GSTAR-RNN.....	65
4.7 Model Penduga Curah Hujan bulanan Wilayah Malang-Pasuruan	69
4.8 Model Penduga Curah Hujan periode harian di Wilayah Surabaya	77
4.9 Model Penduga Curah Hujan periode harian Wilayah Malang-Pasuruan...	84
BAB 5 KESIMPULAN DAN SARAN	89
5.1 Kesimpulan.....	89
5.2 Saran	90
DAFTAR PUSTAKA	91
Lampiran I : Data.....	103
Lampiran II : Program	108
Lampiran III : Output Progam Studi Kasus 1	136

DAFTAR TABEL

Tabel 3.1 Sumber data cuaca dan indeks iklim.....	31
Tabel 3.2 Deskripsi Model GSTAR, RNN, dan hybrid	45
Tabel 4.1 Nilai ρ_{DPCCA} dari lokasi 1	51
Tabel 4.2 Kandidat prediktor untuk curah hujan di Juanda ($X_{1, t(1)}$).....	52
Tabel 4.3 Fitur prediktor hasil seleksi dengan SUF.....	53
Tabel 4.4 Perbandingan kinerja seleksi fitur.....	54
Tabel 4.5 Kinerja model GSTAR berdasarkan metode bobot	58
Tabel 4.6 Kinerja model RNN berdasarkan seleksi fitur	63
Tabel 4.7 Seleksi fitur untuk model hybrid GSTAR-RNN.....	65
Tabel 4.8 Kinerja model GSTAR dan hybrid GSTAR-RNN	66
Tabel 4.9 Kinerja model GSTAR, RNN dan hybrid GSTAR-RNN.....	68
Tabel 4.10 Kinerja model GSTAR untuk studi kasus 2.....	71
Tabel 4.11 Kinerja model untuk studi kasus 2.....	76
Tabel 4.12 Kinerja model GSTAR untuk studi kasus 3.....	78
Tabel 4.13 Kinerja model ST-RNN untuk studi kasus 3	81
Tabel 4.14 Kinerja GSTAR untuk studi kasus 4.....	86
Tabel 4.15 Seleksi fitur untuk studi kasus 4	87

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2.1 Unsur-unsur cuaca dan iklim [124].....	14
Gambar 2.2 Pembagian ZoM di wilayah Jawa Timur (BMKG).....	15
Gambar 2.3 Zona ENSO di samudra pasifik [127].....	16
Gambar 2.4 Contoh data runtun waktu meteorologi [129]	17
Gambar 2.5 Data suhu maksimum berpola stasioner [130].....	17
Gambar 2.6 Data runtun waktu non-stasioner [129].....	18
Gambar 2.7 ACF dan PACF dari curah hujan di Surabaya [129].....	19
Gambar 2.8 <i>Correlogram</i> CCF antara curah hujan dan suhu minimum [133]	19
Gambar 2.9 DPCCA antara curah hujan, Nino3 dan PDO [96]	21
Gambar 2.10 Fitur prediktor yang relevan dan tidak relevan [103].....	23
Gambar 2.11 Recurrent Neural Networks [144].....	27
Gambar 3.1 Metodologi pengembangan model penduga curah hujan	32
Gambar 3.2 Diagram alir untuk ekstraksi fitur <i>time-lags</i> dengan DPCCA [96]...	35
Gambar 3.3 Seleksi fitur relevan dan hapus fitur redundan.....	37
Gambar 3.4 Diagram alir model GSTAR	39
Gambar 3.5 Arsitektur ST-RNN	40
Gambar 3.6 Proses pemodelan ST-RNN	41
Gambar 3.7 Model hybrid GSTAR-ANN.....	42
Gambar 3.8 Proses pemodelan hybrid GSTAR-RNN	43
Gambar 4.1 Pola curah hujan periode bulanan pada 3 lokasi	48
Gambar 4.2 Uji korelasi data curah hujan pada 3 lokasi	49
Gambar 4.3 Perbandingan DPCCA dengan CCF	51
Gambar 4.4 ADF-test untuk data curah hujan di 3 lokasi.....	56
Gambar 4.5 ACF dan PACF Curah Hujan di Juanda	56
Gambar 4.6 Perhitungan nilai bobot dari GSTAR.....	56
Gambar 4.7 Estimasi parameter GSTAR.....	57
Gambar 4.8 Peramalan GSTAR untuk curah hujan di Juanda.....	58
Gambar 4.9 Peramalan GSTAR untuk curah hujan di Perak 1.....	58
Gambar 4.10 Peramalan GSTAR untuk curah hujan di Perak 2.....	59
Gambar 4.11 ACF dari residual model GSTAR untuk wilayah Juanda	59

Gambar 4.12 ACF dari residual model GSTAR untuk wilayah Perak 1	59
Gambar 4.13 ACF dari residual model GSTAR untuk wilayah Perak 2.....	60
Gambar 4.14 Arsitektur dan parameter RNN	60
Gambar 4.15 Visualisasi Jaringan untuk ST-RNN lokasi Juanda	61
Gambar 4.16 Hasil training untuk curah hujan di wilayah Juanda.....	62
Gambar 4.17 Hasil training untuk curah hujan di wilayah perak 1	62
Gambar 4.18 Hasil training untuk curah hujan di wilayah Perak 2.....	62
Gambar 4.19 Peramalan dengan ST-RNN untuk curah hujan di Juanda	64
Gambar 4.20 Peramalan dengan ST-RNN untuk curah hujan di Perak 1	64
Gambar 4.21 Peramalan dengan ST-RNN untuk curah hujan di Perak 2	64
Gambar 4.22 Arsitektur RNN dari error GSTAR.....	66
Gambar 4.23 Peramalan hybrid GSTAR-RNN untuk curah hujan di Juanda	67
Gambar 4.24 Peramalan hybrid GSTAR-RNN untuk curah hujan di Perak 1	67
Gambar 4.25 Peramalan hybrid GSTAR-RNN untuk curah hujan di Perak 2	68
Gambar 4.26 Nilai RMSE dan Jumlah Fitur berdasarkan Metode.....	69
Gambar 4.27 Korelasi data curah hujan pada 3 lokasi	70
Gambar 4.28 Hasil uji stasioner untuk studi kasus 2.....	70
Gambar 4.29 Correlogram ACF dan PACF studi kasus 2.....	71
Gambar 4.30 Perhitungan nilai bobot dengan 2 metode dan kinerja.....	71
Gambar 4.31 Plot data aktual-testing curah hujan di Karangates.....	72
Gambar 4.32 Plot data aktual-training curah hujan di Karangploso	72
Gambar 4.33 Plot data aktual-training curah hujan di Tretes.....	72
Gambar 4.34 Arsitektur RNN untuk studi kasus 2.....	73
Gambar 4.35 Arsitektur RNN untuk prediksi curah hujan di Karangates.....	73
Gambar 4.36 Hasil training untuk curah hujan di Karangates	74
Gambar 4.37 Hasil proses training curah hujan di Karangploso	74
Gambar 4.38 Hasil proses training curah hujan di Tretes	75
Gambar 4.39 Hasil peramalan curah hujan di Karangates	75
Gambar 4.40 Hasil peramalan curah hujan di Karangploso	76
Gambar 4.41 Hasil peramalan curah hujan di Tretes	76
Gambar 4.42 Korelasi curah hujan di 3 lokasi stasiun cuaca	77
Gambar 4.43 Pengujian stasioneritas data curah hujan	78

Gambar 4.44 Hasil training studi kasus 3 – data Juanda	79
Gambar 4.45 Hasil training studi kasus 3 – data Perak 1	79
Gambar 4.46 Hasil training studi kasus 3 – data Perak 2	79
Gambar 4.47 Hasil testing studi kasus 3 – data Juanda	80
Gambar 4.48 Hasil testing studi kasus 3 – data Perak 1	80
Gambar 4.49 Hasil testing studi kasus 3 – data Perak 2	80
Gambar 4.50 Arsitektur RNN untuk studi kasus 3	81
Gambar 4.51 Hasil training metode ST-RNN untuk data Juanda.....	82
Gambar 4.52 Hasil training metode ST-RNN untuk data Perak1	82
Gambar 4.53 Hasil training metode ST-RNN untuk data Perak 2.....	83
Gambar 4.54 Hasil testing metode ST-RNN untuk data Juanda.....	83
Gambar 4.55 Hasil testing metode ST-RNN untuk data Perak 1	84
Gambar 4.56 Hasil testing metode ST-RNN untuk data Perak 2	84
Gambar 4.57 Korelasi curah hujan di wilayah Malang-Pasuruan	85
Gambar 4.58 Pengujian stasioneritas data curah hujan studi kasus 4.....	85
Gambar 4.59 ACF dari residual GSTAR wilayah Karangkates	86
Gambar 4.60 ACF dari residual GSTAR wilayah Karangploso.....	86
Gambar 4.61 ACF dari residual GSTAR wilayah Tretes	87

Halaman ini sengaja dikosongkan

DAFTAR SIMBOL

$X_{i(t)}$:	Data runtun waktu multivariat variabel ke- i , waktu ke- t
$r_{x,y(T)}$:	Korelasi sampel antara variabel x dan y pada periode ke- T
$\sigma_{xy}(T)$:	Kovariansi antara variabel x dan y pada periode ke- T
n	:	Jumlah data (periode) runtun waktu
m	:	Jumlah variabel dari data runtun waktu multivariat
t	:	Periode waktu
k	:	<i>Time-lags</i>
s	:	Lokasi ke- s dari <i>spatio-temporal</i>
P_k^j	:	Profil data runtun waktu variabel ke- j dengan <i>time-lags</i> ke- k
ρ_{DPCCA}	:	Matrik koefisien dari <i>Detrended Partial Cross Correlation</i>
ρ_{xy}	:	Matrik koefisien dari <i>Pearson's Correlation</i>
$C(s)$:	Matrik invers dari koefisien <i>Cross Correlation</i>
$H(X)$:	<i>Entropy</i> dari variabel X
$SU(X, Y)$:	Koefisien <i>Symmetrical Uncertainty</i> antara variabel X dan Y
p	:	Orde dari model <i>Autoregressive (AR)</i>
d	:	Orde dari <i>differencing</i> data runtun waktu
q	:	Orde dari model <i>Moving Average (MA)</i>
w_{ij}	:	Koefisien dari matrik bobot dalam model GSTAR
λ_k	:	Orde <i>spatial</i> dari model GSTAR
L_t	:	Data runtun waktu komponen linier dalam model <i>hybrid</i>
N_t	:	Data runtun waktu komponen non-linier dalam model <i>hybrid</i>
e_t	:	Residual/error dari model GSTAR
$RMSE$:	Nilai <i>Root Mean Square Error</i>
R_{sq}	:	Indeks <i>R-square (%)</i> sebagai ukuran akurasi model
$AICc$:	Nilai dari <i>Akaike Information Criterion corrected</i>
$\pi_{x_i, y}$:	Matriks merit-value

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Cuaca, iklim dan musim senantiasa akrab dalam kehidupan manusia sehari-hari. Gejala alam tersebut berpengaruh kepada banyak sektor kehidupan, diantaranya pertanian, kehutanan, kemaritiman, perikanan, kesehatan, transportasi, pariwisata, pertambangan, energi dan infrastruktur. Dalam kondisi yang ekstrim, cuaca dan iklim bahkan dapat menyebabkan terjadinya bencana, diantaranya wabah penyakit, banjir, angin kencang, tanah longsor, kekeringan dan kebakaran hutan/lahan. Oleh karenanya, informasi, analisis dan pengetahuan terhadap kondisi cuaca dan iklim serta pola perubahan iklim sangat penting untuk dipelajari agar manusia dapat beradaptasi dengan perubahan yang dinamis, menghindarkan atau mengurangi terjadinya resiko akibat bencana dan melakukan tindakan mitigasi.

Dalam mengenali pola cuaca dan iklim dilakukan pengukuran terhadap suhu udara, kelembaban relatif, tekanan udara, penguapan air, kecepatan angin, suhu permukaan laut, lamanya penyinaran matahari dan curah hujan [1]. Nilai variabel tersebut berbeda bergantung pada sejumlah faktor, yaitu ketinggian tempat, garis lintang, tekanan, arus laut dan permukaan tanah. Batas nilai variabel untuk keadaan yang stabil telah ditentukan, namun dapat terjadi penyimpangan yang mengakibatkan masalah tertentu, misalnya hujan dan suhu ekstrim.

Salah satu komponen penting dalam cuaca dan iklim adalah curah hujan. Curah hujan di suatu wilayah diakibatkan oleh berbagai variabel cuaca dan iklim global. Curah hujan merupakan sistem yang kompleks di mana hubungan antar variabel bersifat simultan dan dipengaruhi waktu, wilayah dan kondisi iklim global yang dinamis. Oleh karenanya, hubungan antar variabel tidak hanya bersifat sesaat tetapi terkait dengan kondisi wilayah sekitar dan observasi cuaca/iklim pada periode sebelumnya.

Pemodelan curah hujan di suatu wilayah cukup penting untuk perencanaan sumber daya air secara optimal, perkiraan cadangan air di musim kering/kemarau, kegiatan pertambangan dan merupakan penelitian penting dalam bidang hidrologi.

Berbagai penelitian telah dilakukan untuk pemodelan curah hujan, baik pendekatan model teoritis/konseptual maupun pendekatan data empiris/histori. Pemodelan konseptual bertumpu pada interaksi/proses fisis dinamika atmosfer, suhu permukaan laut, dan posisi matahari [2].

Model empiris efektif untuk menemukan hubungan antara data input dan output ketika pengguna tidak memahami secara lengkap hubungan faktor-faktor fisik yang terjadi [3], namun lebih mementingkan nilai akurasi prediksi curah hujan. Di samping itu, pendekatan konseptual memerlukan proses komputasi yang mahal, kompleksitas tinggi dan padat peralatan *scientific* yang dalam beberapa tahun memerlukan pembaharuan peralatan [4].

Saat ini pemodelan empiris yang banyak digunakan untuk prediksi curah hujan adalah pendekatan statistik dan *data mining/machine learning*. Prediksi curah hujan dengan metode statistik adalah menggunakan pendekatan *time-series*, antara lain dengan *Autoregressive Integrated Moving Average (ARIMA)* dan *Seasonal ARIMA with Exogenous Input (SARIMAX)*. Pendekatan ini memiliki persyaratan diantaranya data bersifat stasioner dan pola hubungan yang linier [3], [5]. Sedangkan dalam permasalahan hidrologi dan lingkungan, pola data lebih banyak bersifat non linier [6], [7].

Untuk memperbaiki kelemahan dari model statistik, telah dikembangkan berbagai model prediksi curah hujan yang berbasis pada kecerdasan buatan, baik menggunakan metode *machine-learning* tunggal maupun gabungan (*hybrid*) dari metode statistik dan *machine-learning*. Metode *machine-learning* untuk prediksi curah hujan harian/bulanan diantaranya adalah *Bayesian Network* untuk analisis cuaca di wilayah India dan Canada [8], digunakannya *k-NN* dan *Mutual Information (MI)* untuk prediksi curah hujan dari data meteorologi di wilayah Dalian, China [9].

Selanjutnya, Ramana dkk. [10] menggunakan metode *Wavelet Neural Network* untuk data curah hujan bulanan di wilayah India. *Support Vector Machine* juga digunakan untuk prediksi curah hujan harian [11], [12]. Pendekatan *Neuro-Fuzzy* antara lain dilakukan oleh Hashim dkk. [13] untuk prediksi curah hujan bulanan di wilayah Patna, India. Pendekatan *Machine Learning* memiliki sifat dapat mengakomodasi pola non linier yang cukup baik.

Pendekatan model *hybrid* dan *ensemble* antara lain adalah dengan menggabungkan metode *Self Organized Map* (SOM) dan *Non-linear Autoregressive with Exogenous Inputs* (NARX) [14], ataupun menggabungkan *Multiple Linear Regression* (MLR) dan *Extreme Learning Machine* (ELM) untuk memprediksi curah hujan bulanan dengan prediktor anomali suhu permukaan laut [15]. Model hybrid ARIMA-NARX digunakan oleh Banihabib dkk. [16] untuk prediksi curah hujan harian dengan prediktor *North Atlantic Oscillation* (NAO). Penggabungan *Artificial Neural Networks* (ANN), MI dan *Particle Swarm Optimization* (PSO) digunakan untuk prediksi curah hujan bulanan di wilayah Australia [17]. Asadi dkk. [3] menggabungkan ANN dengan *Genetic Algorithm* (GA) dan algoritma *Levenverg-Marquardt* (LM) untuk prediksi curah hujan harian di wilayah India.

Pemodelan curah hujan dengan studi kasus wilayah Indonesia telah dilakukan dengan pendekatan ARIMA maupun *machine-learning*. Sejumlah penelitian prediksi curah hujan di wilayah Indonesia antara lain dengan studi kasus di Semarang, Tenggarong, Jakarta, Bandung dan Malang [18]–[26]. Metode yang digunakan antara lain ARIMA, ANN, ANFIS, sistem *Fuzzy* dan *Support Vector Regression*.

Berdasarkan sejumlah studi, dinyatakan bahwa curah hujan di Indonesia dipengaruhi oleh berbagai faktor lokal (cuaca) dan iklim regional/global [27]–[31]. Oleh karenanya penentuan variabel prediktor yang relevan adalah penting untuk efisiensi model dengan mempertimbangkan pola hubungan antar variabel yang kompleks.

Meskipun beberapa penelitian pemodelan iklim di suatu wilayah/negara memiliki akurasi yang tinggi, belum tentu model tersebut sesuai untuk wilayah lainnya dikarenakan perbedaan iklim, posisi geografis, topografi, dan proporsi daratan/maritim [32]. Sebagai contoh, model iklim untuk wilayah/negara lintang tinggi (sub tropis), akan berbeda dengan negara pada lintang rendah (tropis). Hal ini menyebabkan bahwa pemodelan untuk iklim dan prediksi hujan menjadi khas untuk suatu wilayah.

Berdasarkan penelitian yang berkaitan dengan pemodelan iklim dan prediksi curah hujan dengan pendekatan empiris, termasuk juga wilayah Indonesia,

masih menggunakan fitur cuaca/iklim secara parsial. Di sisi lain, secara teori bahwa iklim global, diantaranya *El Nino Southern Oscillation* (ENSO), *Index Oscillation Dipole* (IOD), *Indian Ocean Dipole* (IOD) dan *Dipole Mode Index* (DMI) secara teoritis sangat mempengaruhi curah hujan di Indonesia, utamanya wilayah Jawa dan Bali selain faktor lokal suatu wilayah [24], [29], [33], [34]. Kondisi atmosfer lokal dan faktor topografi suatu wilayah juga turut mempengaruhi curah hujan di wilayah tersebut [31], [35]–[37].

Pendekatan *spatio-temporal* digunakan untuk memodelkan data curah hujan bulanan yang diperoleh dari sejumlah stasiun cuaca di wilayah Australia untuk menghasilkan nilai prediksi. Model ini merupakan model *spatio-temporal* hierarkis yang diterapkan pada data geografis [38]. Metode korelasi *stepwise* berbasis *spatio-temporal* digunakan untuk pemodelan data cuaca bulanan berdasarkan observasi meteorologi lokal di wilayah Korea [39].

Kondisi topografi wilayah Indonesia yang bergunung, berlembah, serta banyak pantai, merupakan fenomena lokal dari negara lintang rendah sehingga model prediksi curah hujan menjadi cukup kompleks. Hal tersebut menyebabkan model iklim wilayah tropis yang mampu mensimulasikan dan memprediksi curah hujan di Indonesia secara efektif belum ditemukan [40].

Oleh karenanya dalam penelitian ini diusulkan suatu pendekatan baru model penduga curah hujan wilayah Zona Musim (ZoM) di Indonesia melalui kombinasi proses ekstraksi dan seleksi variabel prediktor serta penggabungan (*hybrid*) model *spatio-temporal* GSTAR dan RNN. Pendekatan dimaksud yaitu dengan melibatkan sejumlah fitur cuaca dan iklim yang berpola linier dan non linier dalam lingkup observasi cuaca (lokal) dan indeks iklim (global) dengan memperhitungkan faktor *time-lag* (*temporal*) dan lokasi (*spatial*).

Dengan pendekatan baru, baik dari segi prediktor yang simultan maupun model *spatio-temporal* diharapkan hasil prediksi curah hujan lebih akurat sehingga dapat digunakan sebagai informasi perencanaan di sejumlah bidang, antara lain pertanian, pertambangan, sumber daya air. Pemilihan prediktor dengan hanya melibatkan fitur relevan (signifikan) menjadikan proses komputasi lebih efisien sehingga informasi untuk pengambilan keputusan dapat dilakukan lebih cepat.

1.2 Perumusan Masalah

Permasalahan pada penelitian ini adalah:

- a. Bagaimana melakukan ekstraksi fitur tenggang waktu (*time-lags*) dan fitur lokasi (*spatial*) dari variabel prediktor yang berhubungan dengan variabel curah hujan?
- b. Bagaimana melakukan seleksi prediktor untuk model penduga curah hujan sehingga terpilih fitur yang relevan secara linier dan non linier?
- c. Bagaimana membangun model penduga curah hujan yang mampu mengintegrasikan variabel lokal/global, linier/non-linier, dan spatio-temporal?

1.3 Batasan Masalah

Dalam penelitian ini, data yang digunakan yaitu observasi cuaca harian yang diperoleh dari BMKG (<http://dataonline.bmkg.go.id/>) untuk wilayah Surabaya dan Malang, di Jawa Timur. Untuk wilayah Surabaya, meliputi Stasiun Meteorologi Kelas I, Juanda, Stasiun Meteorologi Perak I dan Stasiun Meteorologi Maritim Perak II. Sedangkan untuk wilayah Malang-Pasuruan adalah Stasiun Geofisika kelas II Karangates, Stasiun Klimatologi Kelas II Karang Ploso, dan Stasiun Geofisika kelas II Tretes.

Data indeks iklim global periode harian dan per bulan diperoleh dari <http://climexp.knmi.nl>. Periode data observasi yang digunakan untuk mengembangkan model penduga curah hujan dan pengujian adalah periode Januari 1983 – Desember 2016 (34 tahun).

Variabel prediktor lokal yang digunakan dari data BMKG adalah suhu minimum/rata-rata/maksimum, kelembaban udara, lama penyinaran, kecepatan angin dan maksimum kecepatan angin. Sedangkan variabel prediktor global yang digunakan dari data KNMI adalah data anomali suhu permukaan laut ENSO (Nino1.2, Nino3, Nino3.4, Nino4). Model penduga dibangun dengan menggunakan perangkat lunak R Studio versi 3.4.3.

1.4 Tujuan dan Manfaat Penelitian

Tujuan dari penelitian ini adalah membangun model penduga curah hujan di wilayah ZoM melalui kombinasi proses ekstraksi dan seleksi variabel prediktor, penggabungan (*hybrid*) GSTAR dan Jaringan Syaraf Tiruan untuk meningkatkan akurasi prediksi.

Proses ekstraksi dan seleksi fitur dimaksud adalah menemukan kandidat fitur *time-lags* dan *spatial* dari prediktor lokal/global, menggunakan metode *Detrended Partial Cross-Correlation Analysis* (DPCCA), dan melakukan seleksi prediktor baik hubungan linier maupun non linier menggunakan metode *Pearson-Correlation and Symmetrical Uncertainty* (PCSUF). Hal ini bertujuan untuk memperoleh input prediktor yang relevan dan tidak redundan sehingga proses pemodelan menjadi efisien. Sedangkan model peramalan curah hujan yang dikembangkan adalah *Spatio-Temporal Recurrent Neural Networks* (ST-RNN) dan *hybrid* GSTAR-RNN.

Manfaat dari penelitian ini adalah:

- a. Tersedianya metode seleksi prediktor yang mampu mengidentifikasi fitur linier dan non linier terhadap variabel respon.
- b. Tersedianya model penduga curah hujan berbasis spatio-temporal untuk periode harian/bulanan yang melibatkan sejumlah variabel input secara simultan, baik variabel lokal/global, waktu maupun wilayah.
- c. Dapat digeneralisasi untuk wilayah ZoM di luar Jawa Timur untuk memprediksi curah hujan harian/per bulan, di mana 84% wilayah Indonesia adalah ZoM.

1.5 Kontribusi Penelitian

Kontribusi penelitian ini adalah:

- a. Seleksi fitur yang memisahkan prediktor curah hujan berpola linier, non linier atau gabungan keduanya dan memilih himpunan prediktor yang optimal, baik tingkat relevansi maupun jumlah prediktornya.
- b. Model penduga curah hujan ST-RNN maupun *Hybrid* GSTAR-RNN yang mampu mengintegrasikan input variabel lokal/global, linier/non-linier, dan spatio-temporal.

BAB 2

KAJIAN PUSTAKA DAN DASAR TEORI

2.1 Kajian Pustaka

2.1.1 Penelitian terkait model penduga curah hujan

Pemodelan curah hujan merupakan bagian penting dari penelitian bidang hidrologi dan lingkungan. Sejumlah penelitian terkait hidrologi, antara lain peramalan aliran sungai dengan pendekatan ANN, ANFIS dan *Wavelet* [41]–[43], model penduga kekeringan dengan metode ANN dan ELM [44], [45], penguapan (*evaporation*) dengan metode *Relevance Virtual Machine* ELM, dan MARS [46], pengendapan (*precipitation*) dengan menggunakan metode ELM, ANFIS dan PSO-SVM [13], [47], [48], radiasi sinar matahari [49] dengan model markov, dan pemodelan temperatur udara dengan metode ANN [50], [51].

Model penduga curah hujan secara umum terbagi dua, yaitu model teoritis/konseptual dan model empiris/histori. Model empiris efektif untuk menemukan hubungan antara data input dan output ketika pengguna tidak memahami secara lengkap dari hubungan faktor-faktor fisik yang terjadi [3], cepat dalam pengembangan, dan kemudahan dalam implementasi waktu nyata [52]. Nilai akurasi prediksi curah hujan lebih diutamakan pada riset hidrologi. Di samping itu, pendekatan konseptual memerlukan proses komputasi yang mahal, kompleksitas tinggi dan padat peralatan *scientific* yang dalam beberapa tahun memerlukan pembaharuan peralatan [4], [52], [53].

Model penduga curah hujan berdasarkan data histori, antara lain dengan menggunakan ARIMA dengan berbagai variasinya, baik model autoregresi ARMA [19], ARIMA [33], maupun penggunaan variabel prediktor multivariat, yaitu VAR [20], ASTAR [33], SARIMAX [54], ARIMA dan NARX [16]. Metode penduga curah hujan berbasis ARIMA hanya sesuai untuk variabel dengan pola yang linier [55].

Pendekatan berbasis *machine learning* antara lain menggunakan kNN [9], SVM [11], PSO [48], ELM [51], [56], Fuzzy [57], GA [58]–[60], *Random Forest* [61] dan *Multiple Evaluation Criteria* [62]. Model ANN sebagai penduga curah hujan, baik yang standar, pengembangan maupun penggabungan dengan metode

lain (*hybrid*), cukup mendominasi. Model ANN standar, pengembangan pada tahap pra-proses dan cara representasi data, telah dilakukan untuk mendapatkan akurasi prediksi lebih baik [10], [50], [63]–[66]. Demikian juga model *hybrid* ANN [3], [10], [14] [15], [41], [43], [54], [67]–[72] menghasilkan akurasi yang cukup baik. Keunggulan lainnya dibandingkan ARIMA adalah bahwa ANN adaptif terhadap pola non-linier dan non-stasioner [3], [34], [53], [65], [66], [68].

Model peramalan curah hujan dengan studi kasus di Indonesia antara lain pada penelitian berikut:

- a. Pengembangan model peramalan curah hujan bulanan di wilayah Ngawi, Jawa Timur dengan metode *Adaptive Splines Threshold Autoregressive* (ASTAR), Fungsi Transfer dan model ARIMA dengan melibatkan prediktor data cuaca dan indeks global. Berdasarkan evaluasi nilai RMSE, hasil terbaik adalah model Fungsi Transfer diikuti dengan ASTAR [33].
- b. Metode *Ensemble* berbasis *Wavelet*, ANFIS, ARIMA untuk peramalan total hujan bulanan di wilayah Indramayu, diperoleh hasil bahwa model *Ensemble* memiliki kinerja lebih baik dibandingkan dengan model individual pembentuknya, yaitu dengan nilai korelasi antara data riil dan prediksi adalah 0,72 [2].
- c. Metode gabungan *Fuzzy* dan Algoritma Genetika untuk peramalan curah hujan harian dengan prediktor data cuaca dengan studi kasus wilayah Jakarta. Diperoleh hasil bahwa model gabungan memberikan akurasi cukup baik untuk klasifikasi intensitas hujan harian, yaitu > 90% [75]. Dengan studi kasus yang sama, apabila menggunakan metode *hybrid* GA dan *Partially Connected Feedforward Neural Network* (PCFNN), diperoleh nilai akurasi 81,52% [76].
- d. Metode VAR sebagai penduga curah hujan bulanan di sejumlah wilayah di Semarang (multilokasi) dengan prediktor tunggal yang menghasilkan nilai MAPE < 10%. Dibandingkan dengan ARIMA, akurasi VAR lebih baik [19], [20].
- e. Metode *Local Regression Smoothing* (LRS) dan *Fuzzy Grammatical Evolution* (GE) digunakan sebagai model penduga curah hujan bulanan di wilayah Bandung dengan nilai $R^2=84.62\%$. Apabila dibandingkan dengan ANFIS, $R^2=80\%$ dan GE mempunyai $R^2=74,35\%$ [77], [78].

- f. Metode *Exponential Smoothing Seasonal Planting Index* (ESSPI) untuk menduga curah hujan per bulan di 3 distrik wilayah Boyolali, dengan MAPE=23.73%-35.19% [79].
- g. Metode *Seemingly Unrelated Regression* (SUR) untuk model GSTARIMA sebagai penduga curah hujan per bulan di wilayah Malang, dengan $R^2=58\%$ [26].
- h. Metode Adaptif Jaringan Syaraf Tiruan untuk memprediksi curah hujan per bulan di daerah Bongan, Bali dengan prediktor ENSO, SOI, OLR dan SST mempunyai $R^2 = 74.6\%$ [80].

2.1.2 Penelitian terkait pemilihan variabel prediktor

Model penduga curah hujan dengan variabel prediktor lokal (data cuaca) telah digunakan pada sejumlah penelitian. Variabel dimaksud antara lain suhu minimum/maksimum/rata-rata, kelembaban udara, lama penyinaran, arah dan kecepatan angin, dengan atau tanpa memperhitungkan pengaruh tenggang waktu dari variabel.

Variabel input dalam sejumlah penelitian prediksi curah hujan diantaranya adalah curah hujan dan aliran permukaan pada periode sebelumnya dengan seleksi variabel menggunakan regresi *stepwise* [3], curah hujan dan air tanah data bulanan pada periode sebelumnya yang ditentukan melalui *Partial MI* [14], curah hujan harian periode sebelumnya [15], curah hujan dan limpasan periode harian [64], [67], suhu minimum dan maksimum, curah hujan, limpasan, salju yang turun dengan periode data bulanan [66], curah hujan dan limpasan dengan periode harian dan bulanan [54], dan suhu dengan periode bulanan [69].

Sedangkan penggunaan variabel prediktor global (indeks iklim) sebagai input model penduga curah hujan, antara lain adalah *El-Nino Southern Oscillation* (ENSO) dan *Indian Ocean Dipole* (IOD) yang memperhitungkan pengaruh tenggang waktu dan lokasi berdasarkan nilai korelasi [68], [81], *Southern Oscillation Index* (SOI), *Interdecadal Pacific Oscillation* (IPO), IOD, dan Niño 3.4 dengan hanya memperhitungkan pengaruh tenggang waktu dari curah hujan satu bulan sebelumnya [82].

Penelitian curah hujan di suatu wilayah di Jawa Timur, menggunakan indeks iklim bulanan, yaitu Nino3, Nino4, Nino3.4 dan DMI dengan melibatkan pengaruh tenggang waktu yang diidentifikasi dengan metode *prewhitening Cross-Correlation Function* (CCF) [33]. Penelitian serupa di wilayah Indramayu menggunakan *Sea-Surface Temperature* (SST) dan ENSO periode bulanan sebagai prediktor tanpa melibatkan pengaruh tenggang waktu [83].

Gabungan input lokal dan global sebagai prediktor dalam model penduga curah hujan telah digunakan dalam penelitian. Sebagai variabel input adalah suhu minimum, suhu maksimum, curah hujan periode bulan-bulan sebelumnya, SOI, IPO, DMI dan ENSO [63], [84].

Pelibatan indeks iklim dan pengaruh tenggang waktu (*time-lags*) sebagai input dalam pemodelan curah hujan telah terbukti meningkatkan akurasi prediksi dalam sejumlah penelitian, diantaranya penggunaan ENSO, SST dan SOI untuk curah hujan di wilayah Australia [82], [85], termasuk juga pengaruh iklim global terhadap curah hujan di wilayah Indonesia [21], [30], [86]–[88].

2.1.3 Prediktor tenggang waktu, fitur relevan dan tidak redundan

Humphrey dkk. [89] dalam studi bidang lingkungan menyatakan bahwa input penting dalam pemodelan yang harus dipertimbangkan adalah ruang dan waktu, hubungan kausalitas variabel observasi pada lokasi yang berbeda, dan pengaruh tenggang waktu dari variabel yang sama. Akibatnya, jumlah input yang potensial menjadi sangat banyak. Tantangannya bahwa seleksi variabel adalah memilih jumlah variabel input seminimal mungkin yang mewakili karakteristik hubungan input-output sekaligus meminimalkan redundansi variabel.

Model penduga curah hujan sebagai bagian dari studi lingkungan merupakan salah satu aplikasi dari peramalan data runtun waktu (*time-series*) sehingga keterlibatan dari variabel tenggang waktu akan mempengaruhi akurasi prediksi. Sejumlah cara untuk mengidentifikasi pengaruh dan hubungan tenggang waktu terhadap variabel respon pada pemodelan kasus hidrologi dan curah hujan telah banyak digunakan, antara lain teknik korelasi-silang [47] dan auto-korelasi [57], [58], [90]–[93]. Keduanya hanya dapat mengidentifikasi hubungan variabel tenggang waktu yang pengaruhnya bersifat linier dan stasioner.

Dalam penelitian lain, kandidat prediktor tenggang waktu untuk memprediksi curah hujan tidak ditentukan melalui metode tertentu, tetapi dengan menetapkan *lags t-1* sampai dengan *t-k* untuk mendapatkan kombinasi input terbaik [56], [63], [67], [69]. Cara lainnya adalah memilih fitur yang signifikan dengan metode tertentu, antara lain *Stepwise Regression* [3], [94] dan MI/PMI [14], [51], [57], [95]. Keunggulan dari MI adalah kemampuannya mendeteksi variabel input dengan pengaruh non-linier [6].

Metode lain dengan kemampuan lebih baik dibanding auto-korelasi dan korelasi silang yang mampu mengidentifikasi pengaruh tenggang waktu bersifat linier, non linier dan non stasioner pada studi kasus bidang hidrologi adalah DPCCA [96]–[98].

Untuk memperoleh efisiensi dalam model penduga data runtun waktu, tidak semua variabel prediktor yang telah diidentifikasi, akan digunakan sebagai input dalam model. Diperlukan proses pemilihan variabel input yang berguna secara laingsung (*relevant*), namun tidak berlebihan (*redundant*). Sejumlah metode untuk seleksi variabel input dari model penduga deret berkala telah diperkenalkan, diantaranya adalah berbasis metode korelasi [78]–[81] untuk memilih variabel input dengan pola keterkaitan yang linier dan *Symmetrical Uncertainty* (SU) untuk pola yang non-linier [103]–[105], MI [51, 106–108] untuk pola yang non-linier, *Principle Component Analysis* (PCA) [109], PSO [110], *Entropy* [9], ANFIS [13], maupun *Wavelet* dengan pendekatan model *hybrid* [52].

Untuk mengukur optimalitas himpunan fitur hasil seleksi, antara lain digunakan *MV* berbasis korelasi [111] untuk metode ensemble pada persoalan klasifikasi. *MV* juga digunakan untuk mengevaluasi hasil seleksi fitur dengan metode *Principle Component Analysis* dan *Correlation Attribute Selection* di mana hasil seleksi fitur dikatakan baik apabila nilai $MV > 0.5$ pada persoalan pemodelan kekeringan [112]. Sejumlah penelitian menggunakan cara heuristik untuk menentukan pemberhentian proses seleksi fitur [113], [114].

2.1.4 Penelitian spatio-temporal dan model *hybrid*

Sejumlah penelitian baru telah memperkenalkan pendekatan yang berbeda untuk model penduga data runtun waktu multivariat, yaitu menggunakan *spatio-*

temporal. Faghmous dan Kumar [37] menyarankan bahwa untuk meningkatkan akurasi, skalabilitas dan kemudahan interpretasi dari persoalan deret berkala multivariat, dapat digunakan *Spatio-Temporal Data Mining (STDM)*. Data curah hujan yang bersifat *zero-inflated* akan lebih fleksibel dalam analisisnya jika dimodelkan sebagai persoalan STDM [35]. Dengan menggunakan data curah hujan mingguan di wilayah Brasil, pendekatan STDM menghasilkan MSE lebih baik.

Penelitian spatio-temporal di bidang hidrologi antara lain dilakukan pada peramalan vegetasi untuk mengevaluasi dampak perubahan iklim. Dengan menggunakan multiplikatif ST-ARIMA [115], model ini menghasilkan kinerja lebih baik dibandingkan dengan ARIMA. Penelitian lain dari pemodelan spatio-temporal adalah penggunaan metode *non-Bayesian Hierarchical* untuk prediksi curah hujan di Murray-Darling Basin, Australia [38].

Studi kasus pemodelan curah hujan harian/bulanan dengan pendekatan spatio-temporal telah dilakukan di wilayah Jawa Timur, antara lain untuk menganalisis hujan ekstrim [31], pemodelan GSTAR untuk prediksi hujan bulanan [116] dan model GSTARIMA untuk prediksi curah hujan di wilayah kabupaten Malang [26]. Pendekatan spatio-temporal dapat juga digunakan pada model konseptual, yaitu model *Empirical Orthogonal Function* untuk analisis hujan harian di sejumlah wilayah di Jawa Timur [117].

Pengembangan model penduga untuk data runtun waktu, saat ini menuju kepada penggabungan dua atau lebih metode. Model *hybrid* untuk prediksi curah hujan diantaranya adalah *Autoregressive-ANN* [41], *wavelet-ANN* [46], *wavelet-ANFIS* [17], [59], *ARIMA-NARX* [16], *Soft Computing-MARS* [118], *wavelet-Gene Expression Programming* [58], *PSO-SVM* [48], *SOM-NARX* [14], *CAPSO-ANN* [119], *GA-LM-ANN* [3], dan *Multiple Regression-ANN* [68].

Secara umum, pengembangan model peramalan data runtun waktu berdasarkan penelitian Makridakis dan Hibon [120] antara lain disimpulkan bahwa:

- a. Metode yang canggih atau rumit secara statistik tidak menjamin hasil peramalan yang lebih akurat daripada metode yang sederhana. Namun hal ini tidak berarti bahwa beberapa metode yang canggih tidak menghasilkan kinerja yang baik karena tidak ada batas yang jelas atas klasifikasi metode disebut sederhana atau canggih.

- b. Keakuratan hasil atas kombinasi sejumlah metode lebih baik dibanding metode tunggal (individual) pada proses peamalan, baik satu periode maupun sejumlah periode ke depan.

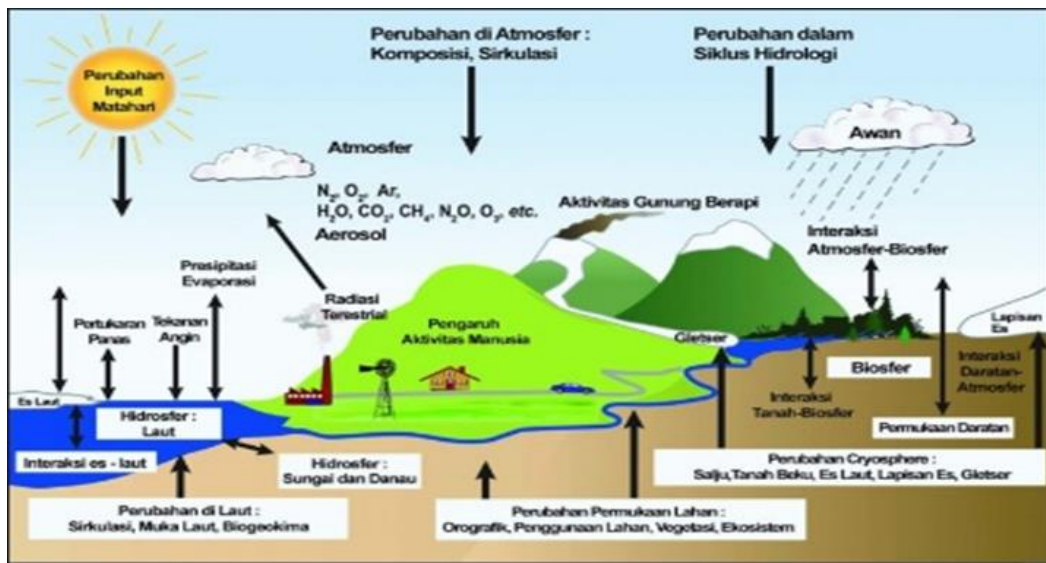
2.2 Dasar Teori

2.2.1 Cuaca dan iklim

Cuaca adalah keadaan/fenomena fisik dari atmosfer yang berhubungan dengan suhu, tekanan udara, angin, awan, kelembaban udara, radiasi, dan jarak pandang di suatu tempat dan pada waktu tertentu (per jam atau harian). Iklim adalah aspek dari cuaca di suatu tempat dan pada waktu tertentu dalam jangka waktu panjang. Misalnya, evaluasi dan prakiraan hujan bulanan, prakiraan musim hujan dan kemarau. Sedangkan musim adalah pembagian iklim dalam rentang waktu satu tahun [121].

Iklim terdiri atas atmosfer daratan, salju dan es, lautan dan badan-badan air lainnya, serta makhluk hidup. Iklim merupakan suatu sistem yang kompleks dan interaktif. Sistem berevolusi terhadap waktu oleh adanya pengaruh-pengaruh yang berasal dari dinamika internalnya dan juga sebagai akibat terjadinya perubahan pada faktor-faktor luar yang mempengaruhi iklim. Iklim umumnya didefinisikan sebagai cuaca rata-rata, dan dengan definisi yang demikian perubahan iklim dan cuaca saling terkait. Meskipun cuaca dan iklim terkait erat, terdapat perbedaan-perbedaan penting. Cuaca memiliki perilaku *chaotic* yang membuat tidak dapat diprediksi setelah beberapa hari ke depan. Proyeksi perubahan iklim (yakni cuaca rata-rata jangka panjang) akibat perubahan komposisi atmosfer atau akibat faktor-faktor lain adalah hal yang sangat berbeda dan sangat mungkin untuk dilakukan [122]. Sistem iklim dijelaskan pada Gambar 2.1.

Berikut dijelaskan definisi dan pengertian sejumlah variabel terkait dengan meteorologi dan iklim [123]. Curah hujan merupakan ketebalan air hujan yang terkumpul pada luasan 1 m². Hujan adalah titik-titik air di udara atau awan yang sudah terlalu berat karena kandungan airnya sudah sangat banyak, sehingga akan jatuh kembali ke permukaan bumi sebagai hujan (presipitasi). Satuan curah hujan yang digunakan adalah mm.



Gambar 2.1 Unsur-unsur cuaca dan iklim [124]

Suhu udara adalah ukuran energi kinetik rata-rata dari pergerakan molekul. Dalam sistem dua benda, benda yang kehilangan panas dikatakan benda yang bersuhu lebih tinggi. Lama penyinaran matahari merupakan salah satu dari unsur iklim, dan didefinisikan sebagai kekuatan matahari yang melebihi 120 w/m^2 .

Kelembaban udara adalah jumlah kandungan uap air yang ada dalam udara. Kandungan uap air di udara berubah-ubah bergantung pada suhu. Makin tinggi suhu, makin banyak kandungan uap airnya. Kelembaban relatif (nisbi), yaitu angka dalam persen yang menunjukkan perbandingan antara banyaknya uap air yang benar-benar dikandung udara pada suhu tertentu dan jumlah uap air maksimum yang dapat dikandung udara.

Secara umum wilayah Indonesia dikenal sebagai wilayah tropis yang memiliki 2 musim, yaitu kemarau dan hujan. Namun berdasarkan sistem perubahan angin musiman (monsun) berkaitan erat dengan sistem penghujan, musim dibagi menjadi 4 yaitu musim hujan (Desember-Februari), musim peralihan hujan-kering/transisi I (Maret-Mei), musim kering (Juni-Agustus), dan musim peralihan kering-hujan/transisi II (September-Nopember). Sedangkan untuk tipe iklim, seringkali dikelompokkan menjadi sejumlah tipe, yaitu sangat basah, lembab, agak kering, dan sangat kering [125].

Untuk keperluan prakiraan musim, BMKG membagi wilayah Indonesia menjadi menjadi 2 (dua) kelompok yaitu daerah ZoM dan daerah non zona musim

(non ZoM). ZoM didefinisikan sebagai daerah yang pola hujan rata-ratanya memiliki perbedaan yang jelas antara periode musim kemarau dan musim hujan. Daerah-daerah yang pola hujan rata-ratanya tidak memiliki perbedaan yang jelas antara periode musim kemarau dan musim hujan, disebut Non ZoM. Secara klimatologis, wilayah Indonesia terdapat 407 pola hujan dimana 342 diantaranya adalah ZoM. Wilayah Jawa Timur merupakan daerah yang memiliki pola monsun dan dibagi dalam 60 daerah ZoM, dimana tiap zona dibatasi oleh garis berwarna biru dan berlabel nomer zona, sesuai pada Gambar 2.2 [126].



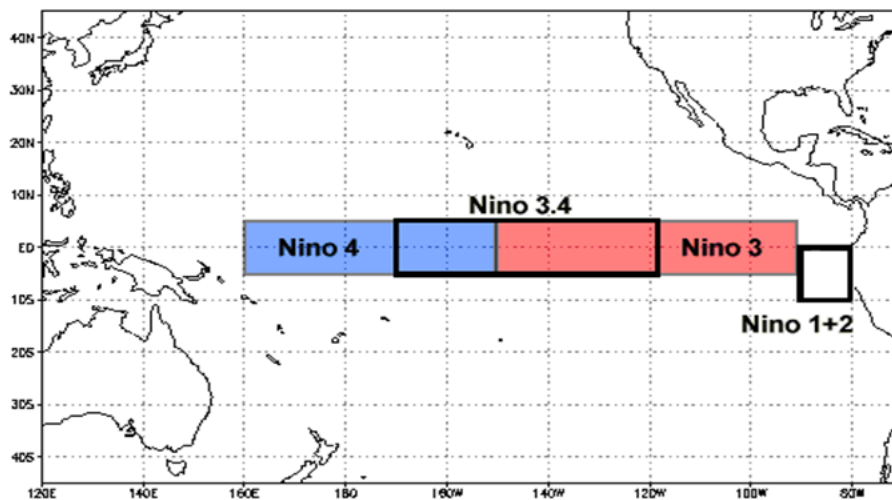
Gambar 2.2 Pembagian ZoM di wilayah Jawa Timur (BMKG)

Hujan di Indonesia dipengaruhi oleh iklim global, diantaranya ENSO (*El Nino-Southern Oscillation*) di mana besarnya pengaruh beragam. Pengaruh itu sangat besar pada daerah yang memiliki pola hujan monsoon. El Nino merupakan salah satu bentuk penyimpangan iklim di Samudra pasifik yang ditandai dengan kenaikan SST (*Sea Surface Temperature*) di daerah katulistiwa bagian tengah dan timur. Sebagai indikator untuk memantau kejadian El Nino dan La Nina, digunakan data pengukuran SST di zona lautan pasifik, sesuai Gambar 2.3 [28], [127].

Bentuk penyimpangan iklim yang menyerupai *El Nino dan La Nina* adalah IOD (*Indian Ocean Dipole*), merupakan anomali SST India yang berkontribusi pada kekeringan di Indonesia. IOD diukur dengan suatu indeks yang dinamakan

Dipole Mode Index (DMI), yaitu perbedaan SST di bagian barat samudra hindia ($50^{\circ} - 70^{\circ}$ BT, 10° LS – 10° LU) dan di bagian timur samudra hindia ($90^{\circ} - 110^{\circ}$, 10° LS – ekuator).

Anomali dari SST diukur berdasarkan indeks ENSO. Nino1_2 adalah wilayah pengamatan anomali SST pada koordinat $0^{\circ}-10^{\circ}$ LS dan $90^{\circ}-80^{\circ}$ BB, Nino3 pada koordinat 5° LU– 5° LS dan $150^{\circ}-90^{\circ}$ BB, Nino3.4 pada 5° LU– 5° LS, 160° BT– 150° BB, serta Nino4 pada 5° LU– 5° LS dan 170° BB– 120° BB.



Gambar 2.3 Zona ENSO di samudra pasifik [127]

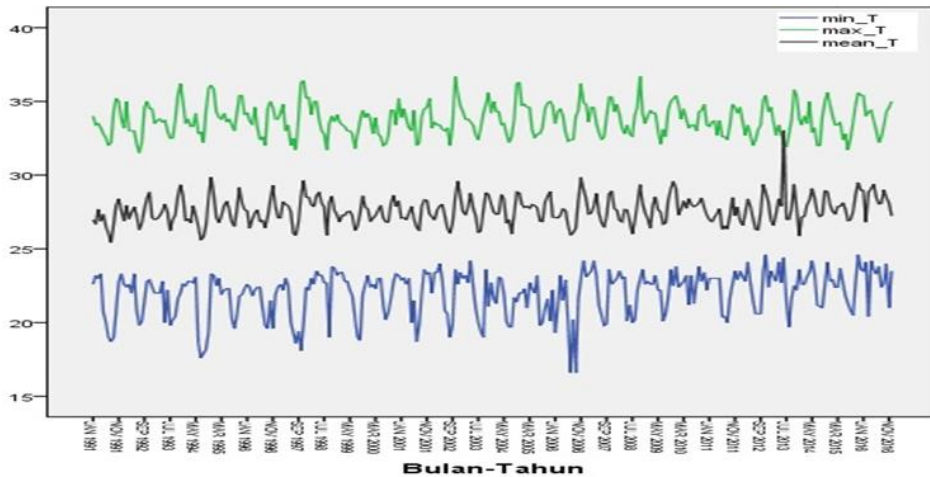
2.2.2 Data runtun waktu multivariat

Runtun waktu adalah representasi data yang terurut berdasarkan waktu kejadian dengan interval waktu yang sama di mana antar nilai saling berhubungan [128]. Data runtun waktu dapat dinotasikan sebagai $X_{i(t)}$; [$i = 1, 2, 3, \dots, m, t = 1, 2, 3, \dots, n$]. Jika nilai $m \geq 2$, disebut sebagai runtun waktu multivariat, sementara apabila $n = 1$ dinamakan *univariate time series* (UTS).

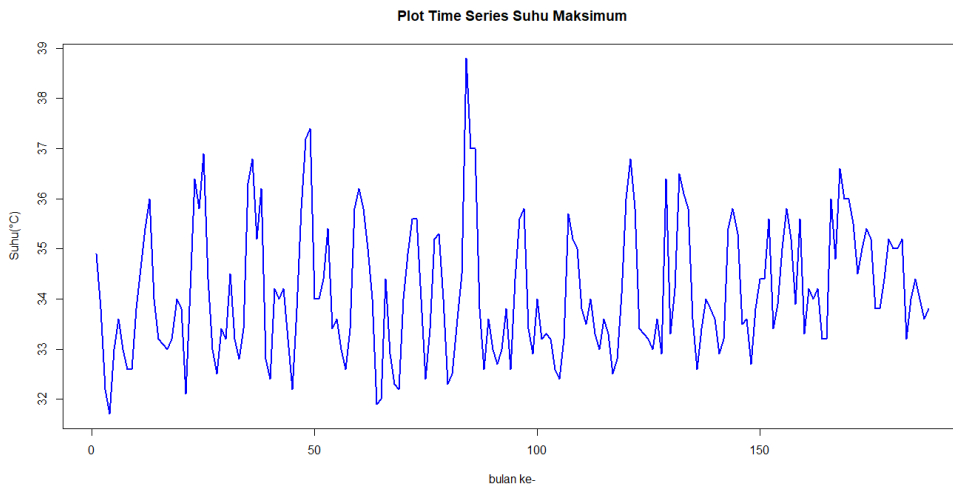
Tipe data runtun waktu dibedakan atas 2 jenis, yaitu stasioner dan non-stasioner. Yang dimaksud dengan UTS stasioner adalah bahwa variasi dan fluktuasi nilainya berada pada sekitar nilai rata-rata (relatif konstan). Data yang disajikan pada Gambar 2.4 yaitu suhu minimum($^{\circ}$ C), suhu rata-rata($^{\circ}$ C) dan suhu maksimum($^{\circ}$ C) adalah data runtun waktu multivariat yang termasuk pada tipe stasioner. Data bertipe stasioner dapat memiliki pola musiman jika terjadi perulangan pola yang relatif sama pada periode berikutnya. Contoh pada Gambar 2.5, yaitu data suhu

maksimum bulanan ($^{\circ}\text{C}$) merupakan data stasioner. Sedangkan data indeks iklim anomali SST pada Gambar 2.6 adalah bertipe non-stasioner.

Yang dimaksud dengan runtun waktu non-linier adalah data hasil proses stokastik yang tidak dapat dimodelkan dengan pendekatan linier. Sifat non-linier terjadi pada data runtun waktu stasioner maupun non-stasioner.



Gambar 2.4 Contoh data runtun waktu meteorologi [129]



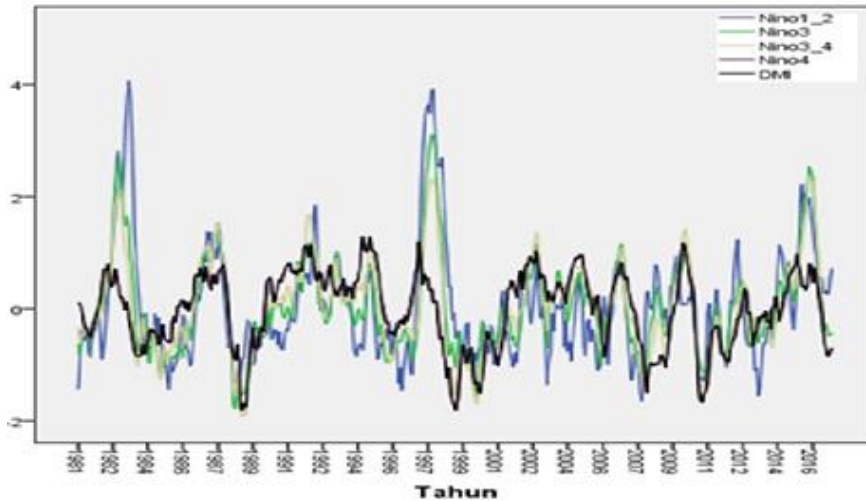
Gambar 2.5 Data suhu maksimum berpola stasioner [130]

Salah satu alat analisis untuk mengidentifikasi data time-series bersifat stasioner, musiman dan adanya pengaruh dari tenggang waktu (*time-lags*) adalah fungsi ACF (*Autocorrelation Function*) dan PACF (*Partial ACF*). Gambar 2.7 adalah *correlogram* dari ACF dan PACF data curah hujan bulanan.

Prosedur pengujian suatu data runtun waktu adalah bersifat stasioner, metode *augmented dickey-fuller test* (*adf-test*) dapat digunakan sebagai salah satu alat uji [20]. Persamaan (2.1) adalah prosedur untuk *adf-test*.

$$\Delta Y_t = \beta_0 + \beta_2 t + \gamma Y_{t-1} \sum_{i=1}^n \alpha_i \Delta Y_t + \varepsilon_t \quad (2.1)$$

$$\text{dimana } \varepsilon_t = \Delta Y_t = Y_{t-1} - Y_{t-2} \quad (2.2)$$



Gambar 2.6 Data runtun waktu non-stasioner [129]

Dengan hipotesis nol bahwa data runtun waktu adalah stasioner, nilai uji *adf* adalah $\hat{\gamma}/(\text{SE}(\hat{\gamma}))$. Semakin negatif nilai uji *adf*, maka data runtun waktu adalah stasioner. Sedangkan prosedur untuk menguji non-linieritas, digunakan metode *white neuralnet test* [131]. Apabila terdapat hubungan rekursif berikut:

$$t(n+1) = f(t(n)) + e \quad (2.3)$$

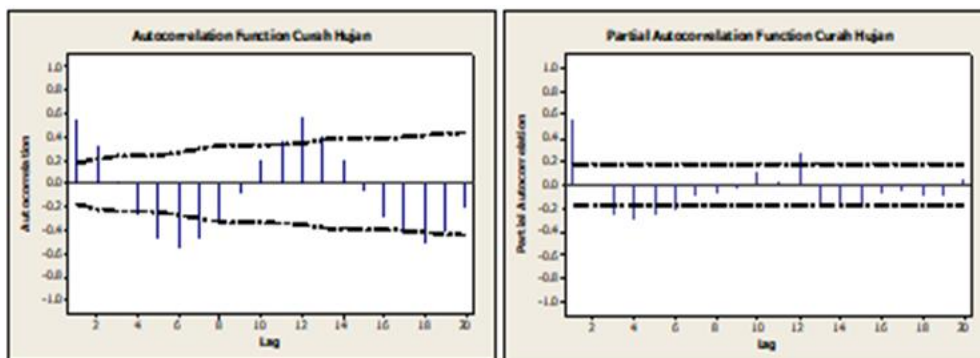
dimana e adalah kesalahan acak, $t(n)$ dan $t(n+1)$ adalah data runtun waktu pada saat n dan $(n+1)$, dan f adalah suatu fungsi, maka *white-test* akan menguji bahwa f adalah fungsi linier $f(x) = ax + b$. Hipotesis data runtun waktu adalah linier akan diterima jika nilai uji *white* kurang dari nilai kritis pada Tabel *Chi-Square*.

2.2.3 Ekstraksi fitur tenggang waktu (*time-lags*)

Untuk mengidentifikasi hubungan linier dua variabel berdasarkan tenggang waktu (*time-lags*) tertentu, alat analisis yang umum digunakan adalah *Cross Correlation Function* (CCF). Untuk menghitung CCF digunakan persamaan (2.4).

$$r_{xy}(T) = \frac{\sigma_{xy}(T)}{\sqrt{\sigma_{xx}(0)}\sqrt{\sigma_{yy}(0)}} \quad (2.4)$$

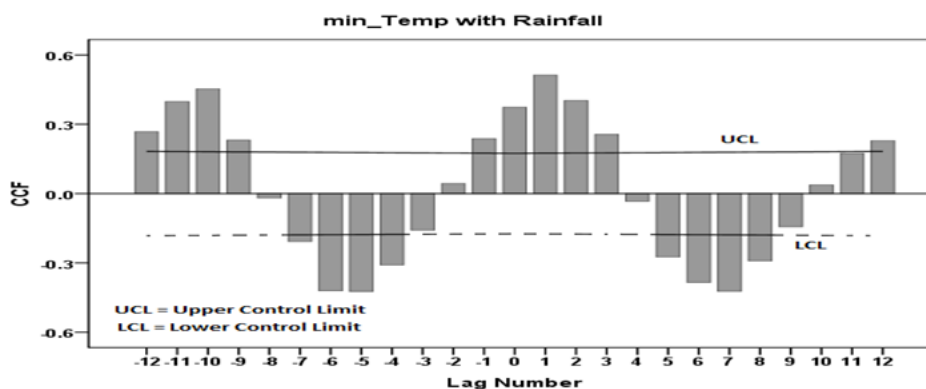
di mana $r_{xy}(T)$ adalah korelasi variabel x dan variabel y pada periode ke T dan σ_{xy} adalah kovariansi(x,y), σ_{xx} adalah variansi(x) dan σ_{yy} adalah variansi(y).



Gambar 2.7 ACF dan PACF dari curah hujan di Surabaya [129]

Dalam hubungan antara dua runtun waktu (y_t dan x_t), data y_t mungkin berhubungan dengan *lags* yang lalu dari data x . Sampel CCF membantu mengidentifikasi *lags* dari variabel x yang mungkin merupakan kandidat prediktor dari y_t . Sampel CCF didefinisikan sebagai himpunan korelasi sampel antara x_{t+h} dan y_t untuk $h = 0, \pm 1, \pm 2, \pm 3$, dan seterusnya.

Nilai negatif dari h menunjukkan bahwa korelasi antara variabel x pada waktu sebelum t dan variabel y pada saat t . Hasil nilai korelasi dapat diuji dengan membandingkan nilai korelasi silang dengan nilai estimasi *standard error* dari korelasi, yaitu $\pm 2/\sqrt{n}$. Pada Gambar 2.8, ditunjukkan *correlogram* CCF curah hujan dengan suhu minimum. Apabila nilai dari korelasi lebih dari batas UCL dan LCL, maka korelasinya signifikan.



Gambar 2.8 *Correlogram* CCF antara curah hujan dan suhu minimum [133]

Karena CCF memiliki keterbatasan hanya untuk identifikasi pengaruh tenggang waktu yang bersifat linier dan stasioner, cenderung redundan dan bersifat *greedy* [132], maka metode lain dikembangkan, yaitu DCCA/DPCCA. Keunggulan metode ini adalah kemampuan identifikasi hubungan diantara dua variabel yang bersifat linier/non-linier dan non-stasioner sehingga dapat digunakan untuk memperoleh kandidat variabel prediktor lebih lengkap.

Langkah-langkah DPCCA adalah sebagai berikut [96], [97]:

1. Jika terdapat m variabel runtun waktu, $i = 1, 2, \dots, n$; $\{x_i^1\}, \{x_i^2\}, \dots, \{x_i^m\}$
2. Masing masing *runtun waktu* dianggap sebagai *random-walk*:

$$P_k^j \equiv \sum_{i=1}^k x_i^j \text{ untuk } j = 1, 2, \dots, m; k = 1, 2, \dots, n \quad (2.5)$$

3. Buat profil P sebanyak $(n-s)$. Setiap kotak berisi nilai $(s+1)$ dari i s/d $(s+i)$.
4. Hitung *trend* lokal dengan *polynomial fit*.
5. Hilangkan *trend* dari *random walk*

$$Y_{(i-1)(s+1)+k-i+1}^j = P_k^i - \widetilde{P}_{k,l}^j; \quad \widetilde{P}_{k,l}^j (i \leq k \leq i+s) \quad (2.6)$$

Diperoleh nilai residual dari runtun waktu

$$Y_l^j, l = 1, 2, \dots, (n-s)(s+1) \text{ untuk setiap series } \{x_i^j\}$$

6. Hitung Kovariansi semua pasangan residual

$$F_{j_1, j_2}^2(s) \equiv \frac{\sum_{l=1}^{(N-s)(s+1)} Y_l^{j_1} Y_l^{j_2}}{(N-s)(s-1)} \quad (2.7)$$

Diperoleh nilai kovariansi

$$F^2(s) = \begin{pmatrix} F_{1,1}^2(s) & F_{1,2}^2(s) & \dots & F_{1,m}^2(s) \\ F_{2,1}^2(s) & F_{2,2}^2(s) & \dots & F_{2,m}^2(s) \\ \vdots & \vdots & \ddots & \vdots \\ F_{m,1}^2(s) & F_{m,2}^2(s) & \dots & F_{m,m}^2(s) \end{pmatrix} \quad (2.8)$$

7. Hitung nilai korelasi silang untuk pasangan data runtun waktu:

$$\rho_{j_1, j_2}(s) \equiv \frac{F_{j_1, j_2}^2(s)}{F_{j_1, j_1}^2(s) F_{j_2, j_2}^2(s)} \quad (2.9)$$

$$\rho(s) = \begin{pmatrix} \rho_{1,1}(s) & \rho_{1,2}(s) & \dots & \rho_{1,m}(s) \\ \rho_{2,1}(s) & \rho_{2,2}(s) & \dots & \rho_{2,m}(s) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{m,1}(s) & \rho_{m,2}(s) & \dots & \rho_{m,m}(s) \end{pmatrix} \quad (2.10)$$

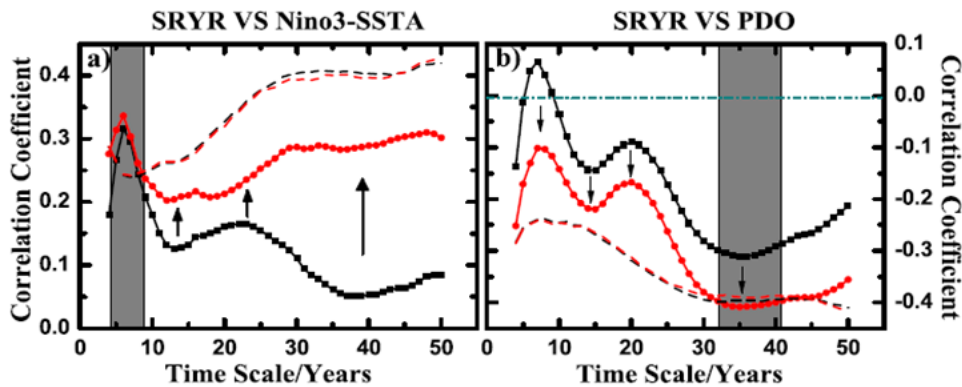
8. Hitung matriks invers dari korelasi silang

$$C(s) = \rho^{-1}(s) = \begin{pmatrix} C_{1,1}(s) & C_{1,2}(s) & \dots & C_{1,m}(s) \\ C_{2,1}(s) & C_{2,2}(s) & \dots & C_{2,m}(s) \\ \vdots & \vdots & \ddots & \vdots \\ C_{m,1}(s) & C_{m,2}(s) & \dots & C_{m,m}(s) \end{pmatrix} \quad (2.11)$$

9. Hitung DPCCA

$$\rho_{DPCCA}(j_1, j_2, s) \equiv \frac{-C_{j_1, j_2}(s)}{\sqrt{C_{j_1, j_1}(s) C_{j_2, j_2}(s)}} \quad (2.12)$$

Gambar 2.9 adalah contoh hasil DPCCA (garis berwarna merah) bila dibandingkan dengan DCCA (garis berwarna hitam) untuk hubungan antara curah hujan dengan Nino3 dan curah hujan dengan *Pasific Decadal Oscilation* (PDO).



Gambar 2.9 DPCCA antara curah hujan, Nino3 dan PDO [96]

2.2.4 Seleksi fitur

Dari studi Yu dkk. [103], fitur prediktor untuk pemodelan MTS dibagi menjadi 3 subset yang saling lepas, yaitu relevan yang sangat kuat, relevan yang lemah dan fitur yang tidak relevan. Subset pertama mutlak/harus ada sebagai prediktor dari variabel respon dan tidak dapat dihapus karena akan berpengaruh

kepada hasil prediksi. Subset kedua kurang diperlukan, kecuali pada kondisi tertentu. Subset ketiga tidak diperlukan sebagai prediktor pada kondisi apapun. Pembagian fitur prediktor atas 3 subset dijelaskan pada Gambar 2.10.

Untuk melakukan seleksi prediktor yang relevan, asosiasi antar fitur diasumsikan linier. Suatu prediktor disebut relevan terhadap variabel respon jika korelasinya signifikan. Formula untuk menentukan prediktor relevan adalah dengan menggunakan korelasi *Pearson*, yang ditulis pada persamaan (2.13).

$$\rho_{xy} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_i (y_i - \bar{y})^2}} \quad (2.13)$$

ρ_{xy} adalah korelasi antara variabel x dan variabel y dan nilainya signifikan jika lebih dari nilai kritis tabel korelasi dengan derajat bebas dan level signifikansi tertentu.

Sedangkan untuk mengevaluasi asosiasi (hubungan) antar fitur yang bersifat non-linier, digunakan *symmetrical uncertainty* (SU). Formula SU berbasis pada *entropy* dan *information gain* [99]. Jika X adalah suatu variabel, maka *entropy* didefinisikan sesuai persamaan (2.14).

$$H(X) = \sum_i P(x_i) \log(P(x_i)) \quad (2.14)$$

dan *entropy* X setelah ada nilai observasi dari variabel lain, didefinisikan sebagai persamaan (2.15).

$$H(X|Y) = - \sum_j P(y_j) \sum_i P(x_i|y_j) \log(P(x_i|y_j)) \quad (2.15)$$

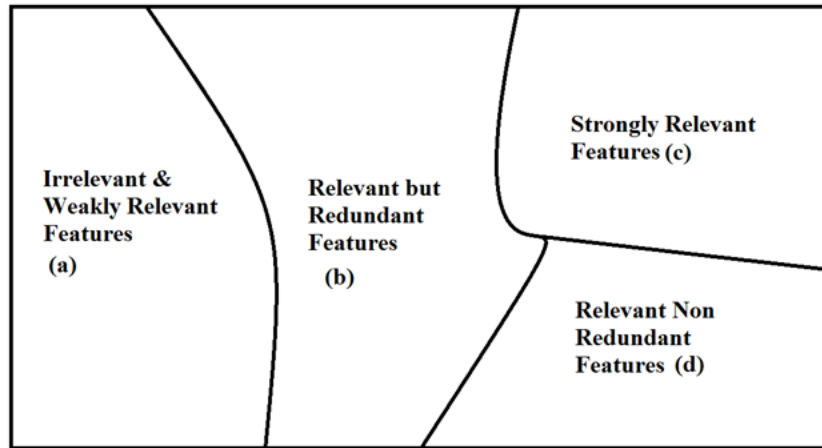
di mana $P(x_i)$ adalah probabilitas *prior* untuk semua nilai dari X , dan $P(x_i|y_j)$ adalah probabilitas *posterior* dari X ketika diberikan nilai dari Y . Jumlahan *entropy* X mengikuti persamaan (2.16).

$$IG(X|Y) = H(X) - H(X|Y) \quad (2.16)$$

Namun, IG bersifat bias sehingga perlu dilakukan normalisasi sesuai persamaan (2.17) sebagai *Symmetrical Uncertainty* (SU).

$$SU(X, Y) = 2[IG(X|Y)/(H(X) + H(Y))] \quad (2.17)$$

$SU(X, Y)$ antara variabel X dan Y disebut signifikan jika nilainya lebih dari nilai kritis Tabel *Chi-Square* dengan derajat bebas *significant level* α [134].



Gambar 2.10 Fitur prediktor yang relevan dan tidak relevan [103]

2.2.5 ARIMA dan GSTAR

Salah satu model penduga data runtun waktu yang populer dan paling sering digunakan adalah model ARIMA. Model AR menunjukkan bahwa terdapat relasi antara nilai sekarang (Z_t) dan nilai pada periode sebelumnya (Z_{t-k}) dengan tambahan nilai random e_t [135]. Model MA menunjukkan bahwa ada relasi antara nilai sekarang (Z_t) dengan residual pada periode lalu (a_{t-k} , dengan $k=1, 2, \dots, q$). Model ARIMA(p, d, q) adalah campuran dari AR(p) dan MA(q) dengan pola non-stasioner dan d adalah level *differencing*. Bentuk ARIMA (p, d, q) adalah sesuai dengan persamaan (2.18).

$$\phi_p(1 - B)^d Z_t = \theta_q(B) a_t \quad (2.18)$$

di mana p adalah order dari model AR dan q adalah order dari model MA dengan level *difference* d .

$$\phi_p(B) = (1 - \phi_1 B - \phi_2 B^2 \dots - \phi_p B^p) \quad (2.19)$$

$$\theta_q(B) = (1 - \theta_1 B - \theta_2 B^2 \dots - \theta_q B^q) \quad (2.20)$$

Bentuk umum dari model ARIMA dengan pola data musiman ditulis dengan ARIMA(p, d, q) (P, D, Q)^s mengikuti persamaan (2.21).

$$\phi_p(B) \Phi_P(B^s) (1 - B)^d (1 - B^s)^D Z_t = \theta_q(B) \Theta_Q(B^s) a_t \quad (2.21)$$

di mana s adalah periode musiman. Persamaannya adalah sebagai berikut:

$$\Phi_P(B^s) = (1 - \Phi_1 B^s - \phi_2 B^{2s} \dots - \Phi_p B^{Ps}) \quad (2.22)$$

$$\text{dan } \Theta_Q(B) = (1 - \theta_1 B^s - \theta_2 B^{2s} \dots - \theta_Q B^{Qs}). \quad (2.23)$$

ARIMAX adalah pengembangan dari ARIMA dengan ditambahkan variabel eksternal (*exogenous Variable*) sebagai penduga terhadap variabel respon [33], [136]. Salah satu bentuk ARIMAX fungsi transfer satu input adalah persamaan (2.24).

$$Z_t = C + \frac{\omega_s(B)}{\delta_r(B)} B^b X_t + N_t \quad (2.24)$$

di mana:

$$N_t = \frac{\theta(B)}{\phi(B)} a_t \quad (2.25)$$

$$\omega_s(B) = \omega_0 - \omega_1 B - \omega_2 B^2 \dots - \omega_s B^s \quad (2.26)$$

$$\delta_r(B) = \delta_0 - \delta_1 B - \delta_2 B^2 \dots - \delta_r B^r \quad (2.27)$$

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 \dots - \phi_p B^p \quad (2.28)$$

$$\theta_q(B) = 1 - \theta_1 B - \theta_2 B^2 \dots - \theta_q B^q \quad (2.29)$$

Apabila data runtun waktu memperhitungkan pengaruh antar lokasi, maka disebut sebagai data spatio-temporal. Data Y_{ts} , pada saat $t, t = 1, 2, \dots, T$, di lokasi $s, s = 1, 2, \dots, S$ dapat diformulasikan sebagai [137]:

$$Y_{ts} = \theta_t + m_s + \varepsilon_{ts} \quad (2.30)$$

Model GSTAR adalah generalisasi dari model STAR yang memungkinkan parameter autoregresif bervariasi per lokasi, sehingga model GSTAR berlaku untuk karakteristik heterogen dari lokasi sampel. GSTAR(λ_k, p) dituliskan sebagai persamaan 2.31 [138].

$$Z_{(Nx1)}(t) = \sum_{k=1}^p \sum_{i=0}^{\alpha \lambda_k} \Phi_{ki(NxN)} Z_{(Nx1)} W_{NxN}^i(t-k) + e_{(Nx1)}(t) \quad (2.31)$$

Di mana:

$Z(t)$: vektor observasi pada saat t

λ_k : order *spatial* dari komponen *Autoregressive* ke- k

Φ_{ki} : matrik diagonal dengan elemen diagonal sebagai AR dan *space-time*.

e_t : residual dengan vektor rata-rata 0 dan matrik variansi-kovariansi $\sigma^2 I$.

GSTAR (1,1) dapat ditulis sebagai model orde satu dari vektor autoregresif. Dari GSTAR (1,1) yang didefinisikan, diasumsikan memiliki model linier untuk GSTAR dengan persamaan (2.32 - 2.33).

$$Z_t = \varphi Z_{t-1} + e_t, \varphi = \varphi_{10} + \varphi_{11} W \quad (2.32)$$

$$Y_i = X_i \beta + e_i \quad (2.33)$$

$$Y_i = \begin{bmatrix} Z_i(2) \\ Z_i(3) \\ \vdots \\ Z_i(T) \end{bmatrix} X_i = \begin{bmatrix} Z_i(1) & V_i(1) \\ Z_i(2) & V_i(2) \\ \vdots & \vdots \\ Z_i(T-1) & V_i(T-1) \end{bmatrix} e_i = \begin{bmatrix} e_i(2) \\ e_i(3) \\ \vdots \\ e_i(T) \end{bmatrix}, \beta_i = \begin{bmatrix} \varphi_{10}^{(1)} \\ \vdots \\ \varphi_{10}^{(N)} \\ \varphi_{11}^{(1)} \\ \vdots \\ \varphi_{11}^{(1)} \end{bmatrix} \quad (2.34)$$

$$V_i(t) = \sum_{j=1}^N w_{ij} z_j, \forall i \neq j \quad (2.35)$$

Oleh karenanya, model untuk semua lokasi secara bersamaan memiliki struktur model linier. Untuk setiap lokasi $i = 1, 2, \dots, S$, akan memiliki model linier terpisah (persamaan 2.33), yang berarti bahwa untuk setiap lokasi, estimator kuadrat terkecil untuk β_i dapat dihitung secara terpisah. Namun, nilai estimator tergantung pada nilai $Z_i(t)$ di lokasi lain, karena bentuk $V_i(t)$. Selanjutnya, estimator dari model GSTAR (1,1) ditulis pada persamaan (2.36).

$$\widehat{\varphi} = (X'X)^{-1}X'Y, \forall i \neq j \quad (2.36)$$

Terdapat 3 metode untuk menetapkan bobot w_{ij} , yaitu :

1. Bobot *Uniform*, yaitu nilai bobot adalah sama untuk semua lokasi. Cara ini seringkali digunakan ketika data observasi diperoleh dari sejumlah lokasi dengan jarak antar lokasi yang relatif sama. [139]. Bobot *Uniform* didefinisikan sebagai :

$$w_{ij} = \frac{1}{n_i}, \text{ di mana } n_i \text{ adalah jumlah lokasi yang dekat dengan lokasi ke-}i$$

2. Bobot inversi jarak, adalah bobot untuk GSTAR berdasarkan jarak aktual antar lokasi dalam satu wilayah dan dinormalisasi [139]. Normalisasi harus memenuhi kondisi $\sum_{i \neq j} w_{ij}^{(1)}$. Misalnya, jarak antar tiga lokasi sebagai berikut:

r_1 : jarak lokasi 1 dan 2

r_2 : jarak lokasi 1 dan 3

r_3 : jarak lokasi 2 dan 3

$$w_{ij} = \begin{pmatrix} 0 & w_{12} & w_{13} \\ w_{21} & 0 & w_{23} \\ w_{31} & w_{32} & 0 \end{pmatrix} = \begin{pmatrix} 0 & \frac{1/r_1}{1/r_1 + 1/r_2} & \frac{1/r_2}{1/r_1 + 1/r_2} \\ \frac{1/r_3}{1/r_1 + 1/r_3} & 0 & \frac{1/r_1}{1/r_1 + 1/r_3} \\ \frac{1/r_2}{1/r_2 + 1/r_3} & \frac{1/r_3}{1/r_2 + 1/r_3} & 0 \end{pmatrix} \quad (2.37)$$

3. *Normalized Cross-Correlation* [140].

Adalah bobot korelasi silang yang dinormalisasi, dinyatakan dalam persamaan (2.38).

$$w_{ij}(k) = \frac{r_{ij}(k)}{\sum_{k=1} |r_{ik}(k)|}, i \neq j \quad (2.38)$$

Di mana nilai bobot harus memenuhi $\sum_{j \neq 1} |w_{ij}| = 1$ dan nilai dari korelasi silang sesuai dengan persamaan (2.39). $r_{ij}(k)$ adalah korelasi antara variabel x dan variabel y pada periode k dan σ_{ij} adalah kovariansi(i,j), $\sigma_{ii}(0)$ adalah variansi(i) and $\sigma_{jj}(0)$ adalah variansi(j) [141].

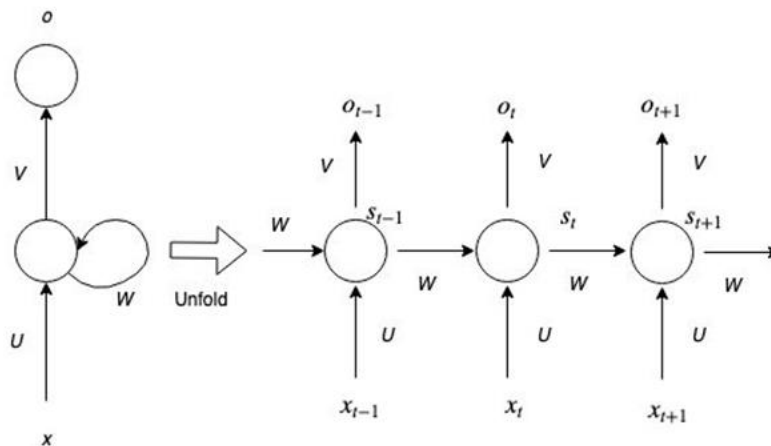
$$r_{ij}(k) = \frac{\sigma_{ij}(k)}{\sqrt{\sigma_{ii}(0)}\sqrt{\sigma_{jj}(0)}} \quad (2.39)$$

2.2.6 Model Recurrent Neural Networks (RNN)

Apabila model peramalan dalam kelompok ARIMA mempunyai keterbatasan hanya mampu memprediksi data runtun waktu dengan pola dan fluktuasi linier, sebaliknya ANN memiliki kemampuan untuk menggambarkan berbagai pola non linier dari data. Satu kelebihan yang signifikan dari model ANN sebagai pendekatan pola non linier dibandingkan metode yang lain adalah bahwa ANN merupakan pendekatan umum untuk fungsi kelas yang besar dengan akurasi yang tinggi [142].

Recurrent neural network (RNN) yang juga disebut jaringan umpan balik adalah jenis jaringan pada ANN dimana terdapat *loop* sebagai koneksi umpan balik dalam jaringan [143], [144]. RNN adalah jaringan yang mengakomodasi output jaringan untuk menjadi input pada jaringan tersebut yang kemudian digunakan untuk menghasilkan output yang baru. Hal ini dijelaskan pada Gambar 2.11.

RNN mempunyai kemampuan dinamis karena perilaku jaringan tidak hanya bergantung pada input saat ini saja melainkan pada operasi sebelum jaringan. Hal ini akan mengurangi jumlah total parameter yang diperlukan untuk proses belajar dibandingkan *deep neural network* tradisional yang menggunakan himpunan bobot dan bias yang berbeda untuk setiap *layer*



Gambar 2.11 Recurrent Neural Networks [144]

Terdapat berbagai tipe RNN, diantaranya adalah Jaringan *Elman*, Jaringan *Hopfield*, *Long Short-Term Memory (LSTM)* dan *Gated Recurrent Unit (GRU)*. RNN dirancang untuk pengenalan pola diantaranya data sekuensial yang berguna untuk prediksi dan peramalan [143]. RNN dapat diterapkan pada teks, *images*, suara maupun data runtun waktu dimana input dan output dinyatakan sebagai vektor.

2.2.7 Model *hybrid* GSTAR dan RNN

Antara kedua model ARIMA dan ANN telah berhasil dalam domain yang berbeda, linier atau non linier. Namun tidak ada dari keduanya merupakan model yang sesuai untuk semua kondisi. Pendekatan ARIMA untuk memodelkan permasalahan non linier yang kompleks tidak cukup baik. Sebaliknya, pemodelan ANN untuk permasalahan linier menghasilkan kinerja yang beragam, bergantung pada karakteristik data, antara lain adanya multikolinieritas, pencilan, jumlah sampel dan tingkat *noise* [142].

Karenanya tidak disarankan mengaplikasikan ANN pada beragam tipe data. Hal ini karena dalam persoalan nyata, akan sulit mengenali secara lengkap karakteristik data sehingga metodologi *hybrid* yang menggabungkan kemampuan pemodelan linier dan non-linier dapat menjadi pilihan untuk digunakan dalam praktek. Melalui kombinasi model yang berbeda, aspek yang berbeda dari pola data dapat dikenali.

Narendra [135] dan Zhang [142] dalam penelitiannya menyatakan bahwa dalam kasus riil peramalan data runtun waktu, sangat jarang ditemukan pola data yang murni linier atau non linier. Pada umumnya adalah gabungan di antara kedua

pola tersebut. Oleh karenanya, model ARIMA atau ANN tidak cukup mampu memodelkan atau meramalkan data runtun waktu yang demikian. Hal ini karena ARIMA memiliki kekurangan tidak bisa mengenali pola non linier,

Kedua penelitian model *hybrid* tersebut diterapkan pada data UTS untuk berbagai data sintetik dan data riil. Banihabib [16] dan Jain [41] telah menerapkan model *hybrid* ARIMA-ANN pada bidang meteorologi dan terbukti meningkatkan nilai akurasi dari penggunaan model parsial. Oleh karenanya dalam penelitian ini dikembangkan model *Hybrid* GSTAR dan ANN. Hal ini sejalan dengan kesimpulan dari penelitian Makridakis dan Hibbon [120] yang menyatakan bahwa kombinasi sejumlah metode lebih baik dibanding metode tunggal dalam model peramalan data runtun waktu.

Berdasarkan hal tersebut, cukup beralasan untuk membagi data runtun waktu atas komponen linier L_t dan struktur komponen non-linier N_t , dengan persamaan (2.40).

$$y_t = L_t + N_t \quad (2.40)$$

Kedua komponen diestimasi dari data di mana langkah pertama adalah pemodelan GSTAR untuk estimasi komponen linier. Selanjutnya, nilai kesalahan dari model tersebut dihitung dengan persamaan (2.41), di mana e_t hanya mengandung komponen non-linier. \hat{L}_t adalah hasil estimasi berdasarkan model GSTAR.

$$e_t = y_t - \hat{L}_t \quad (2.41)$$

Nilai kesalahan e_t cukup penting untuk dianalisis apakah pendekatan linier telah sesuai untuk data runtun waktu. Apabila nilai e_t masih memiliki struktur pola tertentu, maka model GSTAR belum sesuai. Selanjutnya e_t dimodelkan dengan RNN menggunakan persamaan (2.42). Hasil akhir prediksi merupakan gabungan dari estimasi linier dan non linier, sesuai persamaan (2.43).

$$e_t = f(e_{t-1}, e_{t-2}, \dots, e_{t-n}) + \varepsilon_t \quad (2.42)$$

$$\hat{y}_t = \hat{L}_t + \hat{N}_t \quad (2.43)$$

2.2.8 Ukuran kinerja model

Kinerja model penduga data runtun waktu Z_t , dapat menggunakan ukuran *Root Mean Square Error* (RMSE), *Mean Absolute Error* (MAE), Koefisien Determinansi (R^2) dan korelasi (r) [145].

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (Z_t - \hat{Z}_t)^2}{n}} \quad (2.44)$$

$$R_{sq} = 1 - \frac{\sum_{t=1}^n (Z_t - \hat{Z}_t)^2}{\sum_{t=1}^n (Z_t - \bar{Z}_t)^2} \quad (2.45)$$

$$r = \sqrt{R_{sq}} \quad (2.46)$$

Untuk menetapkan nilai optimal dari *trade-off* antara RMSE dan jumlah prediktor p pada proses training (*in-sample*), digunakan *corrected AIC* (*Akaike Information Criterion*) dengan persamaan (2.47), AIC diperoleh dari persamaan (2.48) [6], [132].

$$AIC_{corrected} = AIC + \frac{2(p+1)(p+2)}{n-p} \quad (2.47)$$

$$AIC = n \cdot \log \frac{\sum_{j=1}^n (y_j - \hat{y}_j(p))^2}{n} + 2(p+1) \quad (2.48)$$

Halaman ini sengaja dikosongkan

BAB 3 METODE PENELITIAN

Sejumlah langkah dilakukan untuk membangun model penduga curah hujan ZoM dengan studi kasus wilayah Jawa Timur. Pada Gambar 3.1, dijelaskan diagram alir untuk pemodelan sebagai berikut:

a. Pengumpulan data.

Data curah hujan, suhu min/mean/maks, kelembaban udara, lama penyinaran matahari, rata-rata kecepatan angin, dan kecepatan angin maksimum diperoleh dari BMKG (<http://dataonline.bmkg.go.id/>). Sedangkan untuk data indeks iklim harian/bulanan, yaitu suhu permukaan laut yang meliputi Nino1.2, Nino2, Nino3, Nino3.4, dan Nino4 yang diperoleh dari *The Royal Netherlands Meteorological Institute* (KNMI) secara online pada <http://climexp.knmi.nl/>. Periode data yang digunakan pada penelitian ini adalah Januari 1983–Desember 2016. Sebagai studi kasus adalah data cuaca dari 3 stasiun BMKG di wilayah Surabaya (Juanda, Perak I, Perak II) dan 3 stasiun BMKG di wilayah Malang-Pasuruan (Karang Ploso, Karangtretes, Tretes), baik untuk data periode bulanan maupun harian. Pada Tabel 3.1 disajikan ketersediaan data dan sumber data. Yang dimaksud ketersediaan data 0% adalah bahwa pada satu bulan penuh pada tahun tertentu, tidak tersedia data pengamatan variabel cuaca.

Tabel 3.1 Sumber data cuaca dan indeks iklim

Stasiun Cuaca	Data	Sumber	Ketersediaan
Meteorologi Juanda	Curah Hujan	http://dataonline.bmkg.go.id/	58% - 100%
	Cuaca	http://dataonline.bmkg.go.id/	0% - 100%
Meteorologi Perak I	Curah Hujan	http://dataonline.bmkg.go.id/	81% - 100%
	Cuaca	http://dataonline.bmkg.go.id/	51% - 100%
Meteorologi Perak II	Curah Hujan	http://dataonline.bmkg.go.id/	70% - 100%
	Cuaca	http://dataonline.bmkg.go.id/	0% - 100%
Geofisika Karangates	Curah Hujan	http://dataonline.bmkg.go.id/	0% - 100%
	Cuaca	http://dataonline.bmkg.go.id/	0% - 100%
Klimatologi Karangploso	Curah Hujan	http://dataonline.bmkg.go.id/	0% - 100%
	Cuaca	http://dataonline.bmkg.go.id/	0% - 100%
Geofisika Tretes	Curah Hujan	http://dataonline.bmkg.go.id/	0% - 100%
	Cuaca	http://dataonline.bmkg.go.id/	0% - 100%
KNMI	Indeks Iklim	http://climexp.knmi.nl/	100%

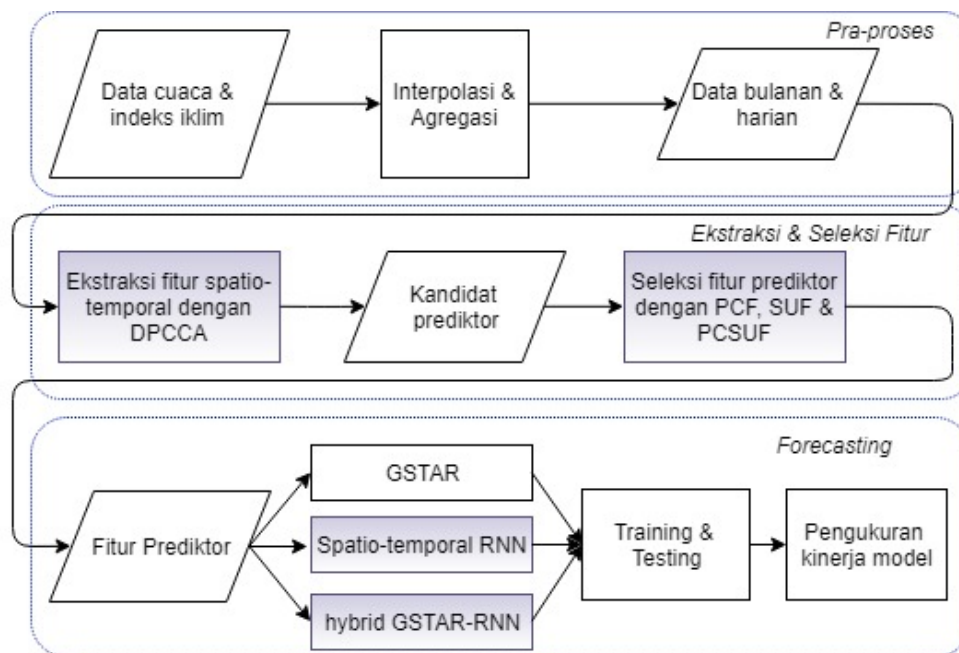
b. Tahap interpolasi dan agregasi.

Dilakukan interpolasi dari data yang tidak tersedia (*missing value*), khususnya data observasi cuaca, di mana kelengkapan data curah hujan dan cuaca adalah

0% - 100%. Untuk kondisi demikian, diisi dengan nilai rata-rata dari variabel dimaksud atau diasumsikan nilainya sama dengan tahun sebelumnya. Untuk pemodelan data runtun waktu bulanan, pengamatan cuaca harian dilakukan agregasi menjadi data bulanan.

c. Tahap ekstraksi dan seleksi fitur (pra-proses).

Pemodelan dan prediksi curah hujan merupakan sistem yang kompleks dimana melibatkan interaksi berbagai faktor, yaitu *time-lags*, lokasi, fitur linier/non-linier dan pengaruh iklim global juga diperhitungkan dalam pemodelan. Pada tahap ini, dilakukan ekstraksi fitur pengaruh *time-lags* dan lokasi menggunakan DPCCA dan seleksi prediktor yang relevan sebagai penduga curah hujan dengan menggunakan PCSUF. Selanjutnya, himpunan prediktor terpilih dievaluasi optimalitasnya.



Gambar 3.1 Metodologi pengembangan model penduga curah hujan

d. Membangun model GSTAR, RNN dan hybrid GSTAR-RNN.

Pada tahap ini, dilakukan pemodelan curah hujan dengan GSTAR, ST-RNN dan hybrid GSTAR-RNN dengan berbagai variasi himpunan prediktor hasil ekstraksi dan seleksi fitur. Pada GSTAR, hubungan antar lokasi stasiun cuaca dan waktu diperhitungkan dalam model dan mengasumsikan bahwa hubungan antar variabel bersifat linier. Sedangkan untuk pola hubungan yang bersifat

non-linier, digunakan metode *RNN* dengan input variabel prediktor yang telah diidentifikasi pada tahap (c) dan residual. Sedangkan model hybrid GSTAR-RNN adalah peramalan berbasis spatio-temporal dengan dua tahap. Tahap pertama adalah pemodelan dengan GSTAR, sedangkan residual dari GSTAR dimodelkan dengan RNN. Prediksi/peramalan merupakan nilai gabungan dari hasil GSTAR dan RNN.

- e. Prediksi (data training) dan peramalan (data testing).

Model *hybrid* diterapkan pada data training dan data testing untuk mengetahui kinerja dari model dalam melakukan peramalan curah hujan. Proporsi antara data training (kalibrasi) dan data testing (validasi) ditentukan berdasarkan persentase. Peramalan dilakukan untuk satu periode ke depan (*one-step ahead*).

- f. Tahap evaluasi

Evaluasi kinerja model diukur melalui nilai RMSE, *R-square* dan Korelasi (*r*) antara nilai aktual dan nilai prediksi. Model memiliki kinerja tinggi apabila memiliki nilai RMSE dan MAE yang rendah serta *R-square* dan *r* yang tinggi. Analisis terhadap nilai residual juga dilakukan untuk menjamin bahwa telah memenuhi *i.i.d.n* ($0, \sigma^2$).

Uji coba model dilakukan melalui beberapa pilihan, yaitu mengevaluasi kinerja model untuk GSTAR, model ANN dan model *hybrid*. Skenario lainnya adalah uji coba tanpa menggunakan ekstraksi fitur maupun seleksi fitur. Jumlah prediktor yang optimal di mana model menghasilkan kinerja terbaik diukur dengan *corrected AIC* pada proses training.

3.1 Tahap Interpolasi dan Agregasi

Tahap interpolasi digunakan untuk melengkapi *missing value* dari data meteorologi BMKG (data cuaca). *Missing-value* diakibatkan kerusakan/terganggunya peralatan sehingga tidak teramati. Ada berbagai teknik interpolasi yang dapat digunakan untuk melengkapi *missing-value*, diantaranya adalah menggunakan nilai rata-rata. Sedangkan agregasi digunakan untuk mengubah data harian menjadi data bulanan. Berikut adalah agregasi untuk sejumlah data pengamatan lokal harian.

- a. Curah hujan bulanan: jumlah curah hujan harian pada satu bulan.

- b. Suhu udara minimum bulanan: nilai minimum dari data suhu minimum dalam satu bulan.
- c. Suhu udara maksimum bulanan: nilai maksimum dari data suhu maksimum dalam satu bulan.
- d. Suhu udara rata-rata bulanan: nilai rata-rata dari data suhu rata-rata dalam satu bulan.
- e. Kelembaban udara bulanan: nilai rata-rata dari data kelembaban udara dalam satu bulan.
- f. Kecepatan angin bulanan: nilai rata-rata dari data kecepatan angin harian dalam satu bulan.
- g. Lama penyinaran bulanan: nilai rata-rata dari data lama penyinaran harian dalam satu bulan.
- h. Kecepatan angin maksimum bulanan: nilai maksimum dari data kecepatan angin maksimum harian dalam satu bulan.

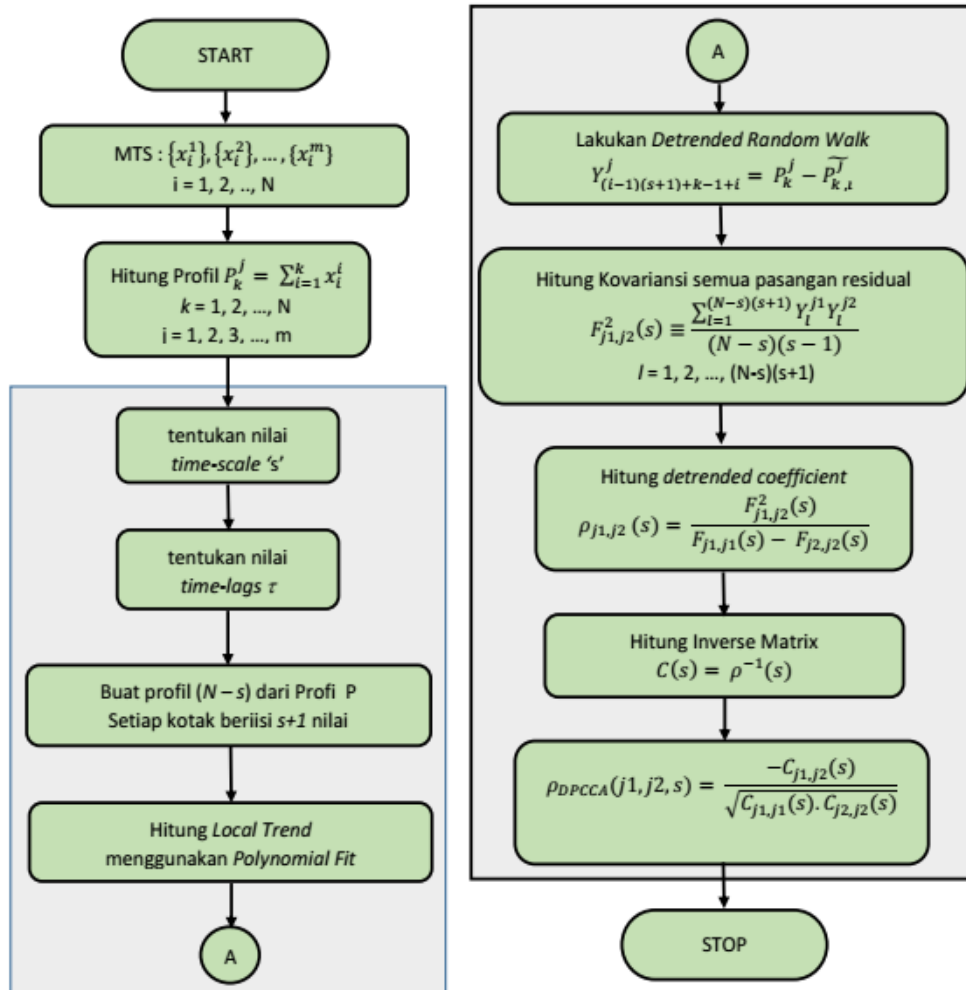
3.2 Tahap Ekstraksi dan Seleksi fitur

3.2.1 Ekstraksi fitur tenggang waktu (*time-lags*)

Untuk mendapatkan himpunan kandidat fitur *time-lags* yang terkait dengan curah hujan dengan mempertimbangkan pengaruh lokasi stasiun cuaca (*spatial*), digunakan metode DPCCA [96] sesuai yang dijelaskan pada sub-bab 2.2.3. Apabila terdapat 11 variabel *prediktor* dan 1 variabel respon pada setiap lokasi, maka untuk 3 lokasi dan 12 *time-lags* akan terdapat nilai koefisien DPCCA sebanyak $3 \times 13 \times 12$. Selanjutnya dilakukan uji signifikansi terhadap koefisien DPCCA dengan membandingkannya terhadap nilai *standard error*, yaitu $\pm 2/\sqrt{n}$ untuk menentukan kandidat prediktor yang akan diseleksi dengan metode PCSUF.

Proses untuk memperoleh nilai DPCCA dijelaskan pada Gambar 3.2. Data MTS dibentuk sebagai *random-walk* P_k^j dan kemudian dibuat *profil* P sebanyak $(N - s)$. Pola *trend* dihilangkan dengan model *Polynomial* sehingga tersisa runtun waktu residual. Nilai residual setiap MTS inilah yang dihitung nilai kovariansi dan

korelasi silang sehingga didapatkan matriks DPCCA. DPCCA yang optimal bergantung pada pemilihan skala waktu s dan *time-lags* k .



Gambar 3.2 Diagram alir untuk ekstraksi fitur *time-lags* dengan DPCCA [96]

3.2.2 Tahap seleksi fitur relevan dan penghapusan redundan dengan PCSUF

Zhang [52] dalam penelitiannya menyimpulkan bahwa tahap pra-proses sebelum pemodelan data runtun waktu, khususnya di bidang hidrologi telah terbukti meningkatkan kinerja prediksi. Pra-proses dimaksud diantaranya adalah berbasis pada transformasi, cara representasi data dan pra-proses berbasis ekstraksi/seleksi fitur [146].

Gambar 3.3 (bagian a) menunjukkan urutan proses seleksi prediktor dari ekstraksi fitur menggunakan *spatio-temporal* DPCCA. Kandidat prediktor yang

relevan secara linier dipilih menggunakan korelasi pearson (PC) antara prediktor (X_i) dengan respon (Y), sedangkan prediktor non linier, digunakan *Symmetrical Uncertainty* (SU).

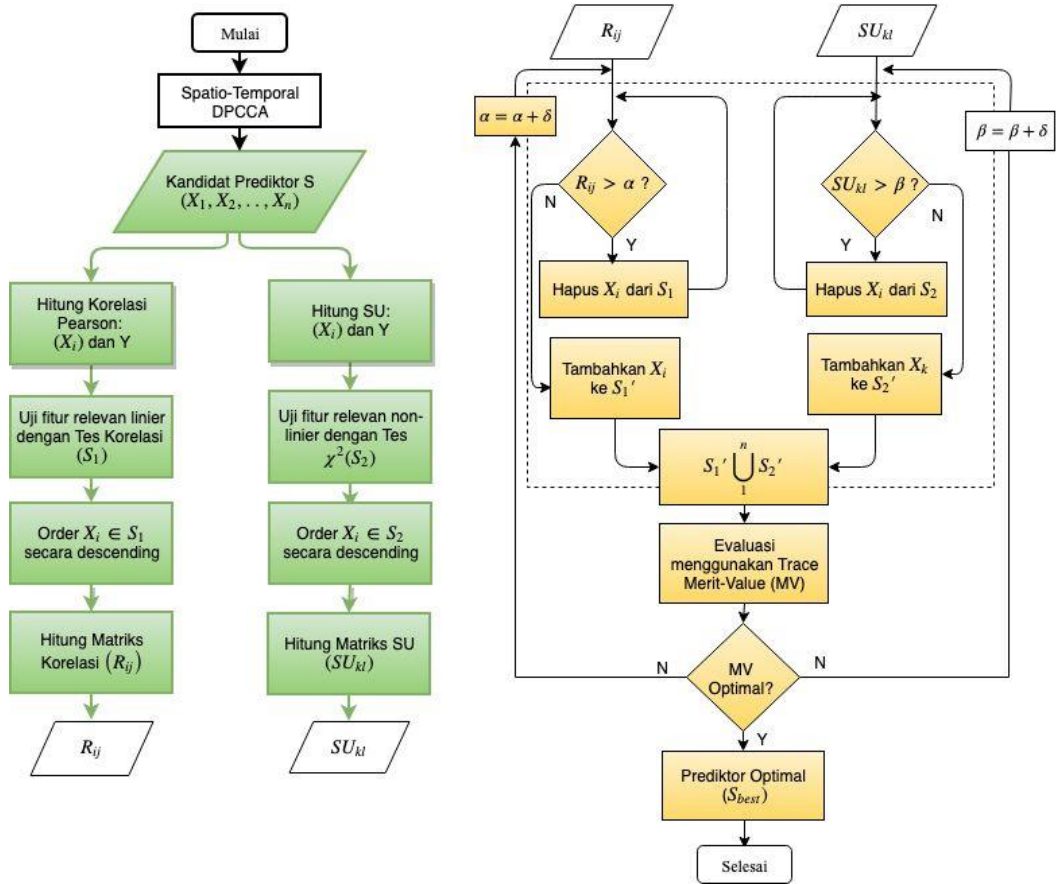
Fitur relevan linier yang signifikan disimpan pada himpunan fitur S_1 , sedangkan untuk fitur relevan non-linier disimpan pada S_2 . Fitur pada S_1 dan S_2 diurutkan secara *descending* berdasarkan nilai korelasi atau nilai SU dan dihitung nilai korelasi antar fitur pada himpunan S_1 (disimpan pada matriks R_{ij}) dan S_2 (disimpan pada matriks SU_{kl}).

Pada Gambar 3.3 bagian b, subset prediktor S_1 dievaluasi kembali dengan menggunakan perbandingan nilai korelasi antar fitur r_{ij} dengan *threshold* α . Apabila $r_{ij} > \alpha$ berarti bahwa antara fitur i dan j adalah terjadi redundansi sebagai prediktor linier. Hal yang sama berlaku untuk evaluasi redundansi prediktor non-linier dengan membandingkan nilai pada SU_{kl} terhadap *threshold* β .

Himpunan prediktor optimal diperoleh melalui iterasi dari kombinasi nilai α dan β . Gabungan 2 metode ini disebut PCSUF yang dikembangkan berdasarkan penelitian dari Jiang [99] dan Lei Yu [103]. Fungsi evaluasi (π) dari subset prediktor optimal menggunakan persamaan (3.1) yang merupakan adaptasi dari fungsi *merit value* [147]. Fungsi ini menyatakan bahwa nilai relevansi antara variabel respon Y dengan prediktor X_i sebanyak k dibagi relevansi antar variabel prediktor.

$$\pi_{X_i,j,Y} = \frac{k(\overline{SU})_{Y,X_i}}{\sqrt{k + k(k-1)(\overline{SU})_{X_i,X_j}}} \quad (3.1)$$

Oleh karena ekstraksi dan seleksi fitur untuk peramalan curah hujan menggunakan pendekatan *spatio-temporal*, maka nilai $\pi_{X_i,j,Y}$ diubah menjadi matriks MV dengan orde $s \times s$ di mana s menyatakan banyaknya lokasi. Sedangkan untuk mendapatkan kombinasi optimal nilai α dan β yang menghasilkan himpunan prediktor terbaik, didasarkan pada *trace* dari MV .



(a) Seleksi fitur (b) Menghapus Redundansi
 Gambar 3.3 Seleksi fitur relevan dan hapus fitur redundansi

Detil dari proses seleksi fitur dan penghapusan fitur redundansi menggunakan PCSUF dijelaskan pada *pseudocode* berikut.

```

Input:
S(F1, F2, ..., Fn, R) //data set with n features, one response variable in every location
α, β // a predefined threshold
Output: Sbest
Begin
  For i = 1 to n do begin
    Calculate Corri,R for Fi;
    If (Corri,R ≥ value of critical table PC)
      append Fi to S1list;
  End
  For i = 1 to n do begin
    Calculate SUi,R for Fi;
    if(SUi,R ≥ value of critical table Chi-Square)
      append Fi to S2list;
  Order S1list and S2list in descending
  Fj = getFirstElement at (S1list)
  do begin
    Fi = getNextElement(S1list, Fj)
    if (Fi <> Null)
      do begin

```

```

    If ( $Corr_{i,R} \geq \alpha$ )
      remove  $F_i$  from  $S1'_{list}$ 
       $F_i = getNextElement(S1'_{list}, F_i)$ 
    End until ( $F_i = Null$ )
     $F_j = getNextElement(S1'_{list}, F_j)$ 
  End until ( $F_j = Null$ )
   $F_j = getFirstElement$  at ( $S2'_{list}$ )
do begin
   $F_i = getNextElement(S1'_{list}, F_j)$ 
  if ( $F_i \neq Null$ )
    do begin
      If ( $SU_{i,R} \geq \beta$ )
        remove  $F_i$  from  $S2'_{list}$ 
         $F_i = getNextElement(S2'_{list}, F_i)$ 
      End until ( $F_i = Null$ )
       $F_j = getNextElement(S2'_{list}, F_j)$ 
    End until ( $F_j = Null$ )
   $S_{best} = S1'_{list} \cap S2'_{list}$ 

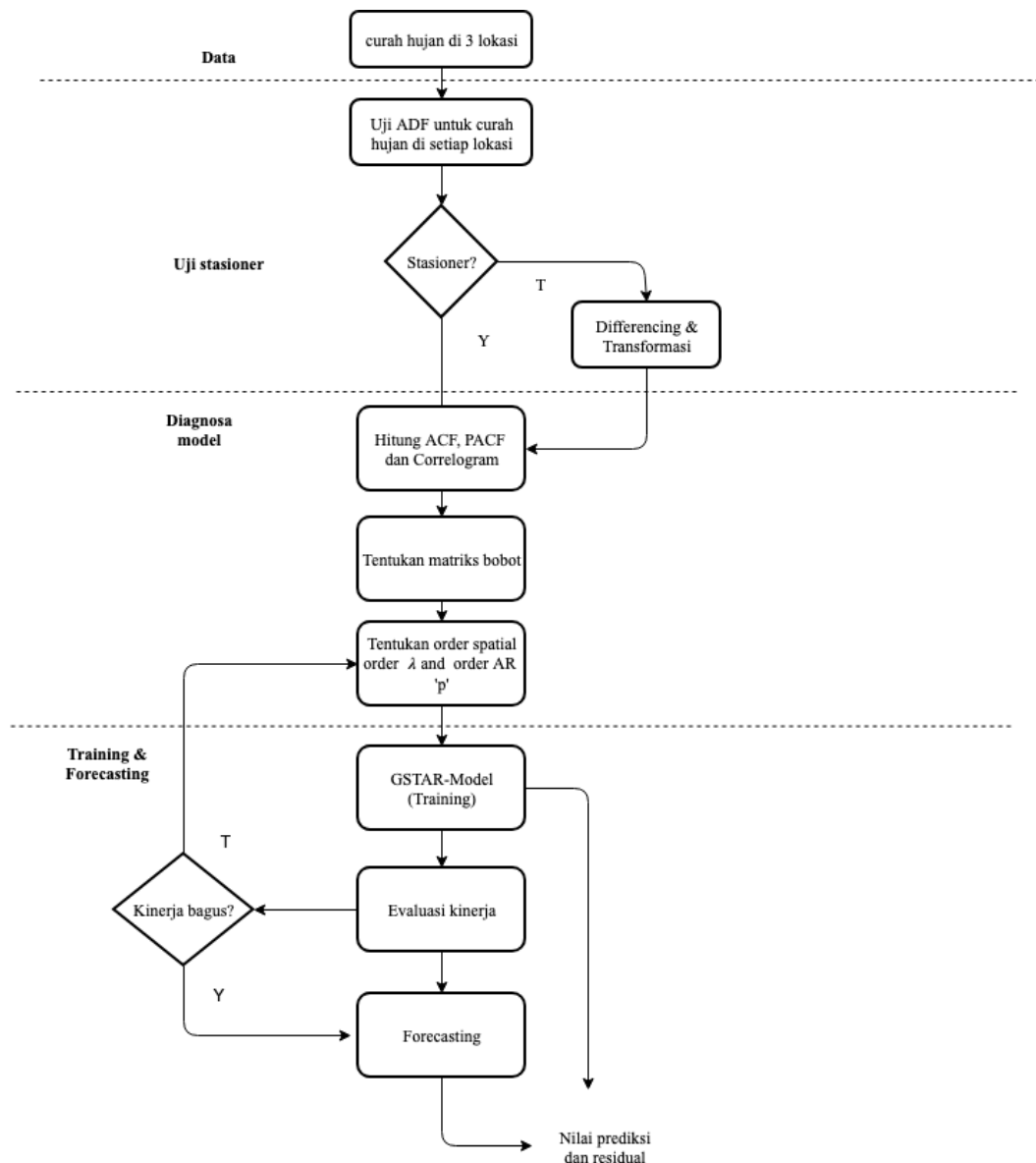
```

3.3 Pemodelan GSTAR

Tahap untuk pemodelan curah hujan pada S lokasi adalah sebagai berikut:

1. Menguji sifat stasioner dari data curah hujan di setiap lokasi dengan menggunakan *Augmented Dickey-Fuller (ADF) test*. Apabila tidak stasioner, maka perlu dilakukan teknik *differencing* atau transformasi.
2. Melakukan identifikasi awal dari model untuk menentukan order dari waktu yaitu p dan order dari spasial (λ_p). Dengan menggunakan *Correlogram* dari ACF dan PACF, maka dapat diidentifikasi nilai p dan λ_p .
3. Tentukan nilai dari matriks bobot dengan menggunakan metode *Uniform*, *Inversi Jarak* atau *Normalization Cross-Correlation*. Pemilihan metode bobot didasarkan atas hasil kinerja terbaik.
4. Lakukan pemodelan GSTAR dengan estimasi parameter order *Autoregressive* dan order *spatial*.
5. Evaluasi kinerja model untuk mendapatkan model terbaik.
6. Lakukan proses testing (*forecasting*).
7. Lakukan analisis residual (*error*) dari GSTAR apakah telah memenuhi syarat identik dan saling bebas (*independent*).

Pada Gambar 3.4 dijelaskan proses pemodelan GSTAR untuk data curah hujan di 3 lokasi.



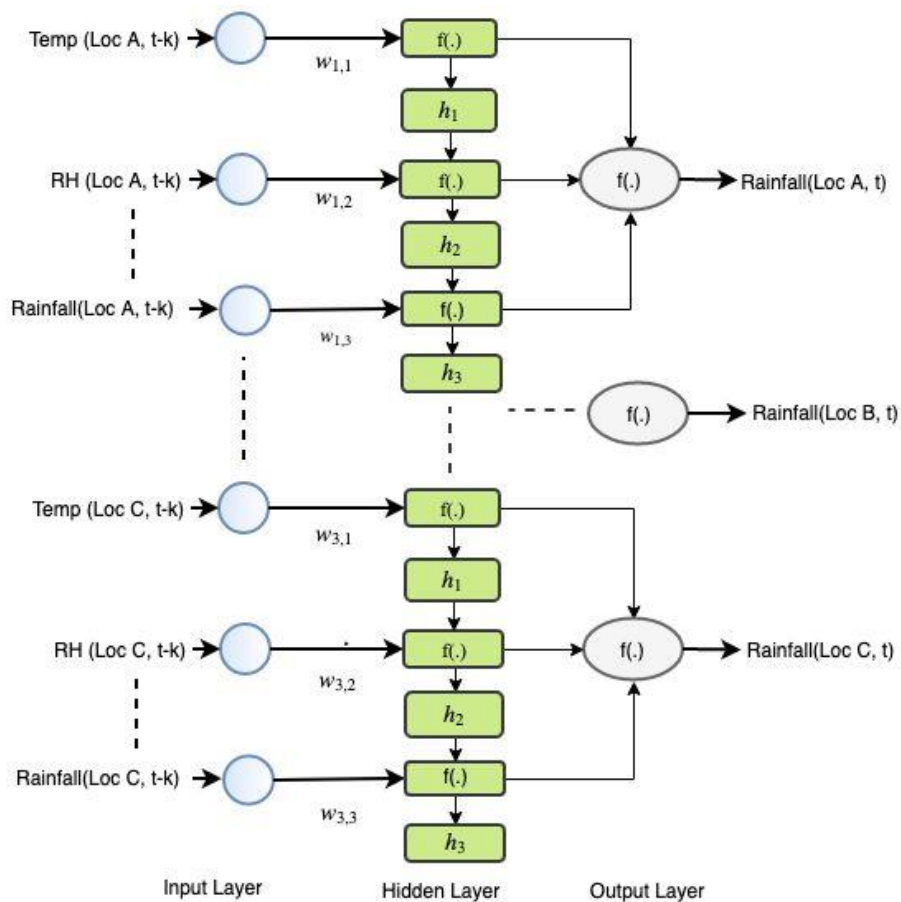
Gambar 3.4 Diagram alir model GSTAR

3.4 Pemodelan spatio-temporal RNN

Proses membangun model peramalan curah hujan dengan menggunakan *spatio-temporal* RNN (ST-RNN) adalah sebagai berikut:

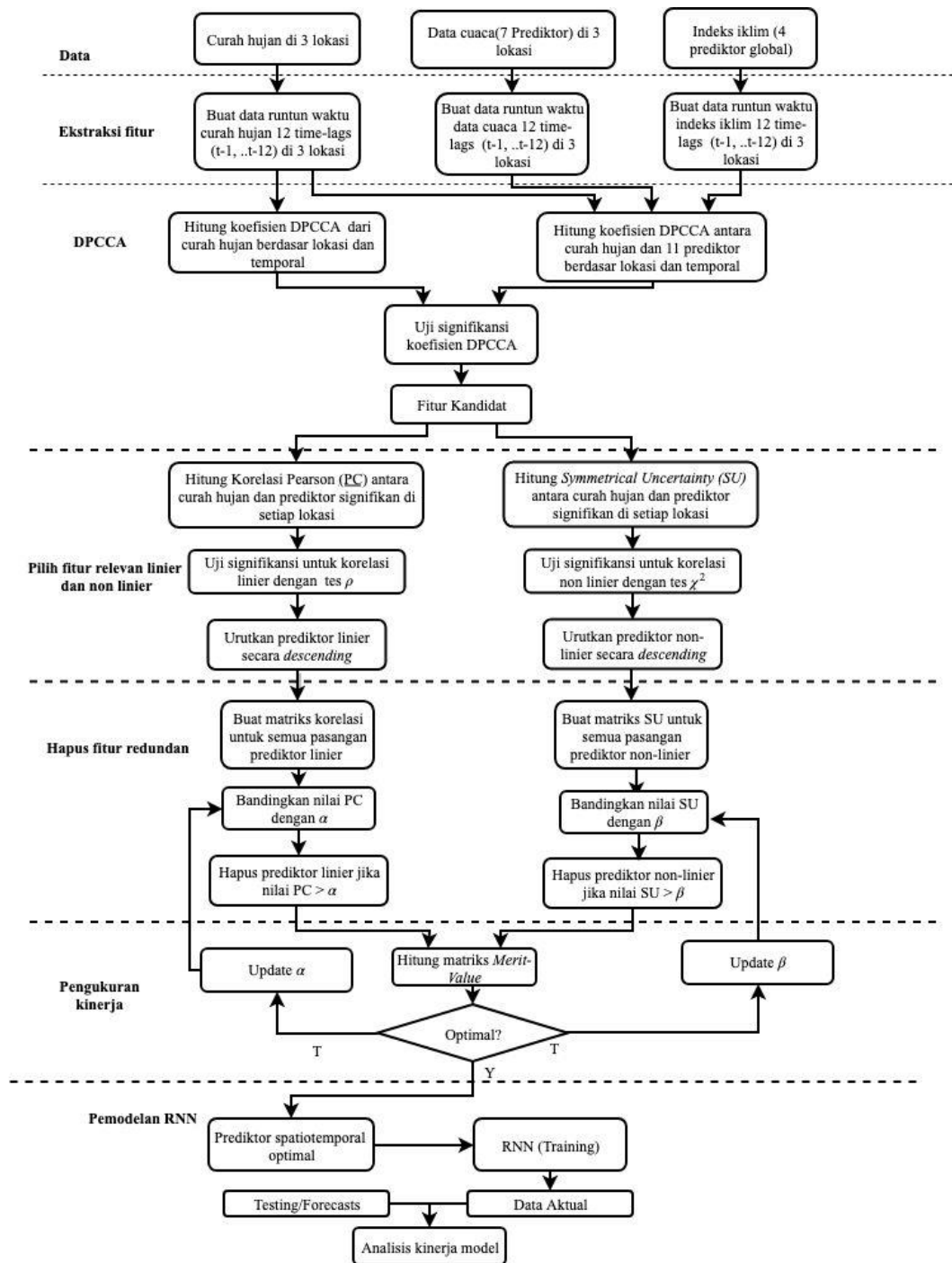
1. Himpunan fitur prediktor yang optimal dari hasil DPCCA-PCSUF menjadi input pada jaringan RNN. $X_{s, m, t-k}$ sebagai notasi dari prediktor lokasi ke- s , variabel ke- m pada *time-lags* $t-k$. $s = 1, 2, \dots, S, m = 1, 2, \dots, M, k = 1, 2, \dots, 12$.
2. Data curah hujan pada S lokasi menjadi nilai target dari jaringan RNN(Y_s, t).
3. Menentukan data training dan testing dengan menggunakan persentase partisi.

4. Melakukan normalisasi dari data prediktor dan curah hujan dengan menggunakan formula : $(\text{data} - \min(\text{data}) / (\max(\text{data}) - \min(\text{data})))$.
5. Menetapkan arsitektur dari jaringan RNN (Gambar 3.5).
6. Melakukan proses training dengan menetapkan terlebih dahulu nilai dari *learning-rate*, *momentum*, *batch* dan jumlah iterasi.
7. Mentransformasi nilai curah hujan hasil proses training dalam skala awal.
8. Mengukur kinerja hasil proses training, meliputi RMSE, AICc, korelasi dan R^2 .
9. Melakukan proses testing/forecasting, mentransformasi kembali dalam skala awal dan mengukur kinerjanya.
10. Menetapkan model terbaik untuk setiap studi kasus.



Gambar 3.5 Arsitektur ST-RNN

Proses lengkap untuk membangun model peramalan curah hujan dengan menggunakan ST-RNN dijelaskan pada Gambar 3.6.



Gambar 3.6 Proses pemodelan ST-RNN

3.5 Pemodelan hybrid GSTAR dan RNN (GSTAR-RNN)

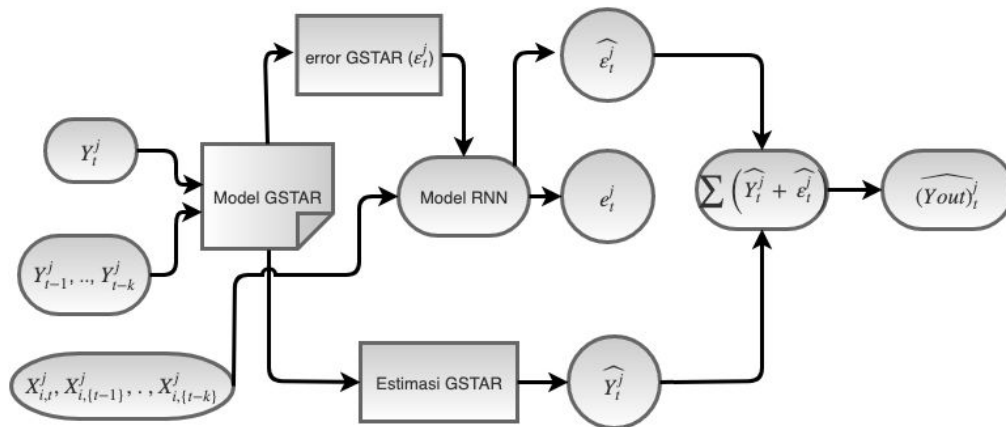
Pengembangan model *Hybrid* GSTAR-RNN dijelaskan pada Gambar 3.7. Input untuk model GSTAR variabel Y_{t-k} dari lokasi j dan sebagai target adalah Y_t . Sedangkan input dari RNN adalah sejumlah variabel prediktor linier/non-linier

$(X_{i,t-k})$ dari variabel prediktor cuaca, indeks iklim dan pengaruh dari curah hujan periode sebelumnya $Y_{i,t-k}$ yang telah diidentifikasi menggunakan PCSUF dengan variabel respon adalah curah hujan pada waktu ke- t .

Luaran dari GSTAR adalah prediksi linier dari curah hujan untuk sejumlah lokasi stasiun cuaca dan nilai *error* dari model GSTAR. Nilai *error* menjadi target dari pemodelan RNN dan sebagai input dari RNN adalah variabel cuaca dan indeks iklim yang berpola non-linier. Hasil dari RNN adalah prediksi atas nilai *error* GSTAR yang juga merupakan prediksi curah hujan untuk komponen non linier. Luaran dari model *hybrid* GSTAR-RNN adalah jumlahan dari luaran GSTAR dan luaran RNN sehingga menjadi nilai prediksi curah hujan yang lengkap.

Hasil dari tahap ini adalah prediksi curah hujan dengan prediktor linier (estimasi GSTAR) dan kesalahan/*error* dari model *GSTAR* (ε^j). Estimasi parameter model GSTAR metode *ordinary least square* [145]. Sedangkan penentuan bobot $w_{ij}^{(l)}$ dari GSTAR ditentukan di awal pemodelan.

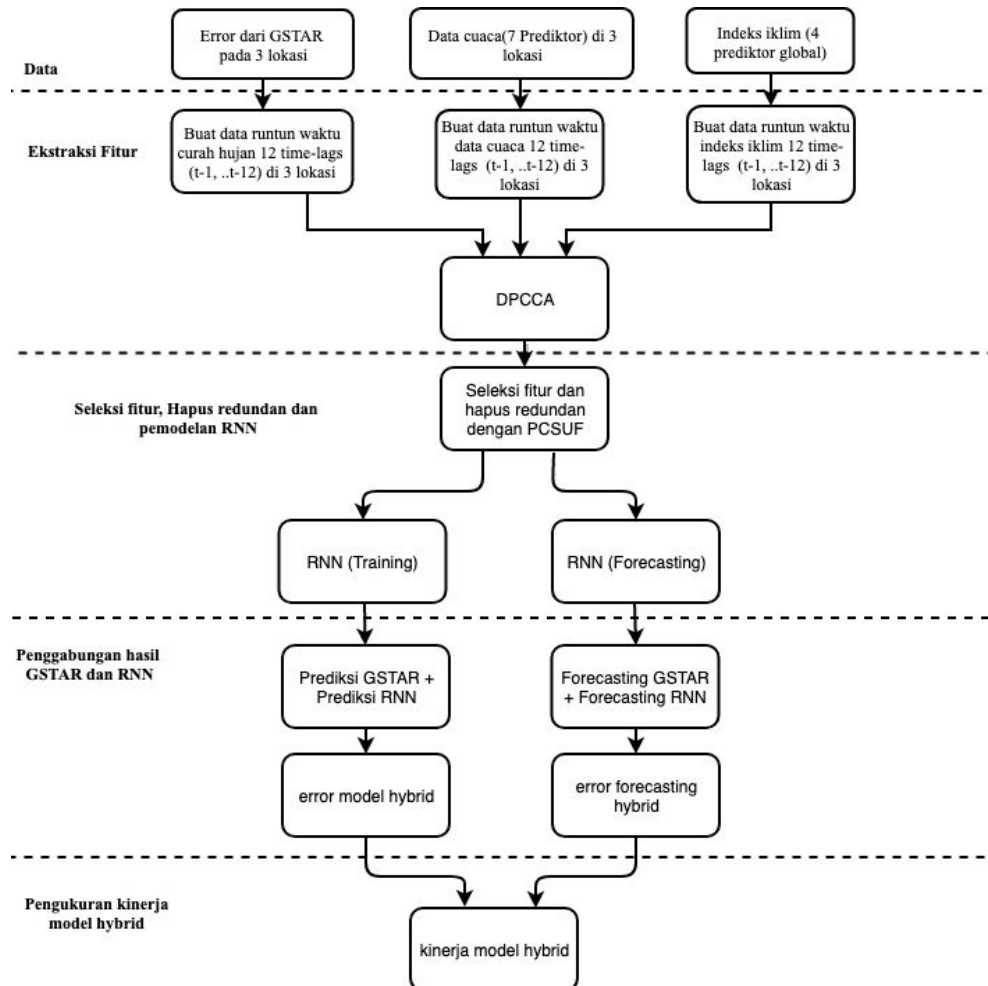
Selanjutnya ε^j menjadi target dari model RNN. Gambar 3.8 adalah detail proses pemodelan GSTAR-ANN. Input dari ANN adalah sejumlah variabel prediktor non linier berdasarkan lokasi stasiun cuaca dan dependensi tenggang waktu yang telah diidentifikasi dengan menggunakan PCSUF. Oleh karenanya, model ANN tersebut dapat diistilahkan sebagai *Spatio-Temporal ANN* [148], [149].



Gambar 3.7 Model hybrid GSTAR-ANN

Estimasi parameter (bobot) dapat menggunakan algoritma *Levenberg–Marquardt* [54], *back-propagation* [67] maupun *Multi Layer Perceptron* [119].

Output dari RNN adalah prediksi dari curah hujan dengan prediktor non linier ($\hat{Y}_{non-linear}$). Hasil keduanya digabungkan untuk memperoleh prediksi akhir, yaitu $\hat{Y} = \hat{Y}_{linier} + \hat{Y}_{non-linear}$ [150].



Gambar 3.8 Proses pemodelan hybrid GSTAR-RNN

3.6 Evaluasi Kinerja

Uji coba untuk mengevaluasi kinerja model dalam meningkatkan kinerja model peramalan data runtun waktu dilakukan pada 4 studi kasus, meliputi data runtun waktu bulanan dan harian. Evaluasi atas pengaruh jumlah prediktor dievaluasi melalui nilai *merit-MV* dan % jumlah input, sedangkan evaluasi terhadap efektifitas model didasarkan pada nilai RMSE, R^2 , r untuk proses training maupun validasi (testing), dan *corrected AIC* untuk proses training.

Uji coba untuk data meteorologi dan iklim, dievaluasi berdasarkan 4 (empat) faktor dan pengaruhnya terhadap efektifitas model yang diusulkan. Hipotesis terhadap faktor-faktor tersebut adalah:

- a. Metode ekstraksi fitur dengan DPCCA dan seleksi fitur dengan PCSUF akan meningkatkan akurasi prediksi.
- b. *Trace-MV* sebagai indikator optimalitas himpunan fitur prediktor konsisten terhadap kinerja model.
- c. Pendekatan spasial (interdependensi antar lokasi stasiun cuaca) akan meningkatkan akurasi prediksi.
- d. Model ST-RNN dan GSTAR-RNN menghasilkan kinerja lebih baik dibandingkan GSTAR.

3.7 Studi Kasus dan Uji Model

Studi kasus untuk pengujian ekstraksi fitur dengan DPCCA, seleksi fitur dengan PCSUF dan peramalan curah hujan dengan model GSTAR, RNN dan *hybrid* adalah pada data periode bulanan dan harian untuk wilayah Surabaya dan Malang-Pasuruan, sehingga terdapat empat studi kasus. Untuk setiap studi kasus, dilakukan uji coba pemodelan sesuai yang disajikan pada Tabel 3.2.

Setiap model akan dievaluasi hasil seleksi fitur berdasarkan *trace-MV* dan jumlah prediktor. Model dengan *trace-MV* yang tinggi dan % jumlah prediktor terkecil menunjukkan optimalitas hasil seleksi fitur. Sedangkan kinerja model dievaluasi berdasarkan nilai AICc, RMSE, R^2 dan r untuk menetapkan model terbaik.

Tabel 3.2 Deskripsi Model GSTAR, RNN, dan hybrid

Model	Metode	Deskripsi
1	GSTAR	<i>Generalized Space-Time Autoregressive</i> dengan 2 dan 3 lokasi dengan variabel curah hujan $t-k$ sebagai prediktor.
2	DPCCA-PCSUF - RNN	<i>Spatio-temporal</i> dengan ekstraksi fitur DPCCA, gabungan PC dan SU untuk memilih fitur relevan dan menghapus redundansi dan peramalan dengan RNN.
3	DPCCA-RNN	<i>Spatio-temporal</i> dengan hanya ekstraksi fitur DPCCA dan peramalan dengan RNN.
4	DPCCA - PC- RNN	<i>Spatio-temporal</i> dengan ekstraksi DPCCA, PC untuk memilih fitur relevan linier, menghapus fitur redundansi dan peramalan dengan RNN.
5	DPCCA - SU-RNN	<i>Spatio-temporal</i> dengan ekstraksi DPCCA, SU untuk memilih fitur relevan non-linier, menghapus fitur redundansi dan peramalan dengan RNN.
6	DPCCA-PCSUF tanpa menghapus redundansi-RNN	<i>Spatio-temporal</i> dengan ekstraksi DPCCA, PC dan SU untuk memilih fitur relevan linier dan non-linier (tanpa menghapus fitur redundansi), peramalan dengan RNN.
7	Data asli-RNN	<i>Spatio-temporal</i> tanpa ekstraksi DPCCA maupun PCSUF, peramalan dengan RNN
8	Hybrid : GSTAR-DPCCA-PCSUF-RNN	GSTAR, Ekstraksi Fitur, Seleksi Fitur dengan PCSUF, Peramalan dengan RNN, Penggabungan
9	Hybrid : GSTAR-DPCCA-RNN	GSTAR, Ekstraksi Fitur, Peramalan dengan RNN, Penggabungan
10	Hybrid : GSTAR-DPCCA-PCF-RNN	GSTAR, Ekstraksi Fitur, Seleksi Fitur dengan PCF, Peramalan dengan RNN, Penggabungan
11	Hybrid : GSTAR-DPCCA-SUF-RNN	GSTAR, Ekstraksi Fitur, Seleksi Fitur dengan SUF, Peramalan dengan RNN, Penggabungan
12	Hybrid : GSTAR-DPCCA- seleksi fitur relevan (tanpa hapus redundansi)-RNN	GSTAR, Ekstraksi Fitur, Seleksi Fitur tanpa menghapus redundansi, Peramalan dengan RNN, Penggabungan

Halaman ini sengaja dikosongkan

BAB 4 HASIL DAN PEMBAHASAN

4.1 Karakteristik Studi Kasus

Pada bab 4 ini dijelaskan hasil uji coba dan pembahasan untuk 4 (empat) studi kasus model penduga curah hujan, yaitu: studi kasus di wilayah Surabaya periode bulanan (studi kasus 1), studi kasus di wilayah Malang periode bulanan (studi kasus 2), studi kasus di wilayah Surabaya periode harian (studi kasus 3), dan studi kasus di wilayah Malang periode harian (studi kasus 4). Untuk wilayah Surabaya, terdapat 3 stasiun cuaca, yaitu Juanda (WMO ID:96935), Perak 1 (WMO ID: 96933) dan Perak 2 (WMO ID: 96937). Sedangkan Malang-Pasuruan, terdapat 3 stasiun cuaca, yaitu Karangates (WMO ID: 96949), Karangploso (WMO ID: 96943) dan Tretes (WMO ID: 96945).

Stasiun cuaca BMKG yang dipilih adalah UPT kelas I dan II yang menyediakan data pengamatan cuaca cukup lengkap. Sedangkan untuk indeks iklim, untuk semua stasiun cuaca di Jawa Timur adalah sama karena bersifat global.

Tahap ekstraksi fitur, seleksi fitur dan pemodelan curah hujan dari studi kasus 1 dijelaskan secara detil pada sub-bab 4.2 sampai dengan 4.6 dengan 12 model sesuai dengan Tabel 3.2. Sedangkan studi kasus 2 – 4 dijelaskan secara ringkas pada sub-bab 4.7 – 4.9. Setiap studi kasus menggunakan variabel yang sama untuk setiap wilayah observasi (stasiun cuaca), yaitu:

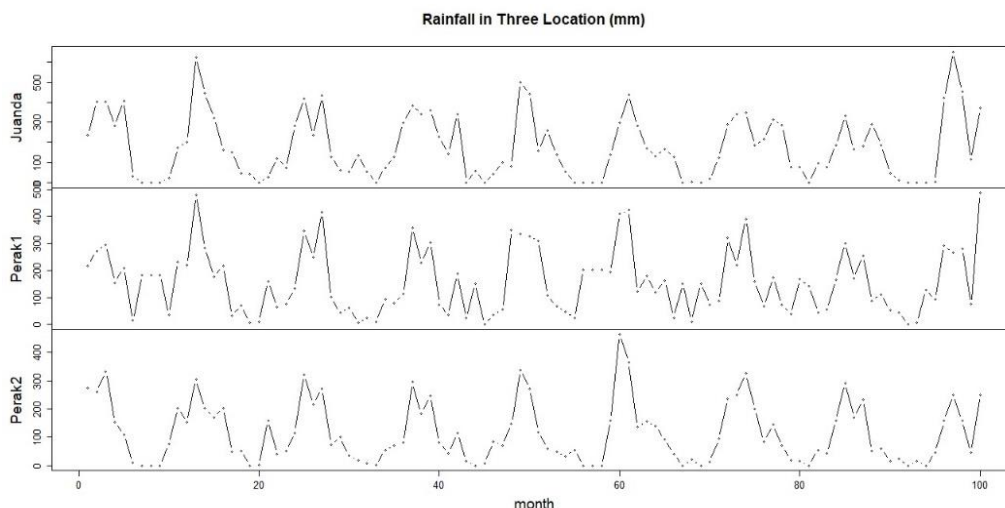
- X_1 = Curah Hujan (mm)
- X_2 = Temperatur minimum (°C)
- X_3 = Temperatur maksimum (°C)
- X_4 = Temperatur rata-rata (°C)
- X_5 = Kelembaban relatif (%)
- X_6 = Lama penyinaran (jam/hari)
- X_7 = Kecepatan angin rata-rata (m/dt)
- X_8 = Kecepatan angin maksimum (m/dt)
- X_9 = Nino1.2
- X_{10} = Nino3
- X_{11} = Nino3.4
- X_{12} = Nino4

X_1 sebagai variabel respon dan $X_2 - X_{12}$ sebagai variabel prediktor.

Input untuk metode ekstraksi fitur adalah suatu *array* berukuran $n \times m \times s$ di mana n adalah jumlah data, m adalah jumlah variabel dan s adalah jumlah lokasi.

Sedangkan input untuk seleksi fitur adalah suatu *array* berukuran 4 dimensi berukuran $n \times m \times s \times s$. Untuk model penduga curah hujan dengan menggunakan GSTAR, input model adalah sebuah matriks berukuran $n \times s$ karena hanya melibatkan variabel X_t untuk setiap lokasi. Sedangkan model penduga curah hujan menggunakan ST-RNN memiliki input suatu *array* berukuran $n \times m_i \times s$ di mana m_i adalah jumlah fitur prediktor yang terpilih untuk setiap lokasi ke- i .

Salah satu indikasi bahwa pendekatan *spatio-temporal* adalah memenuhi syarat jika pola data curah hujan di suatu lokasi memiliki kemiripan dengan curah hujan di lokasi yang lain. Cara mengidentifikasinya adalah dengan membuat plot data runtun waktu dan menghitung korelasi data curah hujan pada sejumlah lokasi. Pada Gambar 4.1 disajikan grafik pola hujan di 3 lokasi, yaitu Juanda, Perak 1 dan Perak 2. Selanjutnya pada Gambar 4.2 adalah hasil perhitungan korelasi untuk data curah hujan pada 3 lokasi.



Gambar 4.1 Pola curah hujan periode bulanan pada 3 lokasi

Dari Gambar 4.1, terdapat kemiripan pola curah hujan periode bulanan di 3 lokasi. Hal ini juga diperkuat dengan nilai korelasi data curah hujan antar lokasi yang semuanya signifikan dan bernilai $> 0,7$ yang ditunjukkan pada Gambar 4.2. Oleh karenanya, pendekatan *spatio-temporal* untuk model penduga curah hujan periode bulanan adalah sesuai dikarenakan adanya dependensi antar lokasi.

```

> cor.test(Data[,1], Data[,2])

        Pearson's product-moment correlation

data:  Data[, 1] and Data[, 2]
t = 17.239, df = 304, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6415304 0.7556362
sample estimates:
      cor
0.7030809

> cor.test(Data[,1], Data[,3])

        Pearson's product-moment correlation

data:  Data[, 1] and Data[, 3]
t = 21.488, df = 304, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.7277757 0.8174808
sample estimates:
      cor
0.7765335

> cor.test(Data[,2], Data[,3])

        Pearson's product-moment correlation

data:  Data[, 2] and Data[, 3]
t = 25.178, df = 304, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.7820922 0.8553966
sample estimates:
      cor
0.822123

```

Gambar 4.2 Uji korelasi data curah hujan pada 3 lokasi

4.2 Ekstraksi Fitur menggunakan metode DPCCA

Langkah pertama dalam seleksi fitur untuk membangun model penduga curah hujan adalah melakukan ekstraksi fitur prediktor. DPCCA lebih efektif dalam mengidentifikasi korelasi silang pada data runtun waktu yang bersifat non-linier, non-stasioner dan tidak dipengaruhi oleh variabel eksternal dibanding ekstraksi fitur dengan CCF.

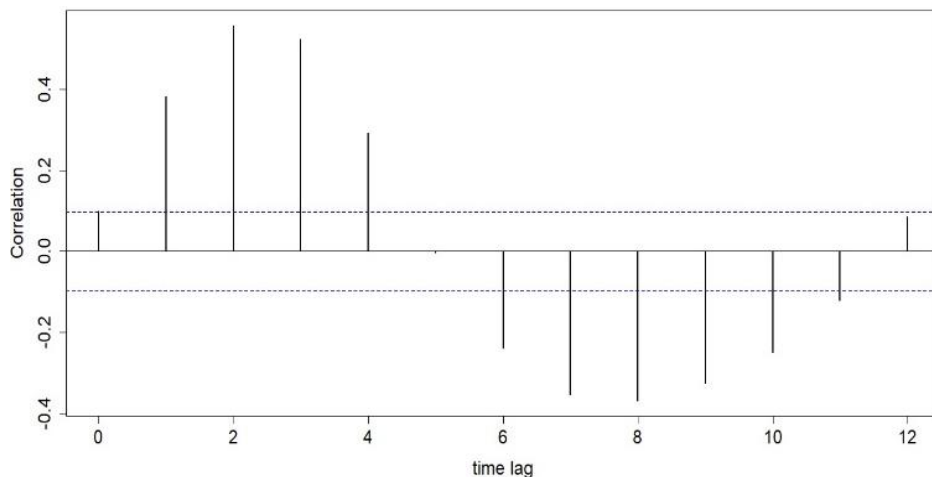
Sebagai perbandingan, pada Gambar 4.3 ditunjukkan hasil identifikasi korelasi dengan menggunakan metode DPCCA dan metode CCF antara variabel curah hujan (X_1) terhadap suhu minimum(X_2). ρ_{DPCCA} memiliki nilai lebih besar dibandingkan nilai ρ_{CCF} pada sejumlah *time-lags* dan lebih banyak yang berhasil

diidentifikasi antara lain karena kemampuannya menghilangkan pengaruh variabel eksternal dan adaptif terhadap pola non-linier [96], [97].

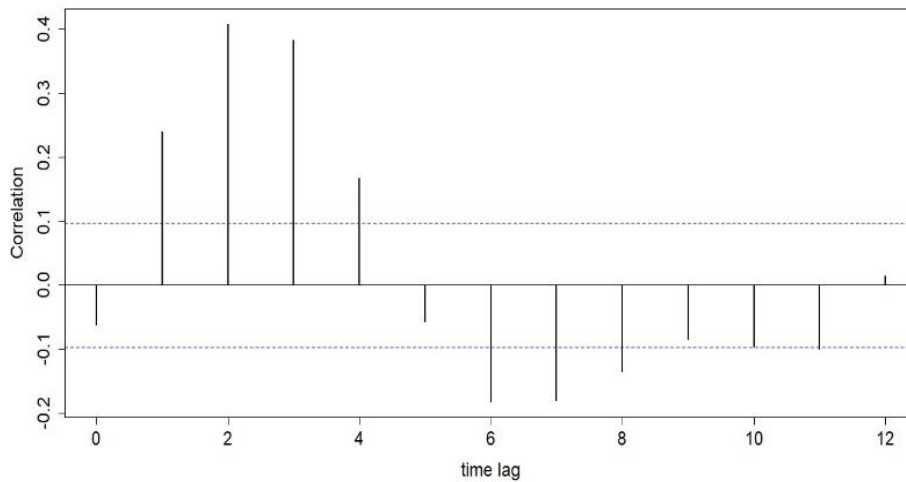
Ekstraksi fitur dengan DPCCA pada penelitian ini melibatkan faktor lokasi, waktu dan variabel. Untuk studi kasus 1, yaitu data periode bulanan di wilayah Surabaya, ρ_{DPCCA} disajikan sebagai *array* 4 dimensi berukuran $13 \times 12 \times 3 \times 3$. Pada Tabel 4.1 disajikan matriks nilai ρ_{DPCCA} dari lokasi 1 (Juanda). Nilai ρ_{DPCCA} dibandingkan dengan koefisien $\pm 2/\sqrt{n}$ di mana n adalah banyaknya data dan k adalah maksimal *time-lags* yang dihitung.

Apabila $|\rho_{DPCCA}| > |2/\sqrt{n}|$ antara variabel curah hujan dengan suatu variabel prediktor, maka prediktor pada lokasi dan *time-lags* tersebut adalah sebagai kandidat prediktor. Dari ukuran fitur hasil ekstraksi *time-lags* sebanyak 468, diperoleh kandidat prediktor untuk curah hujan di Juanda adalah 301 fitur yang nilai dari $|\rho_{DPCCA}|$ memenuhi batas signifikansi, yaitu $> 0,05$.

Selanjutnya dengan cara yang sama, ditentukan kandidat prediktor untuk curah hujan di wilayah Perak 1 dan Perak 2. Tabel 4.2 adalah daftar kandidat prediktor curah hujan di Juanda. Pada kolom X_t dengan prediktor $t - l$ adalah dimaksudkan bahwa curah hujan di Juanda pada saat ke- t berkorelasi secara signifikan dengan curah hujan di Juanda pada saat $t - l$.



(a) DPCCA



(b) CCF

Gambar 4.3 Perbandingan DPCCA dengan CCF

Karena keunggulan dari DPCCA dalam mengidentifikasi fitur yang saling berkorelasi baik linier maupun non-linier, maka diperoleh kandidat fitur prediktor yang masih cukup besar, yaitu 64,3% dari total fitur hasil ekstraksi. Hal ini menyebabkan potensi terjadinya redundansi antar fitur prediktor sangat tinggi sehingga diperlukan proses seleksi fitur prediktor untuk memperoleh fitur yang relevan dan tidak terjadi redundansi antar prediktor.

Tabel 4.1 Nilai ρ_{DPCCA} dari lokasi 1

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12
Lag ke-0	1.00000	0.47871	0.06290	-0.10775	0.78803	-0.74391	-0.28956	0.42965	-0.06978	-0.06203	-0.06608	-0.07710
Lag ke-1	0.66913	0.49186	0.42119	0.21447	0.48147	-0.68880	-0.29814	0.31513	-0.07812	-0.05317	-0.02313	-0.02892
Lag ke-2	0.23148	0.40785	0.67915	0.45727	0.01943	-0.43364	-0.22995	0.16041	-0.06691	-0.02722	0.02351	0.02887
Lag ke-3	-0.13302	0.15829	0.68148	0.43451	-0.40382	-0.07945	-0.02825	0.00059	-0.04127	0.00335	0.05565	0.07043
Lag ke-4	-0.40646	-0.20721	0.40368	0.15907	-0.62760	0.25697	0.20900	-0.08161	-0.00687	0.03673	0.06836	0.07368
Lag ke-5	-0.55784	-0.52980	-0.00344	-0.13937	-0.64189	0.50166	0.34784	-0.18645	0.05214	0.06910	0.07498	0.06780
Lag ke-6	-0.59453	-0.65083	-0.30342	-0.27727	-0.50642	0.60607	0.33599	-0.29808	0.10720	0.10263	0.07556	0.03796
Lag ke-7	-0.50975	-0.52305	-0.44846	-0.22587	-0.32231	0.59579	0.24871	-0.33848	0.13255	0.12744	0.07333	-0.00992
Lag ke-8	-0.31994	-0.25002	-0.48242	-0.11156	-0.07924	0.45595	0.09222	-0.29826	0.11338	0.12872	0.07433	-0.02578
Lag ke-9	-0.05259	0.03833	-0.42898	-0.05818	0.21345	0.18739	-0.07488	-0.13573	0.08547	0.09137	0.05520	-0.02490
Lag ke-10	0.23789	0.24682	-0.33502	-0.10895	0.46646	-0.14632	-0.15831	0.12522	0.04581	0.03694	0.01910	-0.01130
Lag ke-11	0.53173	0.38339	-0.16542	-0.15646	0.63591	-0.48088	-0.15341	0.33163	-0.00022	-0.02520	-0.03478	-0.01021
Lag ke-12	0.70282	0.44730	0.07293	-0.05326	0.63932	-0.68277	-0.21370	0.39895	-0.05991	-0.09827	-0.09649	-0.03638

Tabel 4.2 Kandidat prediktor untuk curah hujan di Juanda ($X_{1,t(1)}$)

Variabel respon: $X_{1,t(1)}$												
Faktor spasial	Prediktor											
	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}
Juanda	t-1	t	t	t	t	t	t	t	t	t	t	t
	t-2	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-3	t-3
	t-3	t-2	t-2	t-2	t-3	t-2	t-2	t-2	t-2	t-5	t-4	t-4
	t-4	t-3	t-3	t-3	t-4	t-3	t-4	t-4	t-5	t-6	t-5	t-5
	t-5	t-4	t-4	t-4	t-5	t-4	t-5	t-5	t-6	t-7	t-6	
	t-6	t-5	t-6	t-5	t-6	t-5	t-6	t-6	t-7	t-8	t-7	
	t-7	t-6	t-7	t-6	t-7	t-6	t-7	t-7	t-8	t-9	t-8	
	t-8	t-7	t-8	t-7	t-8	t-7	t-8	t-8	t-9	t-12	t-9	
	t-9	t-8	t-10	t-8	t-9	t-8	t-9	t-9	t-12		t-12	
	t-10	t-9	t-11	t-9	t-10	t-9	t-10	t-10				
	t-11	t-10	t-12	t-10	t-11	t-10	t-11	t-11				
	t-12	t-11		t-11	t-12	t-11	t-12	t-12				
	t-12		t-12		t-12							
Perak1	t-1	t	t	t	t	t	t	t	t	t	t	t
	t-2	t-1	t-1	t-2	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1
	t-3	t-2	t-2	t-3	t-2	t-2	t-2	t-2	t-2	t-4	t-5	t-3
	t-4	t-4	t-3	t-4	t-3	t-3	t-4	t-3	t-5	t-6	t-6	t-4
	t-5	t-5	t-4	t-5	t-4	t-4	t-5	t-5	t-6	t-7	t-7	t-5
	t-6	t-6	t-5	t-7	t-5	t-5	t-6	t-6	t-7	t-8	t-8	
	t-7	t-7	t-6	t-8	t-6	t-6	t-7	t-7	t-8		t-9	
	t-9	t-8	t-7	t-9	t-8	t-7	t-9	t-8				
	t-10	t-9	t-8	t-10	t-9	t-8	t-10	t-9				
	t-11	t-10	t-9	t-12	t-10	t-9	t-11	t-10				
	t-12	t-11	t-10		t-11	t-10	t-12	t-11				
		t-12	t-11	t-12	t-12	t-11		t-12				
		t-12			t-12							
Perak2	t-1	t	t	t	t	t	t	t	t	t	t	t
	t-2	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1	t-1
	t-3	t-2	t-2	t-2	t-2	t-2	t-2	t-2	t-2	t-2	t-2	t-5
	t-4	t-4	t-3	t-3	t-3	t-3	t-4	t-3	t-5	t-5	t-4	t-6
	t-5	t-5	t-4	t-4	t-4	t-4	t-5	t-4	t-6	t-6	t-5	t-7
	t-6	t-6	t-5	t-5	t-5	t-5	t-6	t-5	t-7	t-7	t-6	t-8
	t-7	t-7	t-6	t-6	t-6	t-6	t-7	t-6	t-8	t-8	t-7	t-11
	t-8	t-8	t-7	t-7	t-7	t-7	t-10	t-7	t-9	t-9	t-8	t-12
	t-9	t-9	t-8	t-8	t-8	t-8	t-11	t-8	t-10	t-10	t-9	
	t-10	t-10	t-9	t-9	t-9	t-10	t-12	t-9		t-12	t-12	
	t-11	t-11	t-10	t-10	t-10	t-11		t-10				
	t-12	t-12	t-11	t-11	t-11	t-12		t-11				
		t-12		t-12			t-12					

4.3 Seleksi Fitur menggunakan metode PCSUF

Proses selanjutnya adalah seleksi fitur prediktor dengan menggunakan metode PCSUF atau variasinya untuk memperoleh fitur relevan dan tidak terjadi redundansi. Proses seleksi fitur mengikuti tahap-tahap sesuai yang dijelaskan pada sub-bab 3.2.2. Sebagai contoh, dilakukan seleksi fitur non-linier dengan

menggunakan *Symmetrical Uncertainty Filter* (SUF) sehingga diperoleh daftar fitur prediktor pada setiap wilayah sesuai dengan Tabel 4.3.

Tabel 4.3 Fitur prediktor hasil seleksi dengan SUF

Variabel Respon: $X_{1,t(1)}$ (Juanda)						
	Prediktor					
Lokasi	X₁	X₂	X₃	X₄	X₅	X₆
Juanda	-	-	-	-	t t-11 t-12	t t-6 t-7
Perak1	t-4 t-5 t-11 t-12	-	-	t-2 t-7 t-8	t-4 t-5	t t-11
Perak2	t-5 t-6	-	-	-	t t-11	t t-1 t-11
Variabel Respon: $X_{2,t(1)}$ (Perak 1)						
	Predictors					
Lokasi	X₁	X₂	X₃	X₄	X₅	X₆
Juanda	-	t-1	t-3 t-4	-	t t-3 t-11	-
Perak1	t-1 t-11	-	-	t-3	t t-3	-
Perak2	t-1 t-6 t-7	t-1 t-6	-	-	t t-11	-
Variabel Respon : $X_{3,t(1)}$						
	Predictors					
Lokasi	X₁	X₂	X₃	X₄	X₅	X₆
Juanda	-	-	t-3	-	t t-6 t-7	t t-11
Perak1	t-7 t-11	-	-	-	t-4 t-5	t t-1 t-5 t-6 t-11 t-12
Perak2	t-5 t-7 t-11	-		-	t	t t-7

Jumlah prediktor curah hujan di Juanda adalah 24, sedangkan di Perak 1 dan di Perak 2 adalah masing-masing 18 dan 22 sehingga secara rata-rata jumlah prediktor di tiap wilayah adalah 22 fitur. Dengan menggunakan ukuran optimalitas seleksi fitur, maka diperoleh matriks *MV* sebagai berikut:

$$\begin{pmatrix} 0.2947 & 0.2653 & 0.2839 \\ 0.1924 & 0.2208 & 0.1727 \\ 0.2905 & 0.2552 & 0.2790 \end{pmatrix}$$

Untuk kemudahan evaluasi sebagai ukuran kesesuaian hasil seleksi fitur, maka matriks MV ditransformasi menjadi nilai skalar sebagai $trace-MV = 0,806$.

Hasil seleksi fitur dengan pilihan variasi PCSUF ditunjukkan pada Tabel 4.4. Berdasarkan $trace MV$, dan % prediktor maka dapat disimpulkan bahwa seleksi fitur dengan SUF adalah yang terbaik karena memiliki nilai $trace$ tertinggi dan % jumlah prediktor terkecil. Persentase prediktor diperoleh dari jumlah prediktor dibagi dengan keseluruhan fitur hasil ekstraksi, yaitu 468.

Tabel 4.4 Perbandingan kinerja seleksi fitur

No.	Metode	Threshold		Trace MV	Jumlah Prediktor	% Pred
		α	β			
1	DPCCA - PCSUF	0,2	0,2	0,703	30	6,4%
2	DPCCA	-	-	0,123	301	64,3%
3	DPCCA-PCF	0,2	-	0,543	38	8,1%
4	DPCCA-SUF	-	0.2	0,806	22	4,7%
5	DPCCA - pilih relevan	-	-	0,186	206	44,0%

Dari hasil seleksi fitur ini, sejumlah kesimpulan dapat diambil sebagai berikut:

1. Nilai $trace$ dari MV tetap dapat dijadikan sebagai ukuran kesesuaian (*goodness of fit*) dari himpunan fitur hasil seleksi dengan pendekatan *spatio-temporal*. Secara teoritis, apabila nilai $MV > 0.5$ adalah dapat diterima [112].
2. *Threshold* untuk penghapusan redundansi fitur linier/non-linier yang menyebabkan $trace-MV$ maksimal dicapai pada nilai $\alpha = 0,2$ dan $\beta = 0,2$. Hal ini menunjukkan bahwa semakin rendah batas nilai redundansi yang ditetapkan atas korelasi dan nilai SU, nilai $trace-MV$ semakin baik.
3. Hasil terbaik seleksi fitur adalah menggunakan DPCCA-SUF (nomer 4) dengan 22 fitur prediktor. Dari himpunan fitur tersebut, dengan menggunakan uji *white.test* diperoleh bahwa 76% fitur terpilih merupakan fitur non-linier.
4. Variabel indeks iklim sebagai prediktor global, terpilih sebagai kandidat fitur melalui proses DPCCA (Tabel 4.2 kolom $X_9 - X_{12}$) namun tidak cukup signifikan sebagai fitur prediktor pada seleksi dengan PCF, SUF dan PCSUF (Tabel 4.3).

5. Prediktor dominan yang terpilih baik menggunakan PCF, SUF maupun PCSUF adalah kelembaban relatif (X_5), lama penyinaran (X_6), dan temperatur udara (min/maks/mean) dengan *time-lags* tertentu ($t - k$) dari sejumlah lokasi.
6. Pengaruh/keterkaitan prediktor antar lokasi cukup signifikan, termasuk pengaruh variabel curah hujan (X_7) di lokasi lain sebagai prediktor (autoregresi) dengan *time-lags* tertentu ($t - k$).

Hasil optimalitas seleksi fitur pada Tabel 4.4 akan diuji konsistensinya pada tahap pemodelan penduga curah hujan menggunakan ST-RNN dan hybrid GSTAR-RNN.

4.4 Model Penduga Curah Hujan dengan GSTAR

Pada bagian ini, dijelaskan pemodelan GSTAR untuk data kasus periode bulanan di wilayah Surabaya. Sesuai yang dijelaskan pada sub-bab 3.3 bahwa pemodelan GSTAR mengikuti tahap sebagai berikut:

1. Menguji sifat stasioner dari data curah hujan di 3 lokasi stasiun cuaca menggunakan *ADF-test*. Hipotesis dari uji stasioneritas dinyatakan :

H_0 : Data tidak stasioner ; H_1 : Data telah stasioner

Untuk 3 wilayah pengamatan, yaitu Juanda, Perak 1 dan Perak 2, diperoleh hasil *ADF-test* pada Gambar 4.4.

Dapat disimpulkan bahwa data curah hujan di 3 lokasi adalah non-stasioner sehingga diperlukan proses *differencing* musiman. Hasil dari proses *differencing* diidentifikasi melalui *correlogram* ACF dan PACF dari data curah hujan 3 lokasi. Pada Gambar 4.5 adalah ACF-PACF dari data curah hujan di lokasi Juanda yang telah melalui proses *differencing* dan menunjukkan telah memenuhi pola stasioner.

```

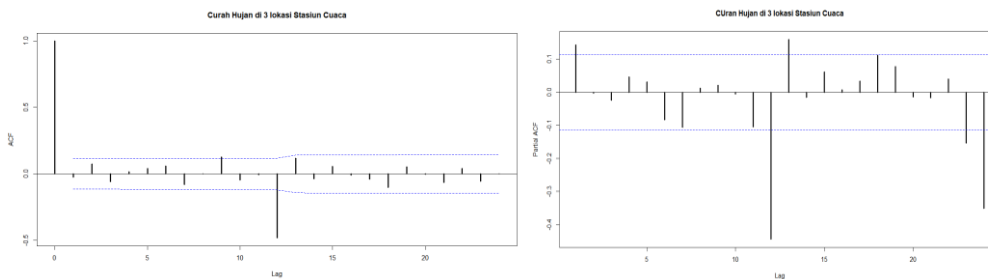
Augmented Dickey-Fuller Test
data: Data[, lok]
Dickey-Fuller = -3.4226, Lag order = 365, p-value = 0.049
74
alternative hypothesis: stationary

data: Data_Asli[, Perak1]
Dickey-Fuller = -3.2147, Lag order = 24, p-value = 0.0855
alternative hypothesis: stationary

data: Data_Asli[, Perak2]
Dickey-Fuller = -2.698, Lag order = 24, p-value = 0.2825
alternative hypothesis: stationary

```

Gambar 4.4 ADF-test untuk data curah hujan di 3 lokasi



(a) ACF

(b) PACF

Gambar 4.5 ACF dan PACF Curah Hujan di Juanda

2. Identifikasi awal untuk menentukan order waktu p dan order λ_p di mana berdasarkan PACF, dapat dicoba dengan nilai $p = 12$ atau 24 dan $\lambda_k = 1$.
3. Menentukan nilai dari matriks bobot dengan 3 metode. Metode terbaik ditentukan berdasarkan ukuran kinerja model GSTAR. Pada Gambar 4.6 dijelaskan perhitungan nilai bobot untuk 3 metode.

```

##Uniform
W<-matrix(c(0,0.5,0.5,0.5,0,0.5,0.5,0.5,0),3,3)

##weight Invers Jarak
r1 <- 17.79
r2 <- 19.49
r3 <- 2.0

W <- matrix(c(0,r2/(r1+r2), r1/(r1+r2), r3/(r1+r3)
, 0, r1/(r1+r3), r3/(r2+r1), r2/(r2+r3),0),3,3)

##weight Normalisasi Korelasi
r12 <- 0.6896245
r13 <- 0.7749152
r23 <- 0.8109261

W <- matrix(c(0,r12/(r12+r13), r13/(r12+r13),
r12/(r12+r13), 0, r23/(r12+r23), r13/(r13+r12),
r23/(r12+r23),0),3,3)

```

Gambar 4.6 Perhitungan nilai bobot dari GSTAR

4. Estimasi parameter model GSTAR(12, 1).

Jumlah parameter dari GSTAR adalah sebanyak 72 dengan 36 parameter pertama adalah untuk order waktu dan 36 parameter kedua adalah untuk parameter *spatial*. Nilai keseluruhan estimasi parameter ditunjukkan pada Gambar 4.7. Selanjutnya nilai estimasi parameter ini digunakan untuk peramalan/testing untuk curah hujan di 3 lokasi.

<pre> ,, 1, 1 [1] [2] [3] [1.] 0.4206652 0.000000 0.0000000 [2.] 0.0000000 -0.128605 0.0000000 [3.] 0.0000000 0.000000 0.3162276 ,, 2, 1 [1] [2] [3] [1.] 0.01129191 0.0000000 0.000000000 [2.] 0.0000000 0.06894872 0.000000000 [3.] 0.0000000 0.0000000 0.009057773 ,, 3, 1 [1] [2] [3] [1.] 0.08603766 0.0000000 0.000000000 [2.] 0.0000000 0.0120826 0.000000000 [3.] 0.0000000 0.0000000 -0.07190331 ,, 4, 1 [1] [2] [3] [1.] -0.04008609 0.0000000 0.000000000 [2.] 0.0000000 0.08747763 0.000000000 [3.] 0.0000000 0.0000000 -0.08516705 ,, 5, 1 [1] [2] [3] [1.] -0.02420594 0.0000000 0.00000000000 [2.] 0.0000000 0.09635135 0.00000000000 [3.] 0.0000000 0.0000000 0.0004365632 ,, 6, 1 [1] [2] [3] [1.] -0.1039024 0.0000000 0.0000000 [2.] 0.0000000 0.04763486 0.0000000 [3.] 0.0000000 0.0000000 -0.0672189 ,, 7, 1 [1] [2] [3] [1.] 0.1357795 0.000000 0.0000000 [2.] 0.0000000 -0.139065 0.0000000 [3.] 0.0000000 0.000000 -0.1636254 ,, 8, 1 [1] [2] [3] [1.] -0.01232212 0.0000000 0.0000000 [2.] 0.0000000 -0.0735526 0.0000000 [3.] 0.0000000 0.0000000 -0.1126429 ,, 9, 1 [1] [2] [3] [1.] 0.1008405 0.0000000 0.0000000 [2.] 0.0000000 0.1049591 0.0000000 [3.] 0.0000000 0.0000000 -0.1130653 ,, 10, 1 [1] [2] [3] [1.] -0.02315266 0.0000000 0.000000000 [2.] 0.0000000 -0.1012834 0.000000000 [3.] 0.0000000 0.0000000 0.09264941 ,, 11, 1 [1] [2] [3] [1.] -0.05861473 0.0000000 0.000000000 [2.] 0.0000000 0.1044217 0.000000000 [3.] 0.0000000 0.0000000 0.000435178 ,, 12, 1 [1] [2] [3] [1.] 0.2449235 0.0000000 0.000000 [2.] 0.0000000 0.1102588 0.000000 [3.] 0.0000000 0.0000000 0.20194 </pre>	<pre> ,, 1, 2 [1] [2] [3] [1.] -0.3432568 0.0000000 0.000000000 [2.] 0.0000000 0.3107412 0.000000000 [3.] 0.0000000 0.0000000 -0.05116365 ,, 2, 2 [1] [2] [3] [1.] -0.0634079 0.0000000 0.000000000 [2.] 0.0000000 0.05210725 0.000000000 [3.] 0.0000000 0.0000000 0.0519358 ,, 3, 2 [1] [2] [3] [1.] -0.8745812 0.000000000 0.000000000 [2.] 0.0000000 -0.008677959 0.000000000 [3.] 0.0000000 0.000000000 0.03589975 ,, 4, 2 [1] [2] [3] [1.] 0.9018405 0.0000000 0.000000000 [2.] 0.0000000 -0.09945892 0.000000000 [3.] 0.0000000 0.0000000 0.0217959 ,, 5, 2 [1] [2] [3] [1.] 0.1476234 0.0000000 0.000000000 [2.] 0.0000000 -0.1262124 0.000000000 [3.] 0.0000000 0.0000000 -0.01515245 ,, 6, 2 [1] [2] [3] [1.] 0.4653045 0.0000000 0.000000000 [2.] 0.0000000 0.0495127 0.000000000 [3.] 0.0000000 0.0000000 0.03206288 ,, 7, 2 [1] [2] [3] [1.] -1.859292 0.000000 0.000000000 [2.] 0.000000 0.105407 0.000000000 [3.] 0.000000 0.000000 0.06573951 ,, 8, 2 [1] [2] [3] [1.] -0.5630961 0.0000000 0.000000000 [2.] 0.0000000 0.08892365 0.000000000 [3.] 0.0000000 0.0000000 0.04206111 ,, 9, 2 [1] [2] [3] [1.] -0.219793 0.0000000 0.000000000 [2.] 0.000000 0.03374064 0.000000000 [3.] 0.000000 0.0000000 0.0953129 ,, 10, 2 [1] [2] [3] [1.] 1.102847 0.0000000 0.000000000 [2.] 0.000000 0.1450623 0.000000000 [3.] 0.000000 0.0000000 0.01200546 ,, 11, 2 [1] [2] [3] [1.] 1.789402 0.0000000 0.000000000 [2.] 0.000000 -0.0235609 0.000000000 [3.] 0.000000 0.0000000 0.08434204 ,, 12, 2 [1] [2] [3] [1.] 1.406311 0.0000000 0.000000000 [2.] 0.000000 0.1088857 0.000000000 [3.] 0.000000 0.0000000 0.1051041 </pre>
--	---

Gambar 4.7 Estimasi parameter GSTAR (12, 1)

5. Kinerja model GSTAR

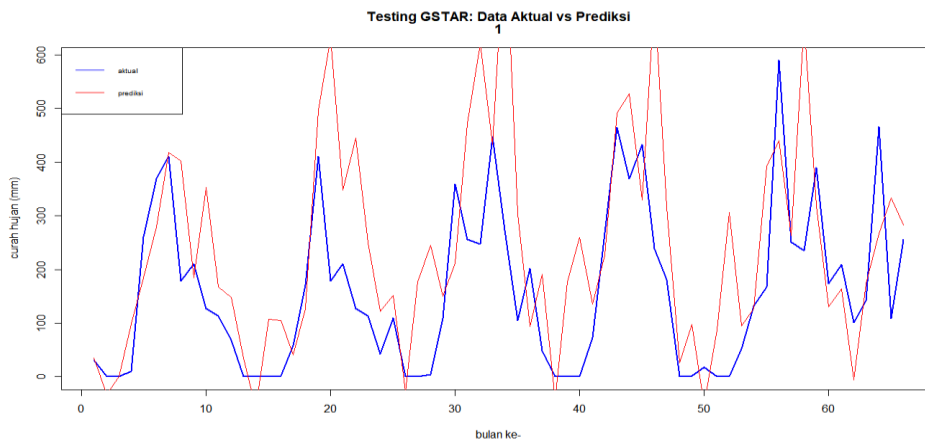
Kinerja model GSTAR berdasarkan RMSE, R^2 dan r dari tahap peramalan, yang terbaik adalah GSTAR dengan metode perhitungan bobot menggunakan nilai *uniform* (nomor 1) sesuai yang ditunjukkan pada Tabel 4.5.

Tabel 4.5 Kinerja model GSTAR berdasarkan metode bobot

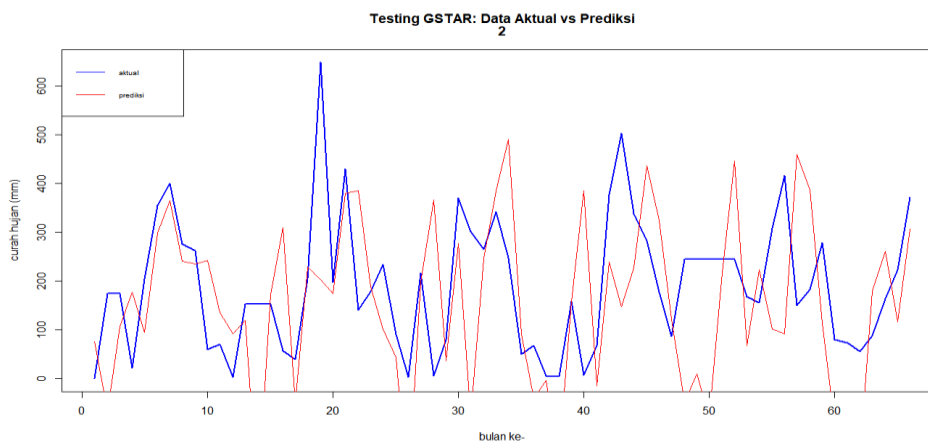
No.	Metode bobot	RMSE		R^2		Korelasi		% Training
		Model	Forecast	Model	Forecast	Model	Forecast	
1	uniform	350,98	468,46	0,430	0,281	0,655	0,515	75
2	jarak	349,48	470,32	0,432	0,272	0,656	0,510	75
3	korelasi	351,21	468,52	0,430	0,280	0,654	0,514	75

6. Testing/forecasting

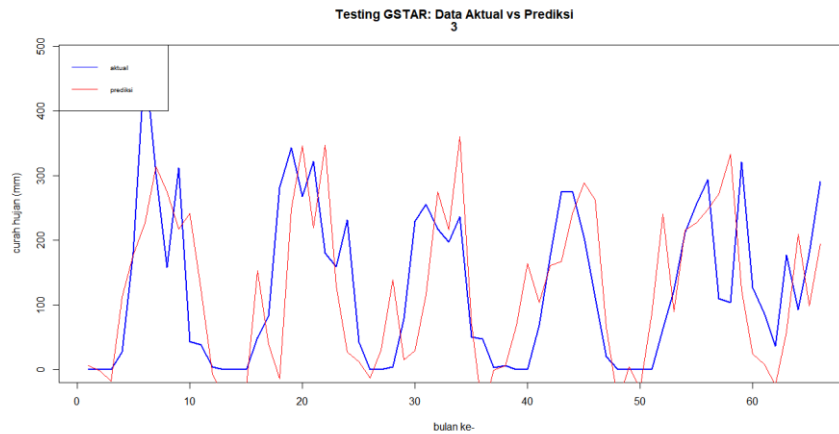
Pada Gambar 4.8 – 4.10 ditunjukkan perbandingan nilai aktual terhadap nilai peramalan untuk curah hujan di 3 wilayah dengan menggunakan GSTAR.



Gambar 4.8 Peramalan GSTAR untuk curah hujan di Juanda

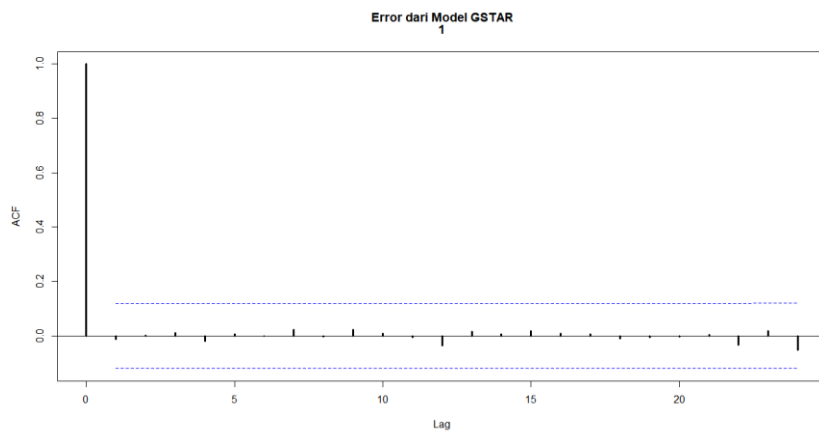


Gambar 4.9 Peramalan GSTAR untuk curah hujan di Perak 1

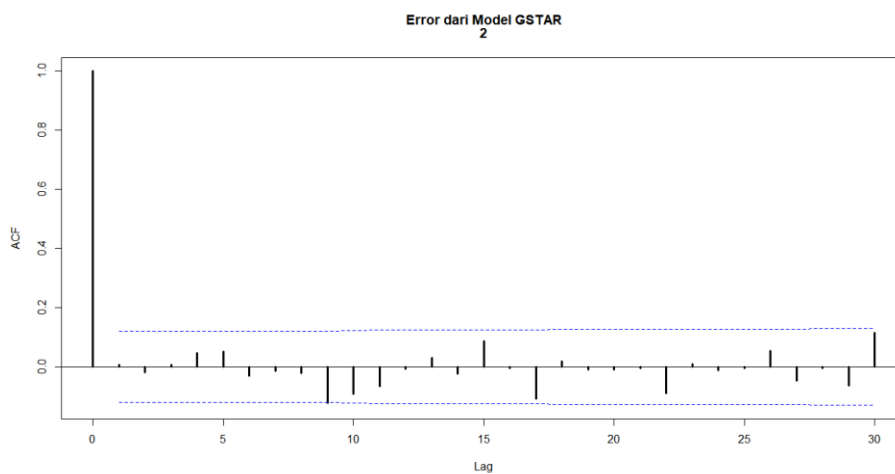


Gambar 4.10 Peramalan GSTAR untuk curah hujan di Perak 2

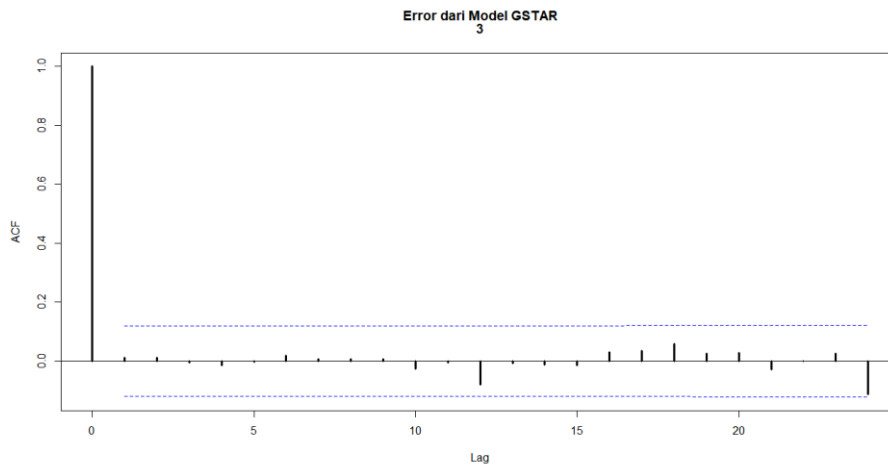
7. Analisis identik dan saling bebas (*independent*) terhadap *error* dari GSTAR. Dari Gambar 4.11 – 4.13 menunjukkan bahwa ACF dari residual telah *white-noise* yang artinya *error* dari GSTAR telah saling bebas (*independent*).



Gambar 4.11 ACF dari residual model GSTAR untuk wilayah Juanda



Gambar 4.12 ACF dari residual model GSTAR untuk wilayah Perak 1



Gambar 4.13 ACF dari residual model GSTAR untuk wilayah Perak 2

4.5 Pemodelan ST-RNN

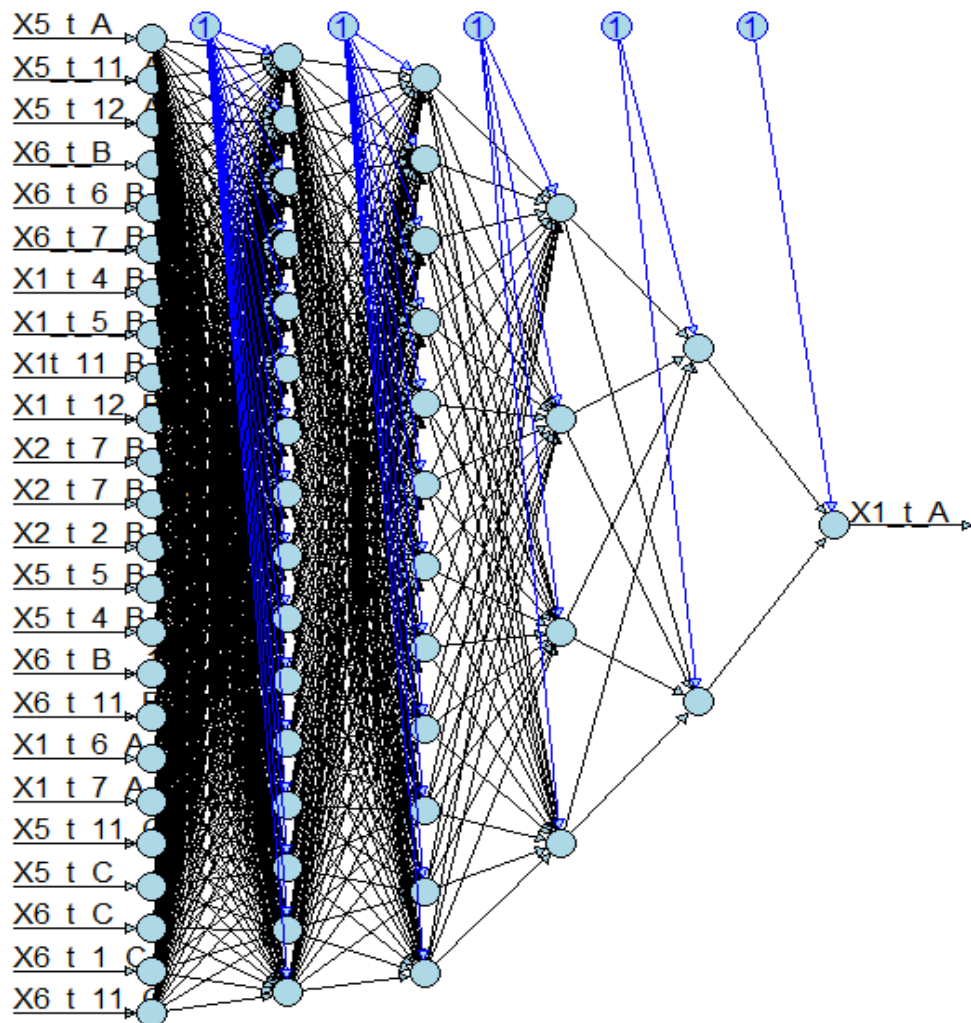
Sesuai yang dijelaskan pada sub-bab 3.4, maka tahap pemodelan ST-RNN untuk curah hujan di 3 wilayah dari studi kasus data periode bulanan di Surabaya adalah sebagai berikut:

1. Melakukan ekstraksi dan seleksi fitur untuk memperoleh input optimal dari RNN. Pada Tabel 4.4 telah diperoleh 22 fitur input dari hasil seleksi fitur DPCCA-SUF dengan *trace*-MV yang terbaik.
2. Data curah hujan ($X_{I, t(s)}$) pada 3 lokasi, yaitu Juanda, Perak 1 dan Perak 2 menjadi nilai target dari RNN.
3. Persentase data training ditetapkan 75% sehingga untuk proses testing menggunakan 25% dari total data 408 periode bulanan.
4. Melakukan normalisasi data prediktor dan curah hujan.
5. Menetapkan arsitektur dari jaringan RNN sebagaimana pada Gambar 4.14.

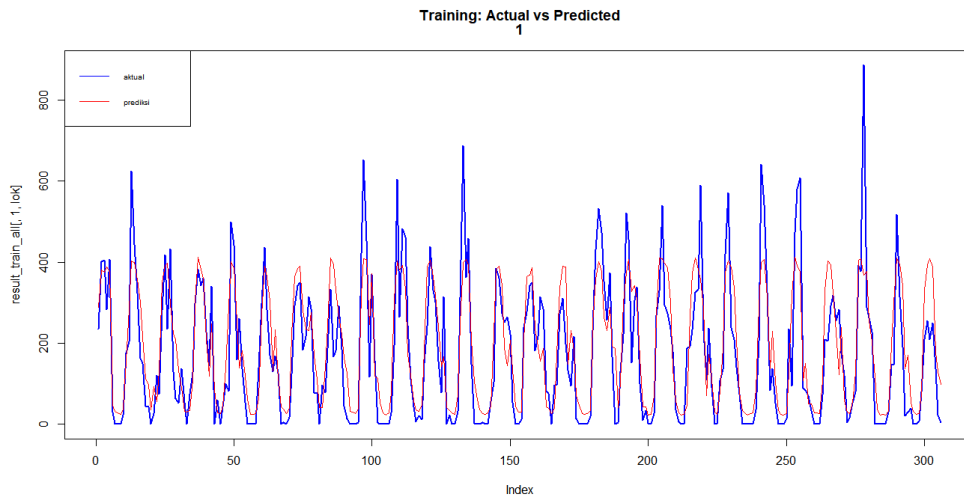
```
## create your model and add layers
set.seed(1000)
model<-trainr(Y=Y_train, X=X_train,
  learningrate=0.1, hidden_dim=c(16, 12, 4, 2),
  numepochs=875,
  momentum=0.8, use_bias = F, learningrate_decay = 1,
  seq_to_seq_unsync = F, update_rule = "sgd",
  network_type="rnn",
  epoch_function = c(epoch_print, epoch_annealing),
  loss_function = loss_L1,
  sigmoid = c("logistic", "Gompertz", "tanh"))
```

Gambar 4.14 Arsitektur dan parameter RNN

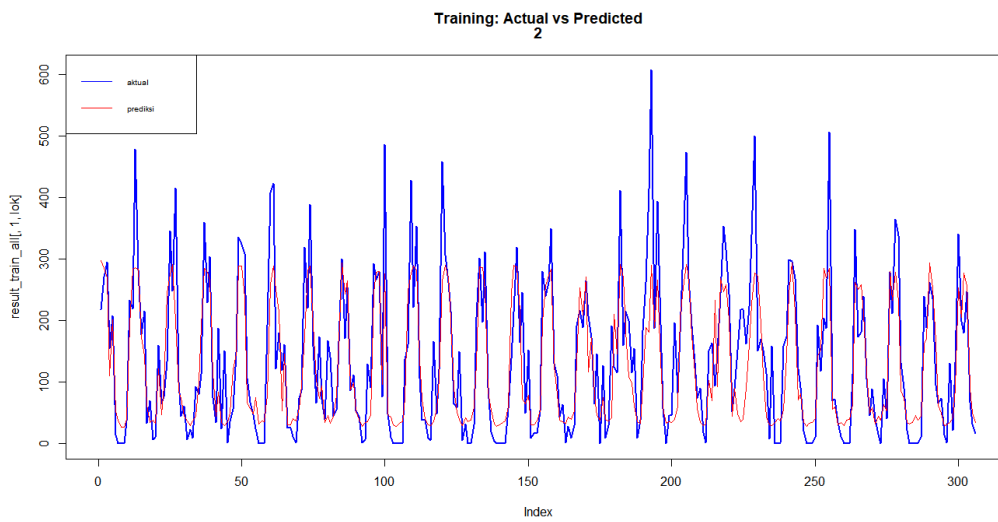
Gambar 4.15 adalah visualisasi ST-RNN untuk wilayah Juanda di mana A, B dan C adalah kode untuk wilayah Juanda, Perak 1 dan Perak 2. Dari Gambar 4.15, variabel respon $X1_t_A$ mewakili nilai prediksi curah hujan pada waktu ke- t di wilayah Juanda dengan 24 fitur input, diantaranya adalah kelembaban relatif (X_5) di wilayah Juanda pada saat ke- t , $t-11$, $t-12$, lama penyinaran matahari (X_6) di wilayah Perak 1 pada saat ke- t , $t-6$, $t-7$ sampai dengan fitur X_6 di wilayah Perak 2 pada saat ke- t , $t-1$ dan $t-11$.



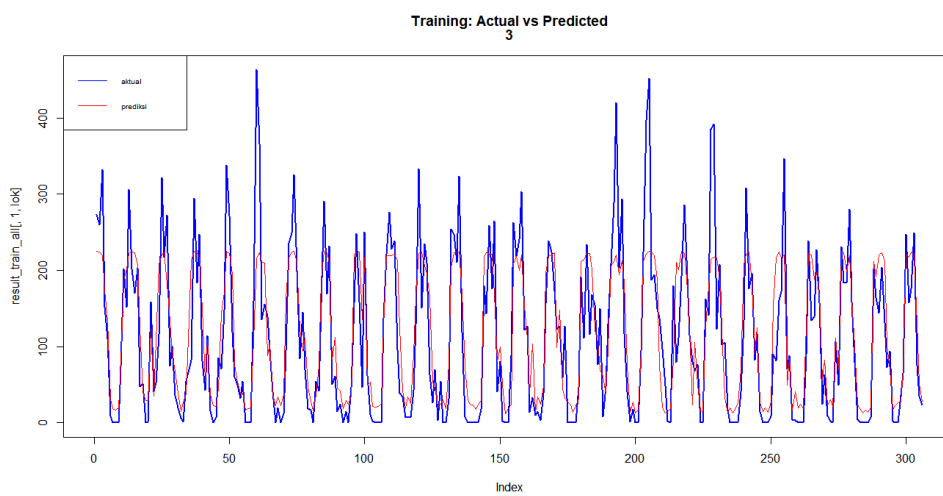
Gambar 4.15 Visualisasi Jaringan untuk ST-RNN lokasi Juanda



Gambar 4.16 Hasil training untuk curah hujan di wilayah Juanda



Gambar 4.17 Hasil training untuk curah hujan di wilayah perak 1



Gambar 4.18 Hasil training untuk curah hujan di wilayah Perak 2

Dari gambar hasil training untuk ketiga lokasi, memiliki kesesuaian antara nilai aktual dengan nilai prediksi dengan masing-masing nilai korelasi adalah 0,861; 0,843 dan 0,863 yang secara rata-rata, nilai korelasi antara data aktual curah hujan dengan data prediksi adalah 0,856 atau nilai indeks R^2 adalah 0,732.

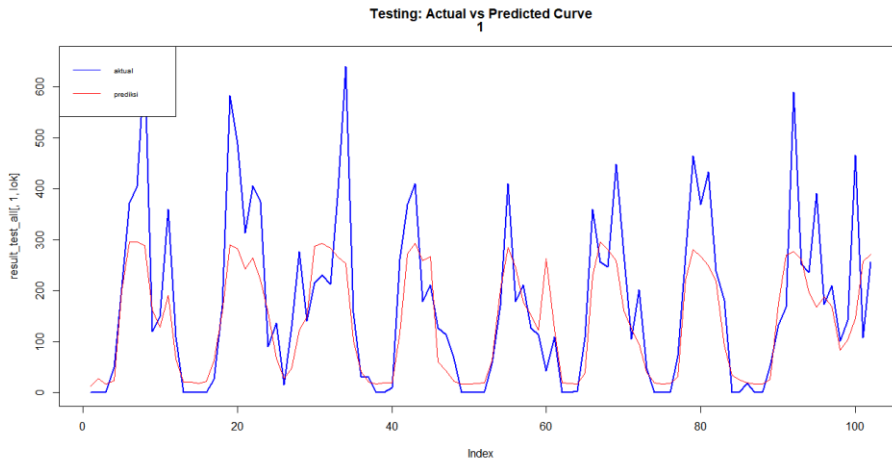
- Melakukan testing dan mengukur kinerja model dengan hasil yang ditunjukkan pada Tabel 4.6.

Tabel 4.6 Kinerja model RNN berdasarkan seleksi fitur

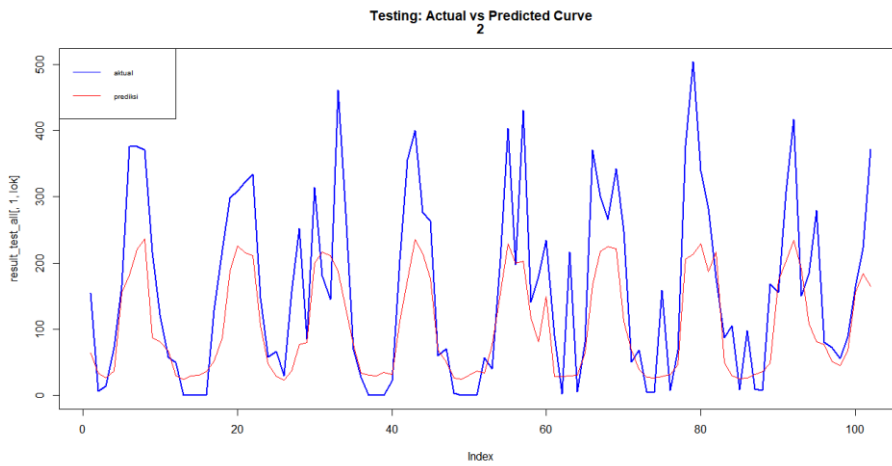
No	Seleksi Fitur				Kinerja Model Forecasting						
	Metode	Trace MV	fitur	% Pred	AICc	RMSE		Korelasi		Indeks R-sq	
						Train	Test	Train	Test	Train	Test
1	DPCCA + PCSUF + RNN	0,703	30	6,4	2623,7	199,04	309,84	0,88	0,81	0,77	0,66
2	DPCCA + RNN	0,123	301	64,3	8588,4	213,91	320,80	0,86	0,71	0,73	0,51
3	DPCCA+PCF + RNN	0,543	38	8,1	2646,4	199,02	314,41	0,91	0,80	0,82	0,64
4	DPCCA+SUF + RNN	0,806	22	4,7	2614,4	211,13	279,10	0,86	0,85	0,73	0,72
5	DPCCA + pilih relevan + RNN	0,186	206	44	3934,8	217,77	317,50	0,83	0,72	0,70	0,52

Pada Gambar 4.19–4.21 disajikan hasil testing dengan model 4 (DPCCA-SUF-RNN) untuk curah hujan di lokasi Juanda, Perak 1 dan Perak 2. Dari gambar tersebut disimpulkan bahwa model ST-RNN dengan input fitur non-linear mampu melakukan peramalan terhadap data aktual dari curah hujan dengan akurasi yang cukup baik.

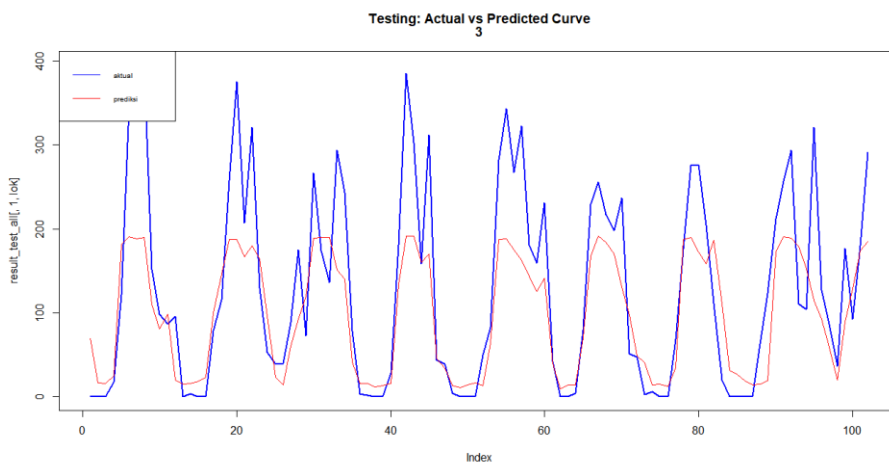
Model 4 tersebut memiliki nilai korelasi antara data aktual dan data peramalan masing-masing sebesar 0,813; 0,859 dan 0,864 atau secara rata-rata adalah 0,845 dengan indeks R^2 sebesar 72%. Salah satu kelemahan dari ST-RNN yang terlihat pada ketiga gambar tersebut adalah bahwa ST-RNN belum mampu mendeteksi curah hujan yang cukup ekstrim (nilai yang tinggi) sehingga memiliki nilai *error* yang besar pada kondisi tersebut.



Gambar 4.19 Peramalan dengan ST-RNN untuk curah hujan di Juanda



Gambar 4.20 Peramalan dengan ST-RNN untuk curah hujan di Perak 1



Gambar 4.21 Peramalan dengan ST-RNN untuk curah hujan di Perak 2

Dari hasil seleksi fitur dan pemodelan RNN pada Tabel 4.6, diperoleh bahwa metode DPCCA-SUF-RNN (model 4) memiliki kinerja terbaik dibandingkan 4 model lainnya berdasarkan nilai AICc, RMSE, korelasi dan R^2 pada tahap testing. Dapat disimpulkan bahwa kinerja RNN konsisten dengan kinerja seleksi fitur pada sub-bab 4.3.

4.6 Pemodelan Data Curah Hujan dengan *Hybrid* GSTAR-RNN

Sesuai dengan yang dijelaskan pada sub-bab 3.5, bahwa model penduga curah hujan untuk studi kasus data periode bulanan di Surabaya dengan menggunakan hybrid GSTAR-RNN adalah sebagai berikut:

1. Pemodelan GSTAR sehingga diperoleh *error* dari GSTAR. Model penduga curah hujan dengan GSTAR telah diperoleh sesuai yang dijelaskan pada sub-bab 4.4 di mana diperoleh model terbaik adalah GSTAR(12, 1) dengan perhitungan bobot menggunakan metode *uniform*. Model *hybrid* dilakukan karena hasil GSTAR untuk studi kasus 1 belum optimal (Tabel 4.5).
2. Ekstraksi dan seleksi fitur dengan metode DPCCA-PCSUF antara *error* GSTAR dan variabel prediktor dengan hasil yang ditunjukkan pada Tabel 4.7.

Tabel 4.7 Seleksi fitur untuk model hybrid GSTAR-RNN

No.	Metode	Threshold		Trace MV	Jumlah Prediktor	% Pred
		α	β			
1	DPCCA - PCSUF	-	-	-	-	-
2	DPCCA	-	-	0,223	215	45,9%
3	DPCCA-PCF	0,2	-	0,804	7	1,5%
4	DPCCA-SUF	-	-	-	-	-
5	DPCCA – pilih rrelevan	-	-	0,479	80	17,1%

Hasil seleksi fitur untuk nomor 1 (metode PCSUF) dan 4 (metode SUF) tidak diperoleh himpunan fitur prediktor karena tidak ada fitur non-linier yang berkorelasi dengan *error* dari GSTAR. Sedangkan untuk nomor 2 dan 5 tidak diperlukan nilai α dan β karena tidak ada pemrosesan untuk menghapus redundansi fitur prediktor.

3. Berdasarkan hasil dari sub-bab 4.3 dan 4.5 bahwa *trace-MV* yang tertinggi dengan jumlah fitur prediktor yang minimal (% Pred) akan menghasilkan kinerja RNN yang lebih baik. Oleh karenanya, pemodelan RNN dari *error*

GSTAR yang dilakukan adalah nomor 3, dengan $trace-MV = 0.804$ dan jumlah input minimal, yaitu 7 fitur (1,5% dari 468 fitur).

4. Dilakukan proses training dan testing untuk pemodelan RNN dari *error* GSTAR dengan persentase training 75% dari keseluruhan data (408 data), dengan arsitektur sesuai Gambar 4.22.

Hasil dari model RNN dari *error* GSTAR dan hybrid GSTAR-RNN ditunjukkan pada Tabel 4.8.

```
## create your model and add layers
set.seed(1)
model<-trainr(Y=Y_train, X=X_train,
              learningrate=0.1,hidden_dim=c(16, 12, 8, 4),
              numepochs=20000,
              momentum=0.9,
              seq_to_seq_unsync = F, update_rule = "sgd",
              network_type="rnn",
              epoch_function = c(epoch_print, epoch_annealing),
              loss_function = loss_L1, sigmoid = "logistic")
```

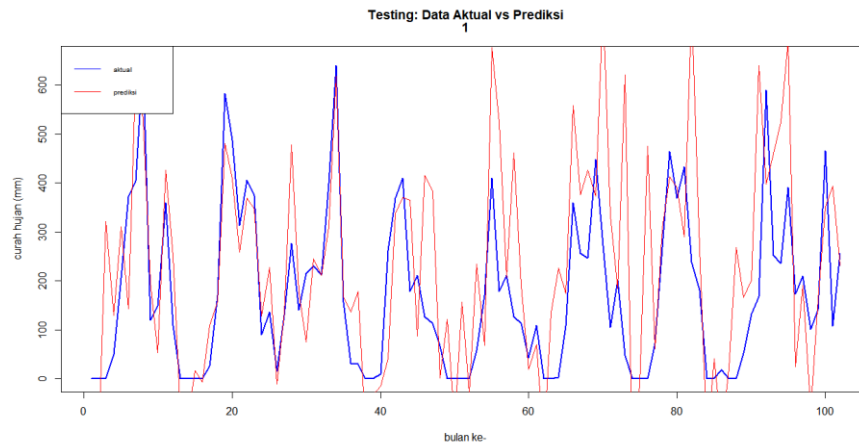
Gambar 4.22 Arsitektur RNN dari error GSTAR

Tabel 4.8 Kinerja model GSTAR dan hybrid GSTAR-RNN

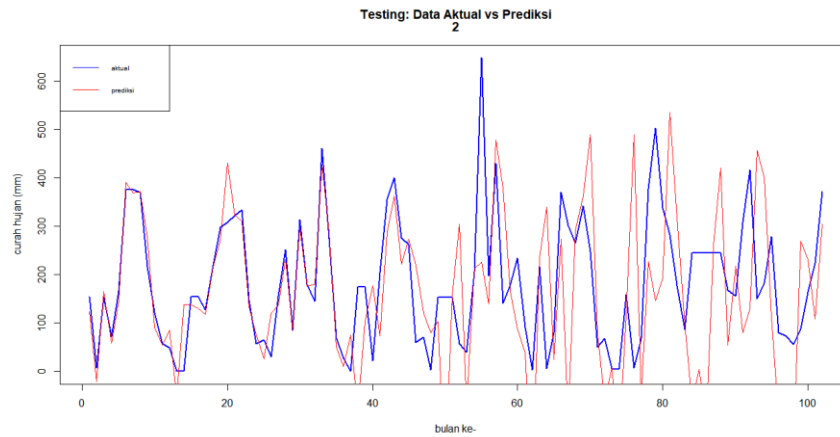
No	Seleksi Fitur				Kinerja Model Forecasting						
	Metode	Trace MV	fitur	% Pred	AICc	RMSE		Korelasi		Indeks R-sq	
						Train	Test	Train	Test	Train	Test
1	GSTAR	0,599	72	15,4	2964,48	350,98	468,46	0,65	0,51	0,43	0,28
2	RNN <i>error</i> GSTAR	0,804	10	2,1	1922,92	42,05	457,04	0,99	0,27	0,98	0,09
3	GSTAR-DPCCA-PCF-RNN	0,804	46	9,8	2588,31	122,54	382,02	0,99	0,58	0,99	0,34

Kinerja model hybrid (nomor 3, Tabel 4.8) tidak lebih baik dari model GSTAR (nomor 1, Tabel 4.8) maupun ST-RNN (nomor 4, Tabel 4.6). Hal ini menunjukkan bahwa model yang lebih kompleks tidak menjamin meningkatkan kinerja, sesuai dengan penelitian Makridakis dan Hibon [120] yang telah dijelaskan pada sub-bab 2.1.4. Penyebab lain adalah bahwa *error* dari hasil pemodelan GSTAR menunjukkan telah saling bebas sesuai analisis yang dijelaskan pada sub-bab 4.4 sehingga kinerja model ST-RNN dari *error* GSTAR sangat rendah (nomor 2, Tabel 4.8).

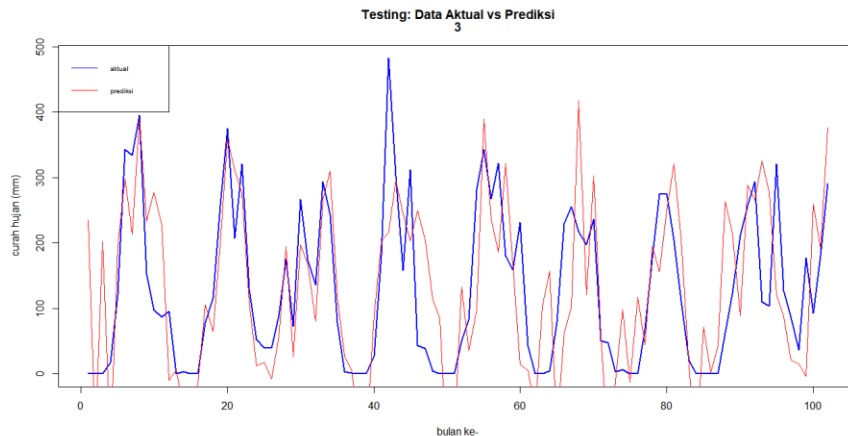
5. Penggabungan prediksi untuk GSTAR dan RNN (hybrid GSTAR-RNN) pada tahap testing/forecasting, ditunjukkan pada Gambar 4.23 – 4.25. Dari gambar tersebut, secara umum nilai peramalan model GSTAR-RNN tidak mendekati nilai aktualnya. Model GSTAR dengan nilai korelasi 0,73 dan indeks R^2 untuk peramalan adalah 54%, masih lebih baik dibandingkan dengan model *hybrid*.



Gambar 4.23 Peramalan hybrid GSTAR-RNN untuk curah hujan di Juanda



Gambar 4.24 Peramalan hybrid GSTAR-RNN untuk curah hujan di Perak 1



Gambar 4.25 Peramalan hybrid GSTAR-RNN untuk curah hujan di Perak 2

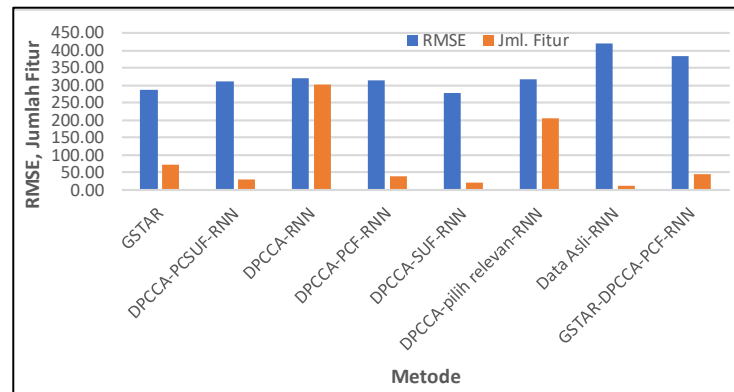
Pada Tabel 4.9 disajikan perbandingan kinerja seleksi fitur dan model penduga curah hujan untuk studi kasus periode bulanan di Surabaya. Pada model RNN tanpa dilakukan ekstraksi dan seleksi fitur (nomor 7), diperoleh kinerja yang rendah dengan jumlah prediktor yang kecil. Hal ini menunjukkan bahwa proses ekstraksi diperlukan untuk memperoleh himpunan prediktor yang menghasilkan kinerja lebih baik, yaitu model DPCCA-RNN. Untuk mengoptimalkan model penduga curah hujan dengan ST-RNN, diperlukan seleksi fitur prediktor (model nomer 2, 4, 5) yang memiliki kinerja lebih baik daripada model nomor 3, 6 dan 7.

Tabel 4.9 Kinerja model GSTAR, RNN dan hybrid GSTAR-RNN

No	Seleksi Fitur				AICc	Kinerja Model Forecasting					
	Metode	trace MV	fitur	Pred (%)		RMSE		Korelasi		Indeks R-sq	
						Train	Test	Train	Test	Train	Test
1	GSTAR	0,599	72	15,4	2964,48	350,98	468,46	0,65	0,51	0,43	0,28
2	DPCCA - PCSUF - RNN	0,703	30	6,4	2623,7	199,04	309,84	0,88	0,81	0,77	0,66
3	DPCCA - RNN	0,123	301	64,3	8588,4	213,91	320,80	0,86	0,71	0,73	0,51
4	DPCCA-PCF - RNN	0,543	38	8,1	2646,4	199,02	314,41	0,91	0,80	0,82	0,64
5	DPCCA-SUF - RNN	0,806	22	4,7	2614,4	211,13	279,10	0,86	0,85	0,73	0,72
6	DPCCA - pilih relevan - RNN	0,186	206	44	3934,8	217,77	317,50	0,83	0,72	0,70	0,52
7	Data Asli-RNN (Temporal)	-	12	2,6	2835,0	295,06	419,03	0,55	0,11	0,38	0,13
8	GSTAR-DPCCA-PCF-RNN	0,804	46	9,8	2588,31	122,54	382,02	0,99	0,58	0,99	0,34

Model terbaik adalah melakukan ekstraksi fitur dengan DPPCA dilanjutkan dengan seleksi fitur menggunakan metode SUF dan dengan model penduga RNN. Dapat disimpulkan bahwa RNN dengan input fitur (prediktor) non linier

menghasilkan kinerja lebih baik dibandingkan dengan prediktor linier, baik GSTAR (nomor 1) maupun PCF-RNN (nomor 4). Evaluasi model berdasarkan jumlah fitur dan nilai RMSE testing, disajikan pada Gambar 4.26.



Gambar 4.26 Nilai RMSE dan Jumlah Fitur berdasarkan Metode

4.7 Model Penduga Curah Hujan bulanan Wilayah Malang-Pasuruan

Berikut dijelaskan proses pemodelan GSTAR ST-RNN untuk studi kasus 2.

4.7.1 Model GSTAR untuk studi kasus 2

Pemodelan GSTAR mengikuti tahap/proses sesuai yang dijelaskan pada sub-bab 3.3.

1. Menguji korelasi curah hujan bulanan antar pasangan lokasi untuk pemenuhan syarat pemodelan *spatio-temporal*. Hasil pengujian nilai korelasi pada Gambar 4.27 menunjukkan bahwa data curah hujan pada 3 lokasi saling berhubungan.
2. Menguji sifat stasioner dari data curah hujan di 3 lokasi stasiun cuaca dengan menggunakan *ADF.test*. Dari uji ini, 3 data runtun waktu masih bersifat non-stasioner sesuai yang ditunjukkan pada Gambar 4.28. Oleh karenanya diperlukan proses *differencing* musiman dengan $d=1$ dan panjang musiman=12. Hasil dari proses ini menunjukkan bahwa data telah memenuhi pola stasioner.

```

data: Malang[, 1] and Malang[, 2]
t = 20.704, df = 354, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6891336 0.7837198
sample estimates:
      cor
0.7400646

data: Malang[, 1] and Malang[, 3]
t = 17.483, df = 354, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6206796 0.7327992
sample estimates:
      cor
0.6807059

Pearson's product-moment correlation

data: Malang[, 2] and Malang[, 3]
t = 22.223, df = 354, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.7160668 0.8034118
sample estimates:
      cor
0.7632038

```

Gambar 4.27 Korelasi data curah hujan pada 3 lokasi

```

Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -2.8579, Lag order = 24, p-value = 0.2148
alternative hypothesis: stationary

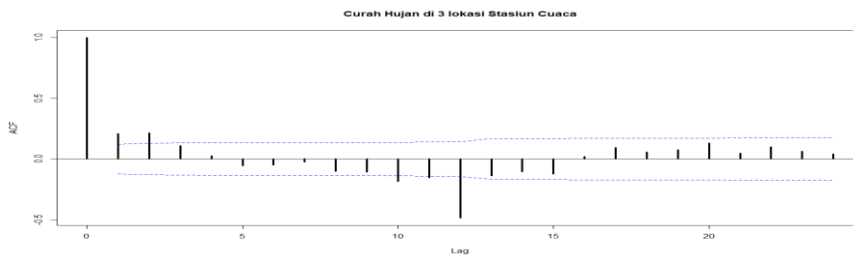
Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -2.8056, Lag order = 24, p-value = 0.2369
alternative hypothesis: stationary

Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -3.2033, Lag order = 24, p-value = 0.08761
alternative hypothesis: stationary

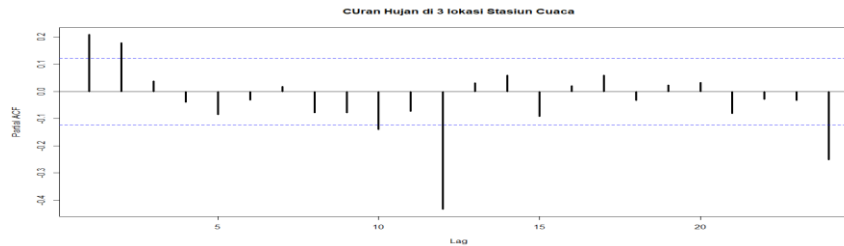
```

Gambar 4.28 Hasil uji stasioner untuk studi kasus 2

- Identifikasi awal untuk menentukan order p dan λ_k dengan menggunakan *correlogram* PACF dari data curah hujan pada lokasi 1 (Karang kates), sesuai Gambar 4.29. Dari ACF dan PACF tersebut, model yang diimplementasikan adalah GSTAR (24, 1).



(a) ACF data Karangkates



(b) PACF data Karangploso

Gambar 4.29 Correlogram ACF dan PACF studi kasus 2

- Menentukan nilai dari matriks bobot dengan metode normalisasi korelasi, sesuai Gambar 4.30.

```
## Perhitungan weight berdasar normalisasi korelasi
r12 <- 0.740
r13 <- 0.681
r23 <- 0.763

W <- matrix(c(0,r12/(r12+r13), r13/(r12+r13), r12/(r12+r13), 0, r23/(r12+r23),
r13/(r13+r12), r23/(r12+r23),0),3,3)
```

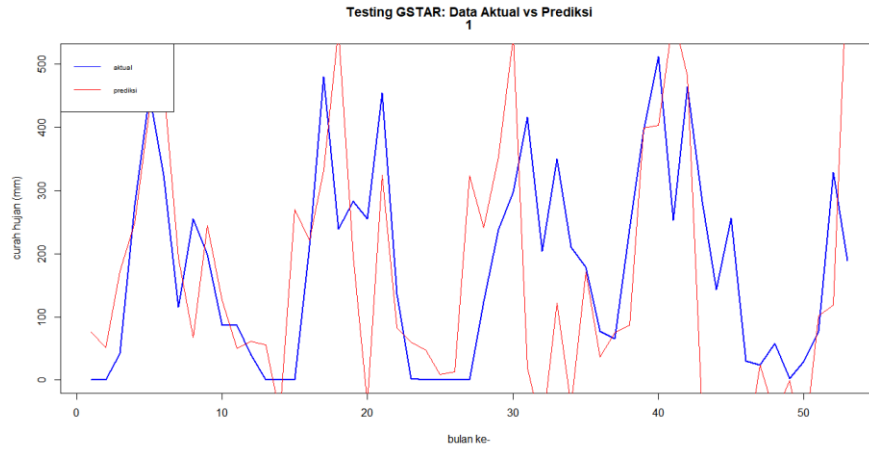
Gambar 4.30 Perhitungan nilai bobot dengan 2 metode dan kinerja

- Estimasi parameter model. Model GSTAR dihasilkan dari 75% dari total jumlah data. Jumlah data studi kasus 2 adalah 356 bulan.
- Mengukur kinerja model GSTAR dengan hasil sesuai Tabel 4.9.

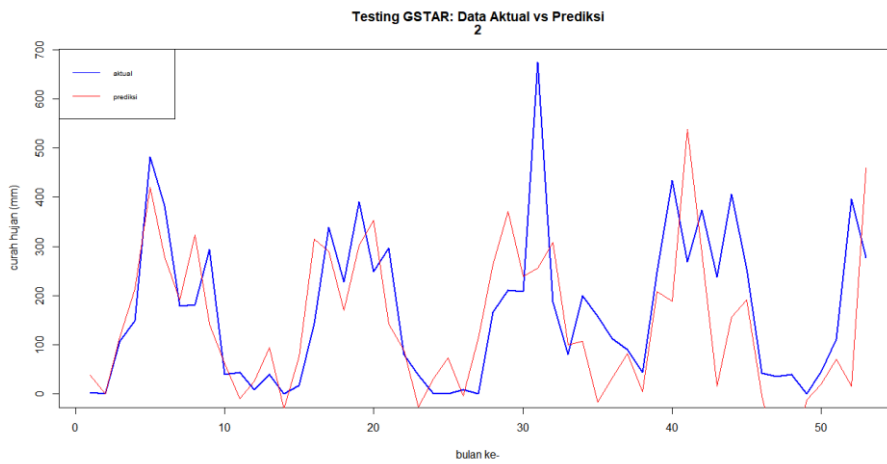
Tabel 4.10 Kinerja model GSTAR untuk studi kasus 2

Seleksi Fitur					Kinerja Model Forecasting					
Metode	trace MV	fitur	Pred (%)	AICc	RMSE		Korelasi		Indeks R-sq	
					Train	Test	Train	Test	Train	Test
GSTAR	0,312	36	7,7	2845,2	570,61	545,19	0,529	0,556	0,286	0,312

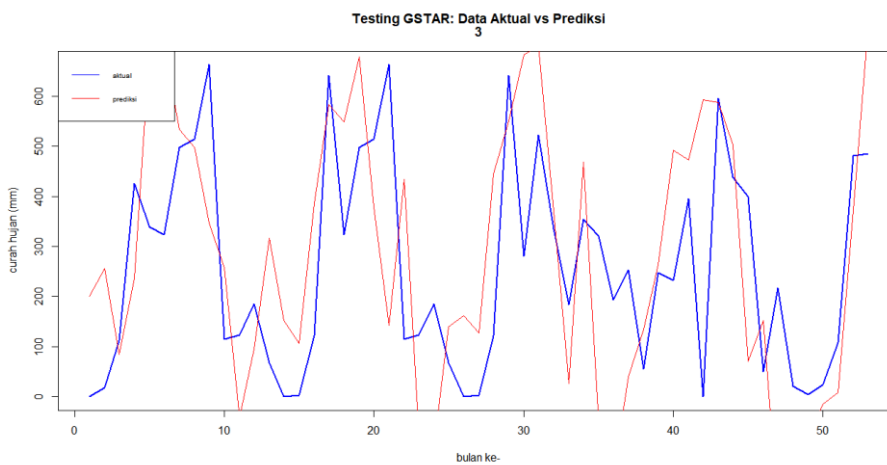
Kinerja testing untuk model GSTAR dari studi kasus 2 ditunjukkan pada Gambar 4.31 – Gambar 4.33.



Gambar 4.31 Plot data aktual-testing curah hujan di Karangates



Gambar 4.32 Plot data aktual-training curah hujan di Karangploso



Gambar 4.33 Plot data aktual-training curah hujan di Tretes

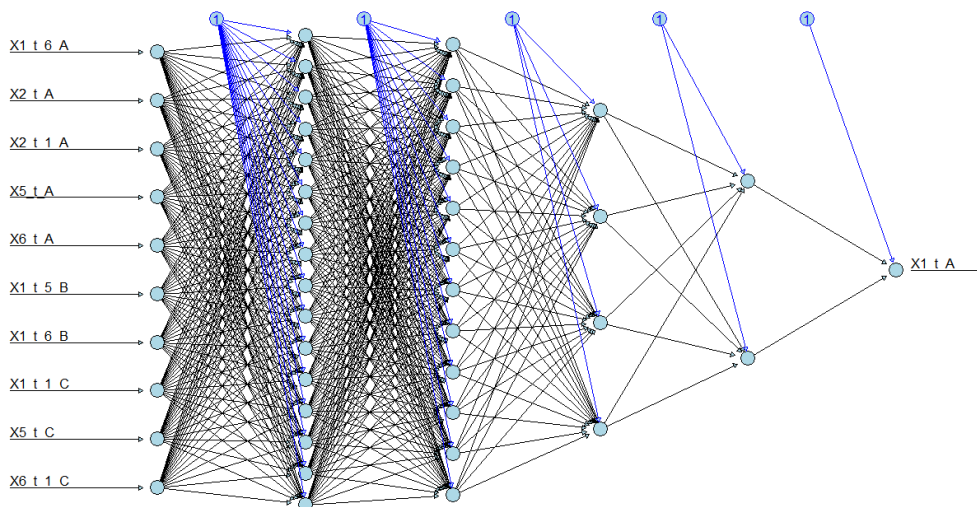
4.7.2 Model ST-RNN untuk studi kasus 2

Tahap pemodelan ST-RNN untuk studi kasus 2 sesuai yang dijelaskan pada sub-bab 3.4 adalah sebagai berikut:

1. Melakukan ekstraksi fitur dengan metode DPCCA dan seleksi fitur dengan metode PCF, SUF maupun PCSUF untuk memperoleh input optimal dari RNN.
2. Data curah hujan pada 3 lokasi, yaitu Karangates, Karangploso dan Tretes menjadi nilai target dari RNN.
3. Persentase data training ditetapkan 75% sehingga untuk proses testing menggunakan 25% dari total data sejumlah 356 periode bulanan.
4. Melakukan normalisasi data prediktor dan curah hujan untuk input dari pemodelan ST-RNN.
5. Menetapkan arsitektur dari jaringan RNN sebagaimana pada Gambar 4.34 dengan visualisasi sesuai Gambar 4.35.

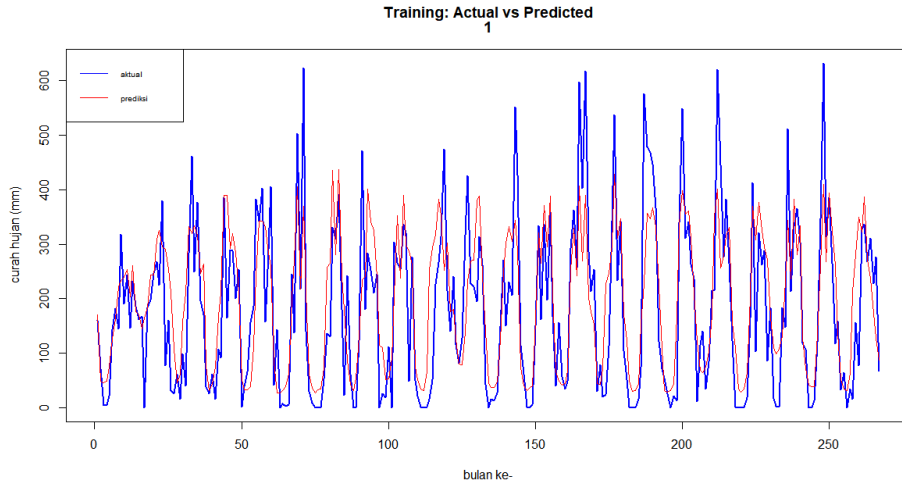
```
set.seed(1)
model<-trainr(Y=Y_train, X=X_train,
  learningrate=0.1, hidden_dim=c(16, 12, 8, 4),
  numepochs=875,
  momentum=0.9, use_bias = F, learningrate_decay = 1,
  seq_to_seq_unsync = F, update_rule = "sgd",
  network_type="rnn",
  epoch_function = c(epoch_print, epoch_annealing), loss_function = loss_L1,
  sigmoid = c("logistic", "Gompertz", "tanh"))
```

Gambar 4.34 Arsitektur RNN untuk studi kasus 2



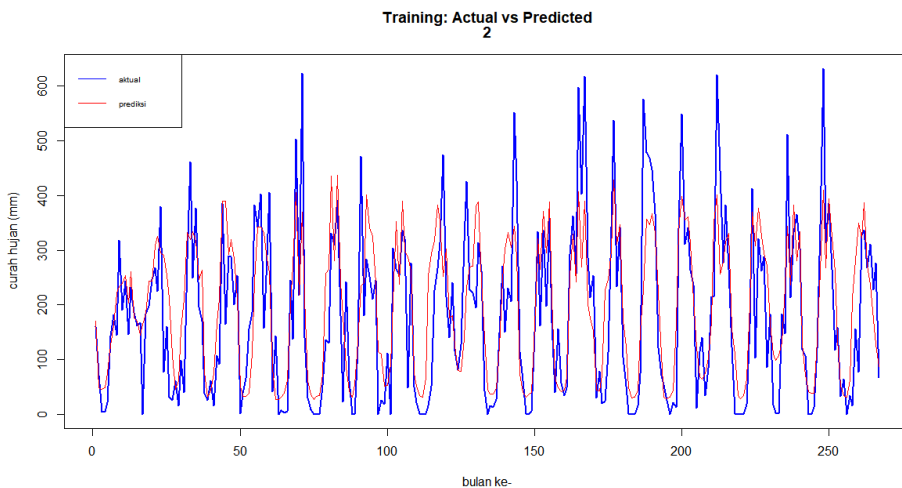
Gambar 4.35 Arsitektur RNN untuk prediksi curah hujan di Karangates

6. Melakukan proses training dan mengukur kinerjanya. Pada Gambar 4.36 – Gambar 4.38 disajikan hasil proses training model DPCCA-SUF-RNN untuk curah hujan di wilayah Karangates, Karangploso dan Tretes.

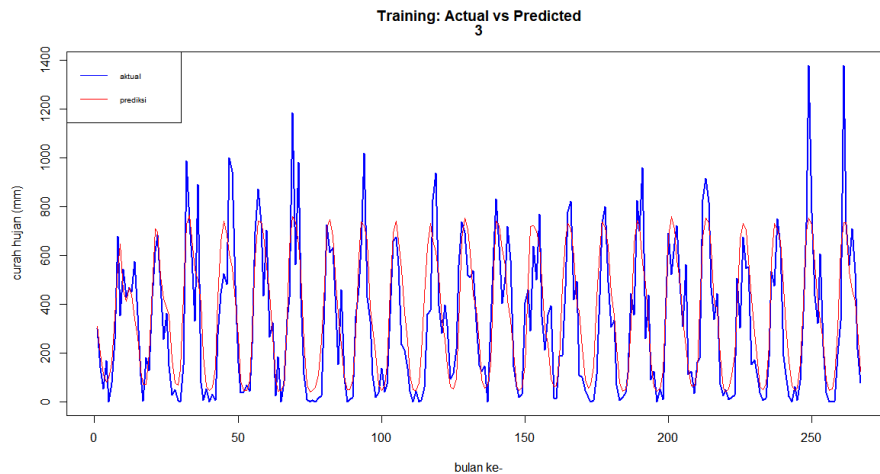


Gambar 4.36 Hasil training untuk curah hujan di Karangates

Dari gambar hasil training untuk ketiga lokasi, memiliki kesesuaian antara nilai aktual dengan nilai prediksi dengan nilai korelasi rata-rata antara aktual dan prediksi adalah 0,806 atau nilai indeks R^2 adalah 65%.

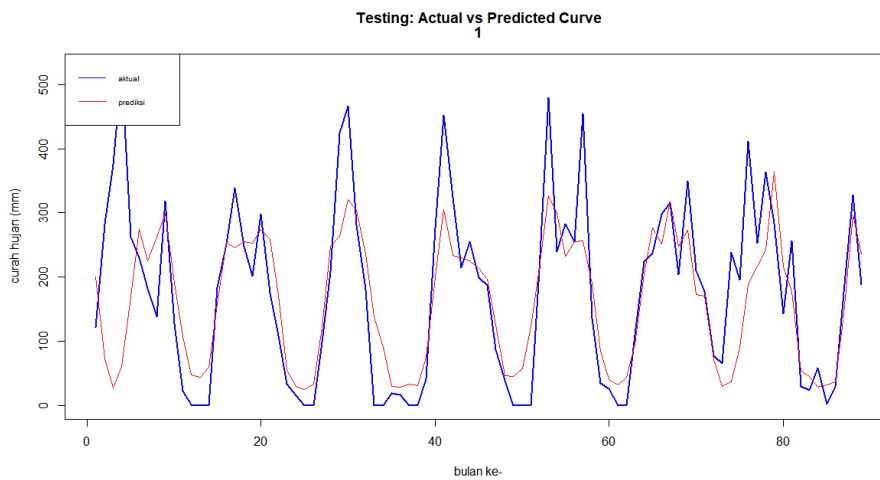


Gambar 4.37 Hasil proses training curah hujan di Karangploso

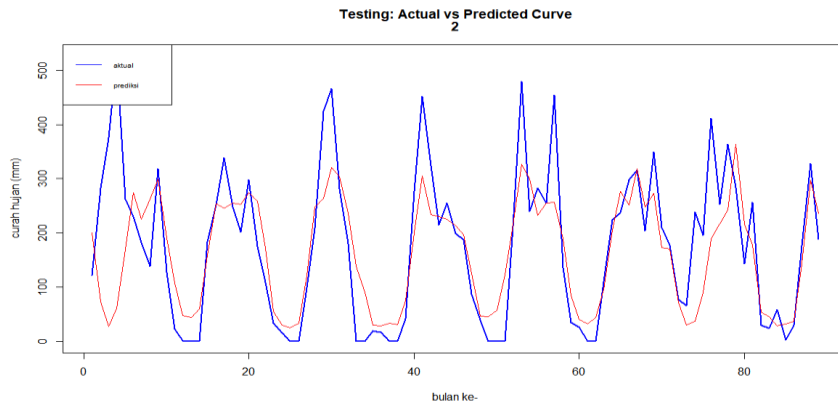


Gambar 4.38 Hasil proses training curah hujan di Tretes

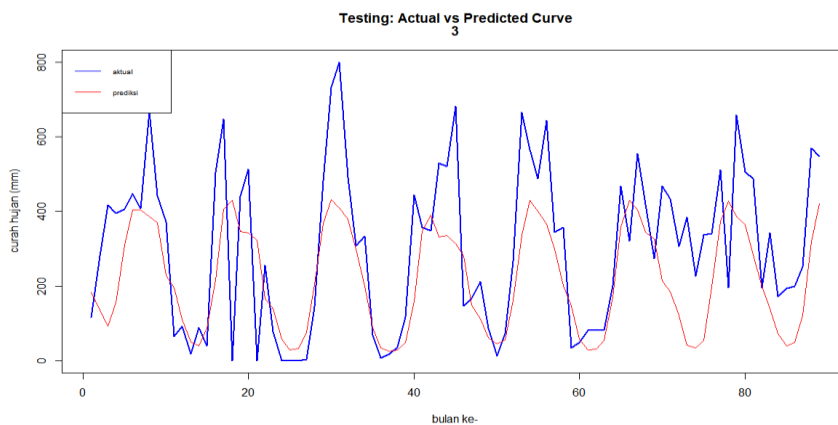
- Melakukan proses testing dan mengukur kinerja model dari studi kasus 2 dengan hasil sesuai yang ditunjukkan pada Tabel 4.11. Sedangkan kesesuaian antara data aktual dengan peramalan, disajikan pada Gambar 4.39 – Gambar 4.41.



Gambar 4.39 Hasil peramalan curah hujan di Karangates



Gambar 4.40 Hasil peramalan curah hujan di Karangploso



Gambar 4.41 Hasil peramalan curah hujan di Tretes

Tabel 4.11 Kinerja model untuk studi kasus 2

No	Seleksi Fitur				AICc	Kinerja Model Forecasting					
	Metode	Trace MV	fitur	Pred (%)		RMSE		Korelasi		Indeks R-sq	
						Train	Test	Train	Test	Train	Test
1	GSTAR	0,312	36	7,7	2845,20	570,61	545,19	0,53	0,56	0,29	0,31
2	DPCCA - PCSUF - RNN	0,498	18	3,8	3180,93	165,23	212,605	0,85	0,76	0,72	0,59
3	DPCCA - RNN	0,058	252	54	5015,74	212,93	276,780	0,49	0,40	0,45	0,31
4	DPCCA-PCF - RNN	0,454	15	3,2	3251,78	189,05	218,548	0,80	0,74	0,65	0,55
5	DPCCA-SUF - RNN	1,178	10	2,0	2536,54	181,76	174,790	0,86	0,87	0,75	0,78
6	DPCCA - pilih relevan - RNN	0,131	96	20	3275,68	135,84	235,811	0,91	0,71	0,83	0,50
7	Data Asli-RNN (Temporal)	-	12	2,6	3248,32	187,30	298,054	0,80	0,45	0,63	0,22
8	GSTAR-DPCCA-PCF-RNN	1,531	25	5,3	883,68	9,393	346,625	0,99	0,608	0,99	0,37

Dari Tabel 4.11 untuk proses seleksi fitur terbaik adalah DPCCA-SUF (nomor 5) yang menghasilkan *trace* MV terbaik ($\pi_{i,j} = 1,178$) dengan himpunan fitur terkecil, yaitu 2% dari total hasil ekstraksi 312 fitur prediktor. Dari pemodelan

dengan ST-RNN, kinerja model 5 pada tahap testing/peramalan adalah terbaik berdasarkan nilai RMSE, korelasi dan akurasi dengan indeks $R^2 = 78\%$ sehingga kinerja antara seleksi fitur dan ST-RNN adalah konsisten.

Secara umum, model ST-RNN dengan seleksi fitur (nomor 2, 4, 5) memiliki kinerja lebih baik dibandingkan tanpa seleksi fitur (nomor 3, 6), model temporal (nomer 7) maupun GSTAR (nomer 1). Demikian juga untuk model hybrid GSTAR-RNN (nomer 8) tidak berhasil meningkatkan kinerja model GSTAR maupun RNN. Seperti halnya pada studi kasus 1, pemodelan ST-RNN untuk studi kasus 2 masih memiliki kelemahan bahwa belum mampu mendeteksi curah hujan yang cukup ekstrim.

4.8 Model Penduga Curah Hujan periode harian di Wilayah Surabaya

4.8.1 Model GSTAR untuk studi kasus 3

1. Menguji korelasi data curah hujan periode harian pada 3 lokasi, yaitu Juanda, Perak 1 dan Perak 2.

Pada Gambar 4.42 disajikan hasil uji korelasi curah hujan. Nilai tersebut menunjukkan bahwa korelasi data curah hujan harian di 3 lokasi adalah signifikan sehingga memenuhi syarat untuk pemodelan *spatio-temporal*.

```
Pearson's product-moment correlation
data: data1[, 1] and data2[, 1]
t = 7.9427, df = 406, p-value = 1.962e-14
sample estimates:
cor
0.3667273

data: data1[, 1] and data3[, 1]
t = 6.601, df = 406, p-value = 1.28e-10
sample estimates:
cor
0.3113209

data: data2[, 1] and data3[, 1]
t = 18.648, df = 406, p-value < 2.2e-16
sample estimates:
cor
0.6792412
```

Gambar 4.42 Korelasi curah hujan di 3 lokasi stasiun cuaca

2. Melakukan uji stasioneritas data curah hujan di 3 lokasi.

Gambar 4.43 menunjukkan hasil pengujian stasioneritas menggunakan uji ADF yang menunjukkan bahwa data runtun waktu curah hujan di 3 lokasi tersebut adalah non-stasioner sehingga memerlukan proses *differencing* musiman.

```

Augmented Dickey-Fuller Test
data: Data_Asli[, 1]
Dickey-Fuller = -2.2136, Lag order = 365, p-value = 0.4879
alternative hypothesis: stationary

Augmented Dickey-Fuller Test
data: Data_Asli[, 2]
Dickey-Fuller = -2.1012, Lag order = 365, p-value = 0.5355
alternative hypothesis: stationary

Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -1.8928, Lag order = 365, p-value = 0.6237
alternative hypothesis: stationary

```

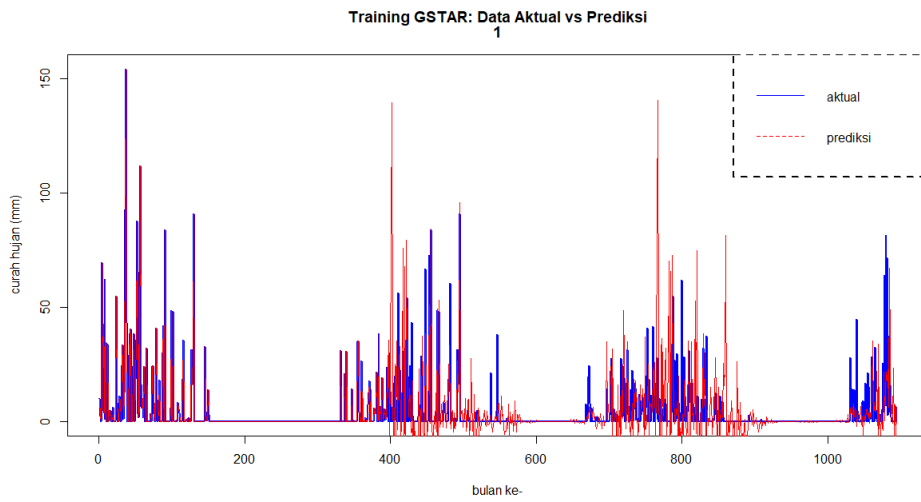
Gambar 4.43 Pengujian stasioneritas data curah hujan

- Melakukan pemodelan GSTAR dengan 60% adalah data training dan koefisien bobot dihitung menggunakan nilai *uniform*, inversi jarak dan korelasi.

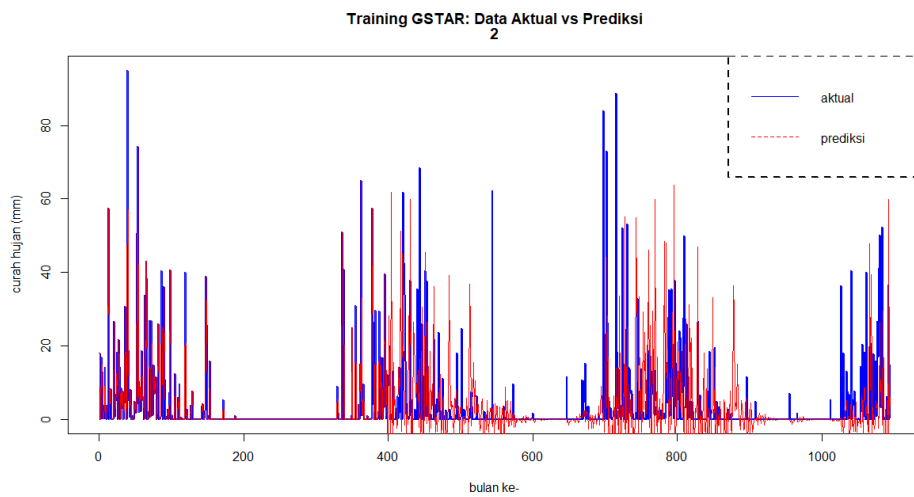
Pada Tabel 4.12 disajikan kinerja model GSTAR dengan menggunakan 3 metode perhitungan bobot setelah data dilakukan proses *differencing*. Model terbaik untuk studi kasus 3 dengan pendekatan GSTAR adalah (1, 30) dengan jumlah fitur prediktor adalah 90 (19,2%) dan *trace-MV*=0,199. Hasil training dan testing dengan perhitungan nilai bobot menggunakan inversi jarak (nomor 2, Tabel 4.12) disajikan pada Gambar 4.44–Gambar 4.49. Perhitungan bobot menggunakan inversi jarak memiliki nilai *AIC-correction* terbaik, yaitu 5428,49. Dari tabel dan gambar tersebut menunjukkan bahwa kinerja model GSTAR untuk studi kasus 3 masih cukup rendah, yaitu dengan indeks R^2 training < 40%.

Tabel 4.12 Kinerja model GSTAR untuk studi kasus 3

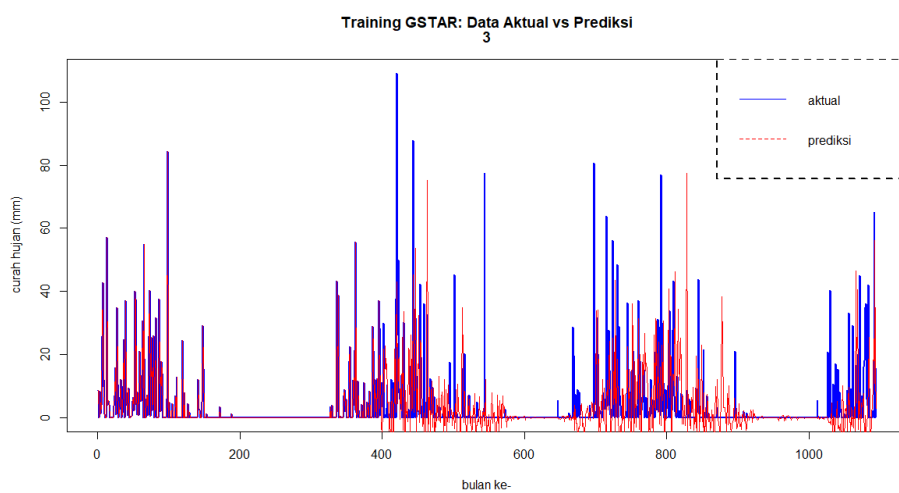
No.	Metode bobot	RMSE		R ²		Korelasi		% Training
		Model	Forecast	Model	Forecast	Model	Forecast	
1	uniform	34,97	59,61	0,302	0,016	0,547	0,125	60
2	jarak	34,95	60,86	0,304	0,012	0,549	0,103	60
3	korelasi	34,95	59,60	0,303	0,015	0,548	0,118	60



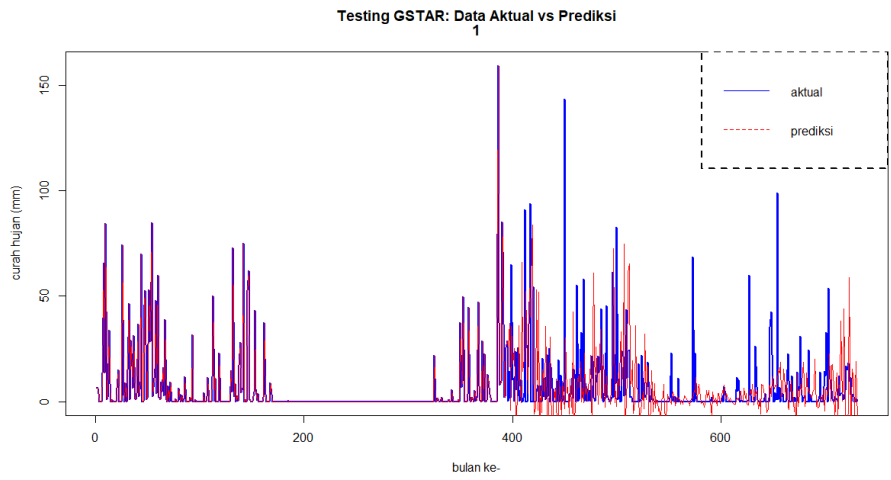
Gambar 4.44 Hasil training studi kasus 3 – data Juanda



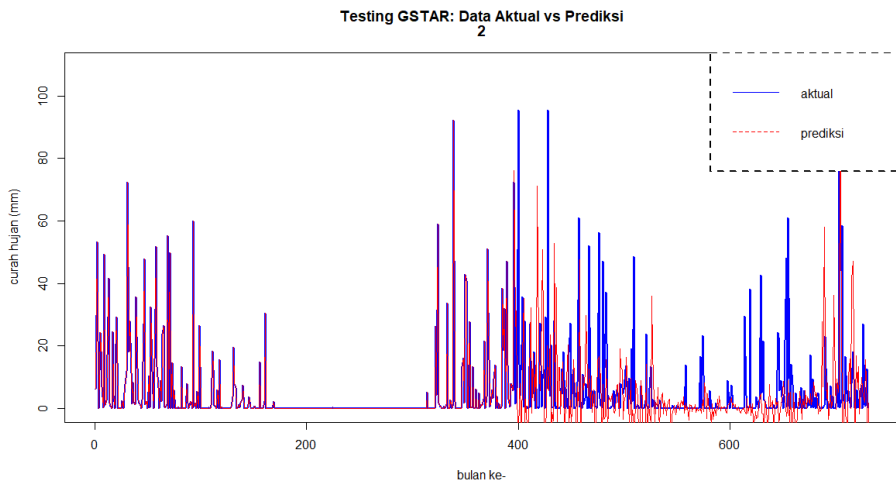
Gambar 4.45 Hasil training studi kasus 3 – data Perak 1



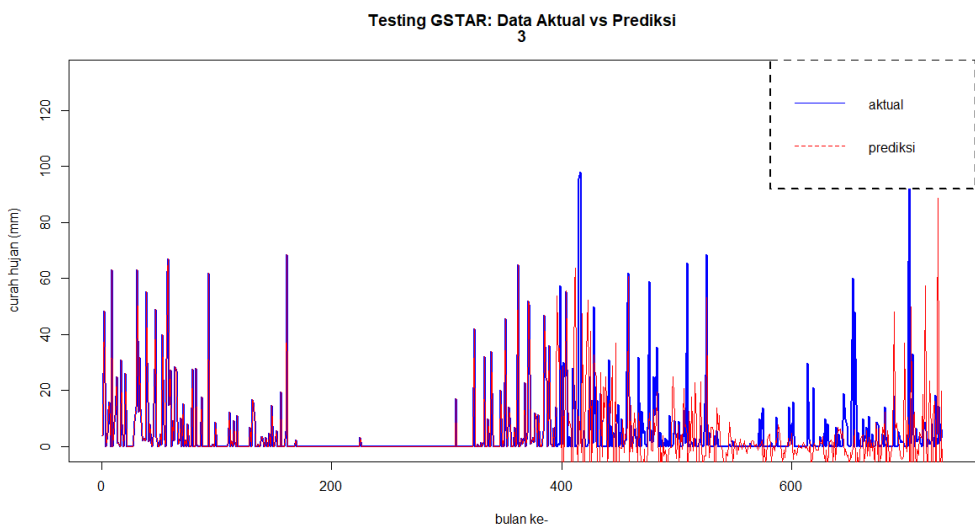
Gambar 4.46 Hasil training studi kasus 3 – data Perak 2



Gambar 4.47 Hasil testing studi kasus 3 – data Juanda



Gambar 4.48 Hasil testing studi kasus 3 – data Perak 1



Gambar 4.49 Hasil testing studi kasus 3 – data Perak 2

4.8.2 Model ST-RNN untuk studi kasus 3

Tahap pemodelan ST-RNN untuk studi kasus 3 adalah sebagai berikut:

1. Melakukan ekstraksi dan seleksi fitur untuk memperoleh input optimal dari ST-RNN. Pada Tabel 4.13 telah diperoleh 20 fitur input dari hasil seleksi fitur DPCCA-SUF dengan *trace-MV* yang terbaik, yaitu 0,529. Hal ini berarti bahwa untuk data periode harian, fitur input prediktor merupakan fitur non-linier sebagai penduga curah hujan.

Tabel 4.13 Kinerja model ST-RNN untuk studi kasus 3

No	Seleksi Fitur				Kinerja Model Forecasting						
	Metode	Trace MV	fitur	% Pred	AICc	RMSE		Korelasi		Indeks R-sq	
						Train	Test	Train	Test	Train	Test
1	DPCCA-PCSUF-RNN	0,337	-	-	-	-	-	-	-	-	-
2	DPCCA-PCF-RNN	0,310	-	-	-	-	-	-	-	-	-
3	DPCCA-SUF-RNN	0,529	20	4,3	2179,71	8,84	17,49	0,97	0,90	0,94	0,81

Selanjutnya, input yang digunakan pada pemodelan ST-RNN adalah dari hasil DPCCA-SUF (Tabel 4.13 nomor 3) dikarenakan nomor 1 dan 2 memiliki nilai *trace-MV* yang lebih kecil.

2. Data curah hujan harian pada 3 lokasi, yaitu Juanda, Perak 1 dan Perak 2 menjadi nilai target dari ST-RNN dengan input 19 fitur yang terdiri atas curah hujan (X_1), kelembaban relatif (X_5), kecepatan angin rata-rata (X_7) pada lags ($t-k$) dan pengaruh spasial (lokasi).
3. Persentase data training ditetapkan 60% sehingga untuk proses testing menggunakan 40% dari total data 1.826 periode harian.
4. Menetapkan arsitektur dari jaringan RNN sebagaimana pada Gambar 4.50.

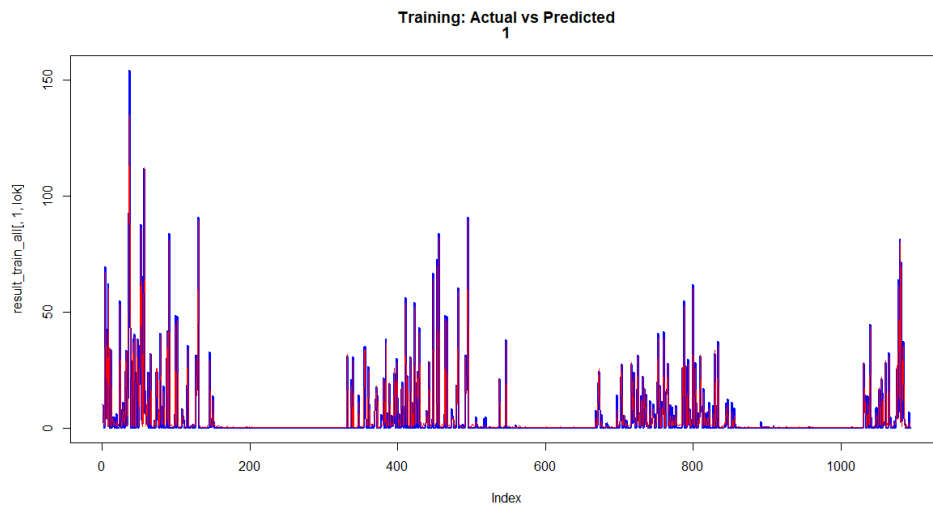
```

set.seed(1)
model<-trainr(Y=Y_train, X=X_train,
  learningrate=0.01, hidden_dim=c(70, 60, 50, 4),
  numepochs=7500,
  momentum=0.9, use_bias = F, learningrate_decay = 1,
  seq_to_seq_unsync = F, update_rule = "sgd",
  network_type="rnn",
  epoch_function = c(epoch_print, epoch_annealing), loss_function =
  loss_L1, sigmoid = "logistic")

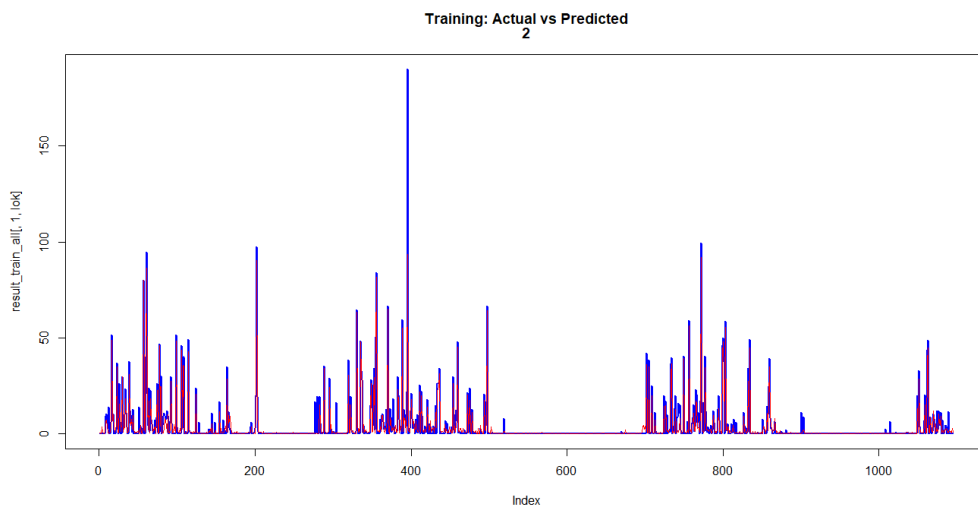
```

Gambar 4.50 Arsitektur RNN untuk studi kasus 3

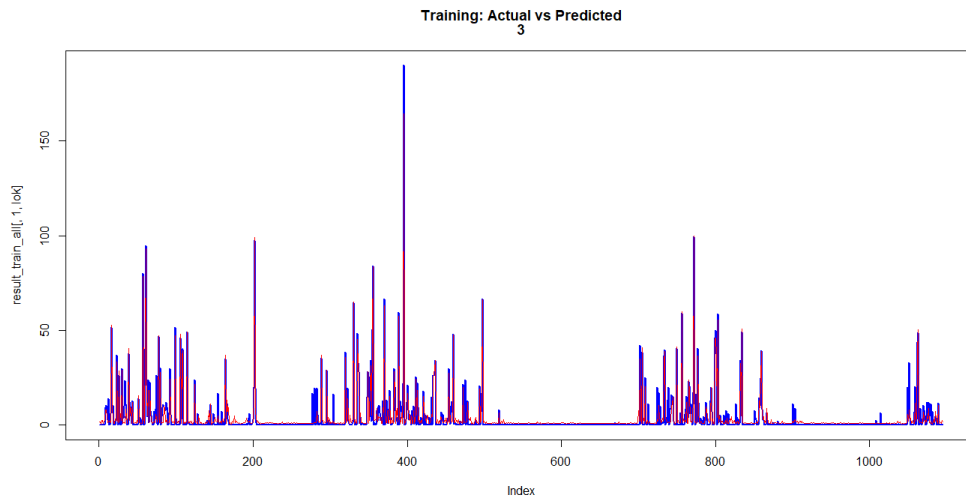
5. Melakukan proses training dengan hasil training sesuai yang ditunjukkan pada Gambar 4.51 – Gambar 4.53. Hasil training pada Tabel 4.13 menunjukkan bahwa ST-RNN memiliki kinerja yang cukup bagus, yaitu korelasi rata-rata bernilai 0,97 dan indeks R^2 bernilai 94%.



Gambar 4.51 Hasil training metode ST-RNN untuk data Juanda

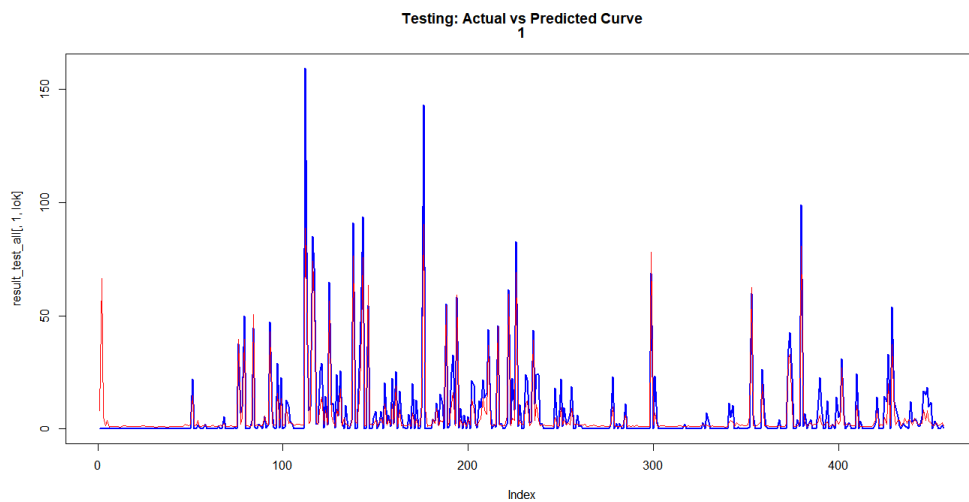


Gambar 4.52 Hasil training metode ST-RNN untuk data Perak1

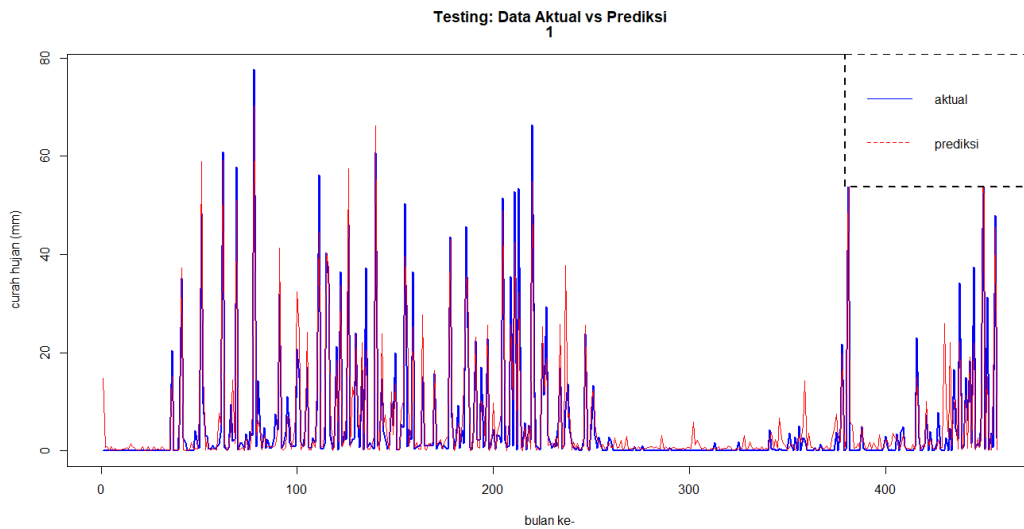


Gambar 4.53 Hasil training metode ST-RNN untuk data Perak 2

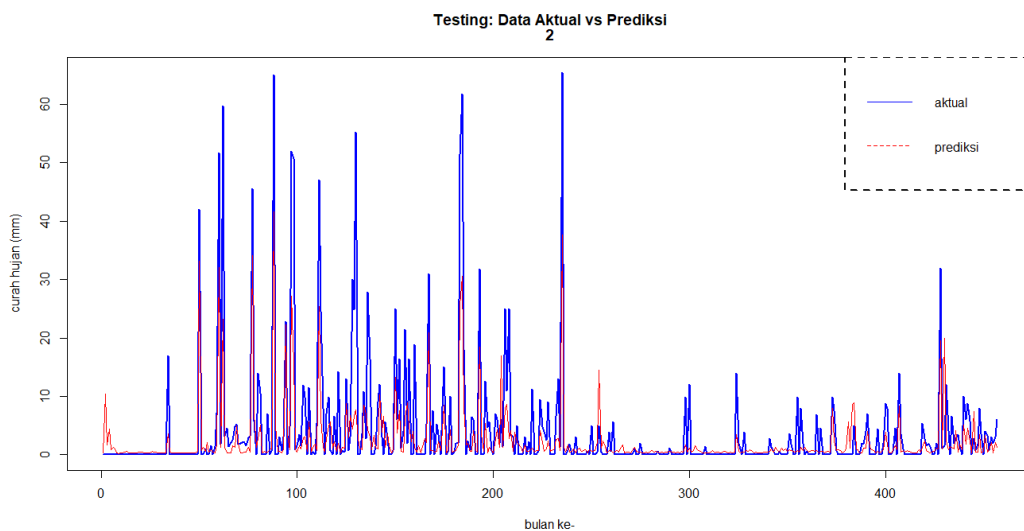
6. Melakukan proses testing dan mengukur kinerja model dari studi kasus 3 dengan hasil sesuai yang ditunjukkan pada Tabel 4.13. Sedangkan kesesuaian antara data aktual dengan peramalan, disajikan pada Gambar 4.54-Gambar 4.56. Hasil testing untuk studi kasus 3 memiliki kinerja yang cukup baik, yaitu dengan korelasi 0,9 dan indeks R^2 bernilai 81%.



Gambar 4.54 Hasil testing metode ST-RNN untuk data Juanda



Gambar 4.55 Hasil testing metode ST-RNN untuk data Perak 1



Gambar 4.56 Hasil testing metode ST-RNN untuk data Perak 2

4.9 Model Penduga Curah Hujan periode harian Wilayah Malang-Pasuruan

1. Melakukan perhitungan korelasi data curah hujan harian di wilayah Malang-Pasuruan, dengan nilai $r_{x,y} < 0,3$ untuk pasangan 3 stasiun cuaca. Nilai ini dapat dianggap cukup rendah sehingga apabila dilakukan pemodelan *spatio-temporal*, misalnya dengan GSTAR, diperoleh kinerja model yang cukup rendah. Nilai korelasi untuk pasangan 3 stasiun cuaca ditunjukkan pada Gambar 4.57.
2. Menguji sifat stasioner dari data curah hujan di 3 lokasi stasiun cuaca dengan menggunakan *ADF.test*. Dari uji ini, ke tiga data runtun waktu masih bersifat

non-stasioner sesuai yang ditunjukkan pada Gambar 4.58. Oleh karenanya diperlukan proses *differencing* musiman dengan $d=1$ dan panjang musiman=12.

```
Pearson's product-moment correlation
data: Data_Asli[, 1] and Data_Asli[, 2]
t = 5.2013, df = 1125, p-value = 2.35e-07
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.09570257 0.20975783
sample estimates:
      cor
0.1532405

data: Data_Asli[, 1] and Data_Asli[, 3]
t = 3.044, df = 1125, p-value = 0.002389
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.03215856 0.14799635
sample estimates:
      cor
0.09038314

data: Data_Asli[, 2] and Data_Asli[, 3]
t = 9.2027, df = 1125, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.2094337 0.3180720
sample estimates:
      cor
0.2645921
```

Gambar 4.58 Korelasi curah hujan di wilayah Malang-Pasuruan

```
Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -1.0158, Lag order = 365, p-value = 0.9362
alternative hypothesis: stationary

Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -1.8003, Lag order = 365, p-value = 0.6629
alternative hypothesis: stationary

Augmented Dickey-Fuller Test
data: Data_Asli[, lok]
Dickey-Fuller = -3.6201, Lag order = 365, p-value = 0.08075
alternative hypothesis: stationary
```

Gambar 4.57 Pengujian stasioneritas data curah hujan studi kasus 4

3. Melakukan pemodelan GSTAR.

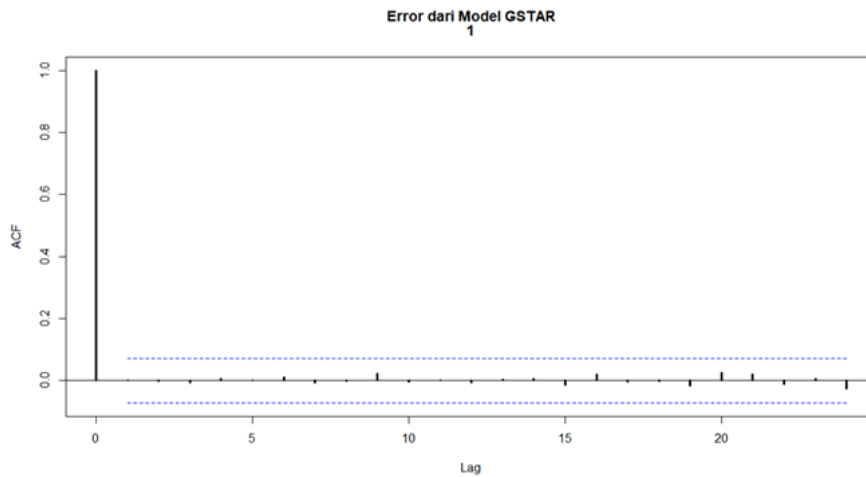
Apabila dilakukan pemodelan GSTAR, kinerja model cukup rendah sesuai yang disajikan pada Tabel 4.14 di mana nilai korelasi dan R^2 baik proses training maupun testing adalah sangat kecil.

Metode	fitur	% Pred	AICc	RMSE		Korelasi		Indeks R-sq	
				Train	Test	Train	Test	Train	Test
GSTAR	36	7,69	6306,3	112,8	71,17	0,15	0,03	0,01	0,001

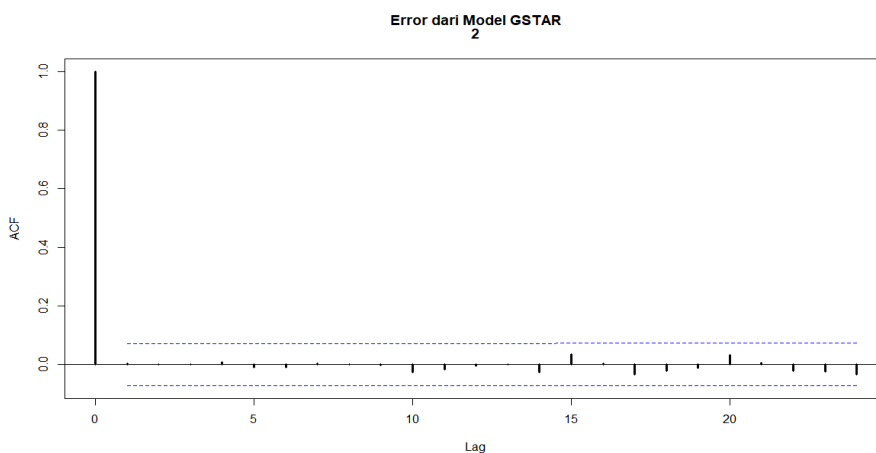
Tabel 4.14 Kinerja GSTAR untuk studi kasus 4

4. Analisis identik dan saling bebas (*independent*) terhadap *error* dari GSTAR.

Dari Gambar 4.59 – 4.61 menunjukkan bahwa ACF dari residual telah *white-*

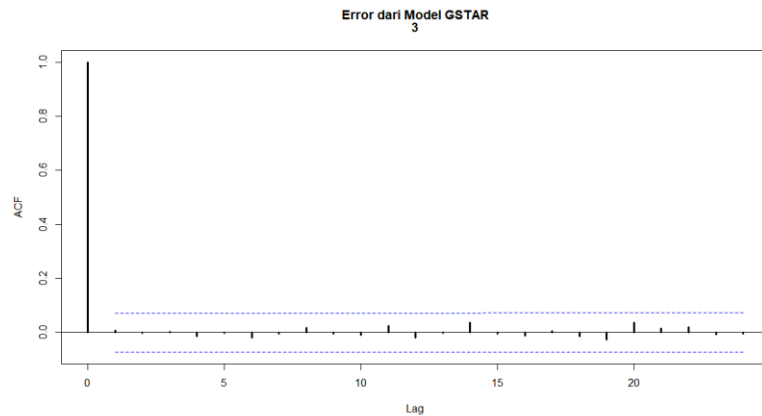


Gambar 4.59 ACF dari residual GSTAR wilayah Karangates



noise yang artinya *error* dari GSTAR telah saling bebas (*independent*).

Gambar 4.60 ACF dari residual GSTAR wilayah Karangploso



Gambar 4.61 ACF dari residual GSTAR wilayah Tretes

5. Melakukan pemodelan dengan ST-RNN

Apabila dibuat model penduga curah hujan harian dengan metode ST-RNN, terlebih dahulu harus dilakukan ekstraksi dan seleksi fitur input. Pada Tabel 4.15 telah diperoleh fitur input yang optimal dari hasil seleksi fitur DPCCA-PCF/SUF/PCSUF. Dari hasil ini menunjukkan bahwa untuk studi kasus 4, pemodelan ST-RNN kurang sesuai karena nilai *Trace-MV* cukup rendah ($<0,5$) sehingga kinerja model ST-RNN juga rendah.

Tabel 4.15 Seleksi fitur untuk studi kasus 4

No.	Ekstraksi-Seleksi	Trace-MV	Threshold		Fitur	% Pred
			α	β		
1	DPCCA-PCSUF	0,4356	0,2	0,2	37	11,9
2	DPCCA-PCF	0,2409	0,2	-	17	5,45
3	DPCCA-SUF	0,3450	-	0,2	19	5,09

Halaman ini sengaja dikosongkan

BAB 5 KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil penelitian ini telah berhasil membangun model penduga curah hujan di wilayah ZoM melalui kombinasi proses ekstraksi variabel prediktor, seleksi fitur, dan pengembangan model ST-RNN.

Dari hasil uji coba untuk 4 studi kasus, dapat disimpulkan hal-hal sebagai berikut:

1. Ekstraksi fitur prediktor menggunakan *spatio-temporal* DPCCA telah berhasil mengidentifikasi kandidat fitur yang memiliki korelasi dengan curah hujan baik berkaitan faktor waktu (temporal), lokasi (spasial) maupun variabel prediktor lokal/global.
2. Nilai *trace* dari *MV* berhasil sebagai ukuran kesesuaian optimalitas (*goodness of fit*) dari himpunan fitur hasil seleksi dengan pendekatan *spatio-temporal* di mana $trace\text{-}MV > 0,5$ akan memberikan hasil kinerja model yang cukup baik.
3. *Threshold* untuk penghapusan redudansi fitur linier/non-linier yang menyebabkan *trace MV* maksimal dicapai pada nilai $\alpha = 0,2$ dan $\beta = 0,2$. Hal ini menunjukkan bahwa semakin rendah batas nilai redudansi yang ditetapkan atas korelasi dan nilai SU, nilai *trace MV* semakin baik.
4. Hasil terbaik seleksi fitur pada studi kasus data bulanan dan harian adalah menggunakan DPCCA-SUF di mana fitur terpilih memiliki korelasi non-linier terhadap curah hujan.
5. Variabel indeks iklim sebagai prediktor global, terpilih sebagai kandidat fitur melalui proses DPCCA namun tidak cukup signifikan sebagai fitur prediktor pada seleksi dengan PCF, SUF, dan PCSUF.
6. Prediktor dominan yang terpilih baik menggunakan PCF, SUF maupun PCSUF adalah lama penyinaran (X_6), kelembaban relatif (X_5), dan temperatur (min/maks/mean) dengan *time-lags* tertentu ($t - k$) dari sejumlah lokasi.
7. Pada studi kasus data bulanan, DPCCA-SUF-RNN memiliki kinerja lebih baik dibandingkan model GSTAR, hybrid GSTAR-RNN, ARIMA maupun Fungsi Transfer Multi Input.

8. Pada studi kasus data harian, DPCCA-SUF-RNN memiliki kinerja lebih baik dibandingkan model GSTAR maupun GSTAR-RNN.
9. Jumlah fitur input yang banyak tidak menjamin bahwa kinerja model ST-RNN semakin baik. Dengan fitur dan arsitektur yang tepat, akan memberikan kinerja model yang cukup baik.

5.2 Saran

Saran berkaitan pengembangan model penduga curah hujan wilayah ZoM di Indonesia melalui kombinasi proses ekstraksi dan seleksi variabel prediktor serta *spatio-temporal* RNN adalah sebagai berikut:

1. Diperlukan uji coba data runtun waktu multivariat sintetik/simulasi yang memiliki dependensi *spatio-temporal* untuk memperkuat kesimpulan atas kinerja dari proses ekstraksi fitur, seleksi fitur dan pemodelan ST-RNN.
2. Diperlukan uji coba untuk proses peramalan $t + k$ periode ke depan (*multi-step ahead*), baik data periode harian maupun bulanan.
3. Penggunaan teknik interpolasi yang lebih baik untuk melengkapi data yang tidak tersedia (*missing value*), khususnya data observasi cuaca sehingga kinerja model menjadi lebih baik.
4. Diperlukan suatu ukuran kesesuaian hasil seleksi fitur input yang mewakili pola korelasi linier dan non-linier dalam satu nilai *MV*.
5. Walaupun model hybrid GSTAR-RNN menghasilkan kinerja training terbaik, namun kinerja testing lebih rendah dibanding ST-RNN. Oleh karenanya diperlukan pengembangan lebih lanjut untuk model hybrid tersebut.

DAFTAR PUSTAKA

- [1] E. Makmur, Y. Koesmaryono, E. Aldrian, and A. H. Wigena, “Wet Season Onset Prediction Model Upon Rice Fields In Northern West Java Using Regional And Global Predictor”, *Jurnal Meteorologi dan Geofisika*, vol. 14, pp. 127–137, 2013.
- [2] Y. S. Swarinoto, Y. Koesmaryono, E. Aldrian, and A. H. Wigena, “Ensemble Prediction System Model For Monthly Rainfall Total Using Weighting Values (Case Of Indramayu District)”, *Jurnal Meteorologi dan Geofisika*, vol. 13, no. 3, pp. 189–200, 2012.
- [3] S. Asadi, J. Shahrabi, P. Abbaszadeh, and S. Tabanmehr, “Neurocomputing A new hybrid artificial neural networks for rainfall – runoff process modeling”, *Neurocomputing*, vol. 121, pp. 470–480, 2013.
- [4] Y. Zou, W. Xue, and S. Liu, “A case study of large-scale parallel I/O analysis and optimization for numerical weather prediction system”, *Future Generating Computer Systems*, vol. 37, pp. 378–389, 2014.
- [5] A. Saikhu, A. Z. Arifin, and C. Fatichah, “Rainfall forecasting by using autoregressive integrated moving average, single input and multi input transfer function”, in *Proceedings of the 11th International Conference on Information and Communication Technology and System (ICTS)*, 2018.
- [6] R. J. May, H. R. Maier, G. C. Dandy, and T. M. K. G. Fernando, “Non-linear variable selection for artificial neural networks using partial mutual information”, *Environmental Modelling & Software*, vol. 23, pp. 1312–1326, 2008.
- [7] S. Bhattacharji, H. J. Neugebauer, J. Reitner, G. M. Friedman, and A. Seilacher, “Nonlinear Time Series Analysis in the Geosciences”, in *Lecture Notes in Earth Sciences*, 2008, p. 390.
- [8] A. Sharma and M. K. Goyal, “Bayesian Network Model for Monthly Rainfall Forecast”, *2015 IEEE 2015 IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India*, pp. 241–246, 2015.
- [9] M. Han, W. Ren, and X. Liu, “Joint mutual information-based input variable selection for multivariate time series modeling”, *Engineering Application. of Artificial Intelligence*, vol. 37, pp. 250–257, 2015.
- [10] R. V. Ramana, B. Krishna, and S. R. Kumar, “Monthly Rainfall Prediction Using Wavelet Neural Network Analysis”, *Water Resource Management*, vol. 27, pp. 3697–3711, 2013.
- [11] J. Sánchez-Monedero, S. Salcedo-Sanz, P. A. Gutiérrez, C. Casanova-Mateo, and C. Hervás-Martínez, “Simultaneous modelling of rainfall occurrence and amount using a hierarchical nominal-ordinal support vector classifier”, *Engineering Application. of Artificial Intelligence*, vol. 34, pp. 199–207, 2014.
- [12] E. G. Ortiz-García, S. Salcedo-Sanz, and C. Casanova-Mateo, “Accurate precipitation prediction with support vector classifiers: A study including novel predictive variables and observational data”, *Atmospheric Research*, vol. 139, pp. 128–136, 2014.
- [13] R. Hashim *et al.*, “Selection of meteorological parameters affecting rainfall

- estimation using neuro-fuzzy computing methodology”, *Atmospheric Research*, vol. 171, pp. 21–30, 2016.
- [14] F. J. Chang, L. C. Chang, C. W. Huang, and I. F. Kao, “Prediction of monthly regional groundwater levels through hybrid soft-computing techniques”, *Journal of Hydrology*, vol. 541, pp. 965–976, 2016.
- [15] N. Acharya, N. A. Shrivastava, B. K. Panigrahi, and U. C. Mohanty, “Development of an artificial neural network based multi-model ensemble to estimate the northeast monsoon rainfall over south peninsular India: An application of extreme learning machine”, *Climate Dynamics*, vol. 43, no. 5–6, pp. 1303–1310, 2014.
- [16] M. E. Banihabib, A. Ahmadian, and F. S. Jamali, “Hybrid DARIMA-NARX model for forecasting long-term daily inflow to Dez reservoir using the North Atlantic Oscillation (NAO) and rainfall data”, *GeoResJ*, vol. 13, pp. 9–16, 2017.
- [17] X. He, H. Guan, and J. Qin, “A hybrid wavelet neural network model with mutual information and particle swarm optimization for forecasting monthly rainfall”, *Journal of Hydrology*, vol. 527, pp. 88–100, 2015.
- [18] A. H. Setyaningrum and P. M. Swarinata, “Weather prediction application based on ANFIS (Adaptive neural fuzzy inference system) method in West Jakarta region”, *2014 International Conference Cyber IT Service Management CITSM 2014*, pp. 113–118, 2014.
- [19] A. Nugroho and B. H. Simanjuntak, “ARMA (Autoregressive Moving Average) Model for Prediction of Rainfall in Regency of Semarang - Central Java - Republic of Indonesia”, *International Journal of Computer Science Issues*, vol. 11, no. 3, pp. 27–32, 2014.
- [20] A. Nugroho, S. Hartati, and K. Mustofa, “Vector Autoregression (Var) Model for Rainfall Forecast and Isohyet Mapping in Semarang–Central Java–Indonesia”, *International Journal of Advanced Computer Science and Applications*, vol. 5, no. 11, pp. 44–49, 2014.
- [21] H. Lee, “General Rainfall Patterns in Indonesia and the Potential Impacts of Local Seas on Rainfall Intensity”, *Water*, vol. 7, no. 4, pp. 1751–1769, 2015.
- [22] S. Hardwinarto and M. Aipassa, “Rainfall Monthly Prediction Based on Artificial Neural Network: A Case Study in Tenggara Station, East Kalimantan - Indonesia”, *Procedia Computer Science*, vol. 59, no. Iccsci, pp. 142–151, 2015.
- [23] F. Nhita, S. Annisa, and S. Kinasih, “Comparative Study of Grammatical Evolution and Adaptive Neuro-Fuzzy Inference System on Rainfall Forecasting in Bandung”, in *2015 3rd International Conference on Information and Communication Technology (ICoICT) A.*, 2015, pp. 6–10.
- [24] M. N. Nur’utami and R. Hidayat, “Influences of IOD and ENSO to Indonesian Rainfall Variability: Role of Atmosphere-ocean Interaction in the Indo-pacific Sector”, *Procedia Environmental Science*, vol. 33, pp. 196–203, 2016.
- [25] D. Munandar, “Optimization Weather Parameters Influencing Rainfall Prediction Using Adaptive Network-Based Fuzzy Inference System (ANFIS) and Linear Regression”, in *2015 International Conference on Data and Software Engineering Optimization Weather*, 2015, pp. 154–159.

- [26] S. C. Nisak, “Seemingly Unrelated Regression Approach for GSTARIMA Model to Forecast Rain Fall Data in Malang Southern Region Districts”, *Cauchy - Journal. Pure & Applied Mathematics*, vol. 4, no. 2, pp. 57–64, 2016.
- [27] J. Toggweiler and R. Key, “Ocean circulation: Thermohaline circulation”, *Encyclopedia Atmospheric Science.*, vol. 4, no. December 2007, p. 1549–1555., 2001.
- [28] J.-I. Hamada, M. D. Yamanaka, J. Matsumoto, S. Fukao, P. A. Winarso, and T. Sribimawati, “Spatial and Temporal Variations of the Rainy Season over Indonesia and their Link to ENSO”, *Journal of the Meteorological Society of Japan*, vol. 80, no. 2, pp. 285–310, 2002.
- [29] E. Aldrian and R. Dwi Susanto, “Identification of three dominant rainfall regions within Indonesia and their relationship to sea surface temperature”, *International Journal of Climatology*, vol. 23, no. 12, pp. 1435–1452, 2003.
- [30] H. H. Hendon, “Indonesian rainfall variability: Impacts of ENSO and local air-sea interaction”, *Journal Climatology*, vol. 16, no. 11, pp. 1775–1790, 2003.
- [31] Supari, “Spatiotemporal Characteristics Of Extreme Rainfall Events Over Java Island, Case: East Java Province”, *Indonesian Journal of Geography, IJG Vol. 44, No. 1, June 2012 (62 - 86)*, 2012.
- [32] E. Aldrian, *Meteorologi Laut Indonesia*, no. November. Jakarta: Badan Meteorologi dan Geofisika, Jakarta, 2008.
- [33] B. Widjanarko Otok, “Development of Rainfall Forecasting Model in Indonesia by using ASTAR, Transfer Function, and ARIMA Methods”, *European Journal of Scientific Research*, vol. 38, no. 3, pp. 1450–216, 2009.
- [34] D. J. Vimont, D. S. Battisti, and R. L. Naylor, “Downscaling Indonesian precipitation using large-scale meteorological fields”, *International Journal of Climatology*, vol. 30, no. 11, pp. 1706–1722, 2010.
- [35] M. V. Fernandes, A. M. Schmidt, and H. S. Migon, “Modelling zero-inflated spatio-temporal processes”, *Statistical Modelling*, vol. 9, no. 1, pp. 3–25, 2009.
- [36] M. Fuentes, B. Reich, and G. Lee, “Spatial-temporal mesoscale modeling of rainfall intensity using gage and radar data”, *Ann. Applied Statistics*, vol. 2, no. 4, pp. 1148–1169, 2008.
- [37] J. H. Faghmous and V. Kumar, “Spatio-Temporal Data Mining for Climate Data: Advances , Challenges , and Opportunities”, *Data Mining, Knowledge. Discovery, Big Data Vol. 1*, pp. 83–116, 2013.
- [38] G. Nowak, A. H. Welsh, T. J. O’Neill, and L. Feng, “Spatio-temporal modelling of rainfall in the Murray-Darling Basin”, *Journal of Hydrology*, vol. 557, pp. 522–538, 2018.
- [39] M. Herrnegger, T. Senoner, and H. P. Nachtnebel, “Adjustment of spatio-temporal precipitation patterns in a high Alpine environment”, *Journal of Hydrology*, vol. 556, pp. 913–921, 2018.
- [40] Lilies Handayani, “Statistical Downscaling Dengan Model Aditif Terampat Untuk Pendugaan Curah Hujan Ekstrim”, *Thesis Sekolah Pasca Sarjana IPB*, 2014.
- [41] A. Jain and A. M. Kumar, “Hybrid neural network models for hydrologic

- time series forecasting”, *Applied Soft Computing Journal*, vol. 7, no. 2, pp. 585–592, 2007.
- [42] A. M. Kalteh, “Monthly river flow forecasting using artificial neural network and support vector regression models coupled with wavelet transform”, *Computer & Geosciences*, vol. 54, pp. 1–8, 2013.
- [43] H. Badrzadeh, R. Sarukkalige, and A. W. Jayawardena, “Hourly runoff forecasting for flood risk management: Application of various computational intelligence models”, *Journal of Hydrology*, vol. 529, pp. 1633–1643, 2015.
- [44] R. C. Deo and M. Şahin, “Application of the extreme learning machine algorithm for the prediction of monthly Effective Drought Index in eastern Australia”, *Atmospheric Research*, vol. 153, pp. 512–525, 2015.
- [45] H. Memarian, M. Pourreza Bilondi, and M. Rezaei, “Drought prediction using co-active neuro-fuzzy inference system, validation, and uncertainty analysis (case study: Birjand, Iran)”, *Theoretical and Applied Climatology*, vol. 125, no. 3–4, pp. 541–554, 2016.
- [46] R. C. Deo, P. Samui, and D. Kim, “Estimation of monthly evaporative loss using relevance vector machine, extreme learning machine and multivariate adaptive regression spline models”, *Stochastic Environmental Research and Risk Assessment*, vol. 30, no. 6, pp. 1769–1784, 2016.
- [47] M. Tiwari, J. Adamowski, and K. Adamowski, “Water demand forecasting using extreme learning machines”, *Journal of Water & Land. Development*, vol. 28, no. 1, 2016.
- [48] J. Du, Y. Liu, Y. Yu, and W. Yan, “A prediction of precipitation data based on Support Vector Machine and Particle Swarm Optimization (PSO-SVM) algorithms”, *Algorithms*, vol. 10, no. 2, 2017.
- [49] F. O. Hocaoglu and F. Serttas, “A novel hybrid (Mycielski-Markov) model for hourly solar radiation forecasting”, *Renewable. Energy*, vol. 108, pp. 635–643, 2015.
- [50] J. Marohasy and J. Abbot, “Assessing the quality of eight different maximum temperature time series as inputs when using artificial neural networks to forecast monthly rainfall at Cape Otway, Australia”, *Atmospheric Research.*, vol. 166, pp. 141–149, 2015.
- [51] M. Han and W. Ren, “Global mutual information-based feature selection approach using single-objective and multi-objective optimization”, *Neurocomputing*, vol. 168, pp. 47–54, 2015.
- [52] X. Zhang, Y. Peng, C. Zhang, and B. Wang, “Are hybrid models integrated with data preprocessing techniques suitable for monthly streamflow forecasting? Some experiment evidences”, *Journal of Hydrology*, vol. 530, pp. 137–152, 2015.
- [53] J. Anderson, H. Van Den Dool, A. Barnston, W. Chen, W. Stern, and J. Ploshay, “Present-Day Capabilities of Numerical and Statistical Models for Atmospheric Extratropical Seasonal Simulation and Prediction”, *Bulletin of the American Meteorological Society*, vol. 80, no. 7, pp. 1349–1361, 1999.
- [54] V. Nourani, Ö. Kisi, and M. Komasi, “Two hybrid Artificial Intelligence approaches for modeling rainfall-runoff process”, *Journal of Hydrology*, vol. 402, no. 1–2, pp. 41–59, 2011.
- [55] Z. M. Yaseen, A. El-shafie, O. Jaafar, H. A. Afan, and K. N. Sayl, “Artificial

- intelligence based models for stream-flow forecasting: 2000-2015”, *J. Hydrol.*, vol. 530, pp. 829–844, 2015.
- [56] X. Wang and M. Han, “Improved extreme learning machine for multivariate time series online sequential prediction”, *Engineering Application & Artificial Intelligent*, vol. 40, pp. 28–36, 2015.
- [57] A. Talei, L. H. C. Chua, C. Quek, and P. E. Jansson, “Runoff forecasting using a Takagi-Sugeno neuro-fuzzy model with online learning”, *Journal of Hydrology*, vol. 488, pp. 17–32, 2013.
- [58] M. Shoaib, A. Y. Shamseldin, B. W. Melville, and M. Muneer, “Runoff forecasting using hybrid Wavelet Gene Expression Programming (WGEP) approach”, *Journal of Hydrology*, vol. 527, pp. 326–344, 2015.
- [59] E. G. Ortiz-garcía, S. Salcedo-sanz, and C. Casanova-mateo, “Accurate precipitation prediction with support vector classifiers : A study including novel predictive variables and observational data”, *Atmospheric Research*, vol. 139, pp. 128–136, 2014.
- [60] O. Kisi and M. Cimen, “Precipitation forecasting by using wavelet-support vector machine conjunction model”, *Engineering.Application &Artificial Intelligent*, vol. 25, no. 4, pp. 783–792, 2012.
- [61] M. Kühnlein, T. Appelhans, B. Thies, and T. Nauss, “Improving the accuracy of rainfall rates from optical satellite sensors with machine learning - A random forests-based approach applied to MSG SEVIRI”, *Remote Sensing of Environment*, vol. 141, pp. 129–143, 2014.
- [62] C. C. F. Plouffe, C. Robertson, and L. Chandrapala, “Comparing interpolation techniques for monthly rainfall mapping using multiple evaluation criteria and auxiliary data sources: A case study of Sri Lanka”, *Environmental Modelling & Software*, vol. 67, pp. 57–71, 2015.
- [63] J. Abbot and J. Marohasy, “Input selection and optimisation for monthly rainfall forecasting in queensland, australia, using artificial neural networks”, *Atmospheric. Research*, vol. 138, pp. 166–178, 2014.
- [64] X. Li, H. R. Maier, and A. C. Zecchin, “Improved PMI-based input variable selection approach for artificial neural network and other data driven environmental and water resource models”, *Environmental Modelling & Software*, vol. 65, pp. 15–29, 2015.
- [65] E. Toth, A. Brath, and A. Montanari, “Comparison of short-term rainfall prediction models for real-time flood forecasting”, *Journal of Hydrology*, vol. 239, no. 1–4, pp. 132–147, 2000.
- [66] A. P. Piotrowski and J. J. Napiorkowski, “A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modelling”, *Journal of Hydrology*, vol. 476, pp. 97–111, 2013.
- [67] V. Nourani, M. Komasi, and A. Mano, “A multivariate ANN-wavelet approach for rainfall-runoff modeling”, *Water Resources Management.*, vol. 23, no. 14, pp. 2877–2894, 2009.
- [68] F. Mekanik, M. A. Imteaz, S. Gato-Trinidad, and A. Elmahdi, “Multiple regression and Artificial Neural Network for long-term rainfall forecasting using large scale climate modes”, *Journal of Hydrology*, vol. 503, pp. 11–21, 2013.
- [69] M. J. Alizadeh, M. R. Kavianpour, O. Kisi, and V. Nourani, “A new

- approach for simulating and forecasting the rainfall-runoff process within the next two months”, *Journal of Hydrology*, vol. 548, pp. 588–597, 2017.
- [70] J. Wu, J. Long, and M. Liu, “Evolving RBF neural networks for rainfall prediction using hybrid particle swarm optimization and genetic algorithm”, *Neurocomputing*, 2015.
- [71] M. Valipour, M. E. Banihabib, and S. M. R. Behbahani, “Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir”, *Journal of Hydrology*, vol. 476, pp. 433–441, 2013.
- [72] R. P. Raymundo and D. M. S. Raymundo, “Artificial Neural Network Model and Multiple Regression Analysis Model in Predicting Rainfall – the Case of Isabela , Philippines”, *Jornal of Art & Science*, vol. 5, no. 4, pp. 243–263, 2012.
- [73] G. J. Sawale and S. R. Gupta, “Use of Artificial Neural Network in Data Mining For Weather Forecasting”, *International Journal of Computer Science & Application*, vol. 6, no. 2, pp. 383–387, 2013.
- [74] N. Acharya *et al.*, “A neurocomputing approach to predict monsoon rainfall in monthly scale using SST anomaly as a predictor”, *Acta Geophysics*, vol. 344, no. 2, pp. 695–707, 2012.
- [75] F. Nhita and Adiwijaya, “A Rainfall Forecasting using Fuzzy System Based on Genetic Algorithm”, *2013 Int. Conference of Informtion & Communication Technology*, pp. 111–115, 2013.
- [76] S. Nurcahyo, F. Nhita, and Adiwijaya, “Rainfall prediction in kemayoran Jakarta using hybrid genetic algorithm (GA) and partially connected feedforward neural network (PCFNN)”, *2nd International Conference on Information and Communication Technology, ICoICT 2014*, pp. 166–171, 2014.
- [77] S. W. Pratama, “Implementation of Local Regression Smoothing and Fuzzy-Grammatical Evolution on Rainfall Forecasting for Rice Planting Calendar”, *2016 Fourth International Conference on Information and Communication Technologies (ICoICT)*, vol. 4, no. c, 2016.
- [78] F. Nhita, D. Saepudin, Adiwijaya, and U. N. Wisesty, “Comparative Study of Moving Average on Rainfall Time Series Data for Rainfall Forecasting Based on Evolving Neural Network Classifier”, *Proc. - 2015 3rd International Symposium Computational & Business Inteligence. ISCBI 2015*, pp. 8–12, 2016.
- [79] K. D. Hartomo and E. Winarko, “ESSPI : Exponential Smoothing Seasonal Planting Index , A New Algorithm For Prediction Rainfall”, *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 14, no. 6, pp. 1–10, 2016.
- [80] B. Kanigoro and A. G. Salman, “Recurrent gradient descent adaptive learning rate and momentum neural network for rainfall forecasting”, *Proc. - 2016 International Seminar of Application Technology & Information Communication, ISEMANTIC 2016*, pp. 23–26, 2017.
- [81] S. Lestari, J.-I. Hamada, F. Syamsudin, Sunaryo, J. Matsumoto, and M. D. Yamanaka, “ENSO Influences on Rainfall Extre[1] S. Lestari, J.-I. Hamada, F. Syamsudin, Sunaryo, J. Matsumoto, and M. D. Yamanaka, ‘ENSO

- Influences on Rainfall Extremes around Sulawesi and Maluku Islands in the Eastern Indonesian Maritime Continent,' *Sola*, vol. 12, ", *Sola*, vol. 12, no. 1, pp. 37–41, 2016.
- [82] J. Abbot and J. Marohasy, "Using lagged and forecast climate indices with artificial intelligence to predict monthly rainfall in the brisbane catchment, Queensland, Australia", *International Journal of Sustainable. Development Planning.*, vol. 10, no. 1, pp. 29–41, 2015.
- [83] Y. S. Andrew W. Robertson, Vincent Morona, "Seasonal predictability of daily rainfall statistics over Indramayu district, Indonesia", *International Journal of Climatology.*, vol. 29, no. 27 Nopember 2008, pp. 1449–1462, 2009.
- [84] J. Abbot and J. Marohasy, "Skilful rainfall forecasts from artificial neural networks with long duration series and single-month optimization", *Atmospheric. Research*, vol. 197, no. July, pp. 289–299, 2017.
- [85] A. Schepen, Q. J. Wang, and D. Robertson, "Evidence for using lagged climate indices to forecast Australian seasonal rainfall", *Journal oof Climatology.*, vol. 25, no. 4, pp. 1230–1246, 2012.
- [86] R. Tresnawati and K. E. Komalasari, "Skenario tenggang waktu sst nino 3.4 terhadap curah hujan untuk meningkatkan akurasi prediksi kalman filter", *Jurnal Meteorologi dan Geofisika*, vol. 12, no. 3, pp. 243–251, 2011.
- [87] Y. S. Swarinoto, A. H. Wigena, and D. Statistika, "Statistical Downscaling Suhu Muka Laut Global Untuk Prediksi Total Hujan Bulanan", *Jurnal Meteorologi dan Geofisika*, vol. 12, pp. 9–19, 2011.
- [88] A. F. R. H. Harisa Bilhaqqi Qalbi, "Future rainfall variability in Indonesia under different ENSO and IOD composites based on decadal predictions of CMIP5 datasets", *J. Phys. Earth and Environmental Science*, vol. 755, p. 011001, 2017.
- [89] G. B. Humphrey, S. Galelli, a Castelletti, H. R. Maier, G. C. Dandy, and M. S. Gibbs, "A new evaluation framework for input variable selection algorithms used in environmental modelling", *International Environmental Modelling and Software Society (iEMSs)*, vol. 62, pp. 33–51, 2014.
- [90] P. Cortez, "Sensitivity Analysis for Time Lag Selection to Forecast Seasonal Time Series using Neural Networks and Support Vector Machines", *Proceedings of the International Joint Conference on Neural Networks*, 2010.
- [91] A. Bagnall, J. Lines, J. Hills, and A. Bostrom, "Time-Series Classification with COTE: The Collective of Transformation-Based Ensembles", *IEEE Transaction of Knowledge and Data Engineering*, vol. 27, no. 9, pp. 2522–2535, 2015.
- [92] M. S. Park and T. Heo, "Seasonal Spatial-temporal Model for Rainfall Data of South Korea", *Journal of Applied Sciences Research*, vol. 5, no. 5, pp. 565–572, 2009.
- [93] R. H. Kripalani and A. Kulkarni, "Rainfall variability over South–east Asia—connections with Indian monsoon and ENSO extremes: new perspectives", *International Journal of Climatology*, vol. 17, no. 11, pp. 1155–1168, 1997.
- [94] J. Sulaiman, H. Darwis, and H. Hirose, "Forecasting monthly maximum 5-

- day precipitation using artificial neural networks with initial lags”, *Sixth International Symposium on Computational Intelligence and Design*, 2013, vol. 2, pp. 3–7, 2013.
- [95] H. D. Tran, N. Muttill, and B. J. C. Perera, “Selection of significant input variables for time series forecasting”, *Environmental Modelling & Software*, vol. 64, pp. 156–163, 2015.
- [96] N. Yuan, Z. Fu, H. Zhang, L. Piao, E. Xoplaki, and J. Luterbacher, “Detrended Partial-Cross-Correlation Analysis: A New Method for Analyzing”, *Scientific Report*, pp. 1–7, 2015.
- [97] X. Qian, Y. Liu, Z. Jiang, B. Podobnik, W. Zhou, and H. E. Stanley, “Detrended partial cross-correlation analysis of two nonstationary time series influenced by common external forces”, *Physical Review Letters*, vol. 062816, pp. 6–11, 2015.
- [98] R. Rak, S. Drozd, J. Kwapien, and P. Oswiecimka, “Detrended cross-correlations between returns, volatility, trading activity, and volume traded for the stock market companies”, *Europhysics Letters*, vol. 48001, pp. 1–6, 2015.
- [99] S. Y. Jiang and L. X. Wang, “Efficient feature selection based on correlation measure between continuous and discrete features”, *Information Processing Letter*, vol. 116, no. 2, pp. 203–215, 2016.
- [100] B. D. Fulcher and N. S. Jones, “Highly comparative feature-based time-series classification”, *IEEE Transaction Knowledge & Data Engineering*, vol. 26, no. 12, pp. 3026–3037, 2014.
- [101] H. Yoon, K. Yang, and C. Shahabi, “Feature subset selection and feature ranking for multivariate time series”, *IEEE Transaction on Knowledge & Data Engineering*, vol. 17, no. 9, pp. 1186–1198, 2005.
- [102] M. Z. Hossain, M. M. Kabir, and M. Shahjahan, “A robust feature selection system with Colin’s CCA network”, *Neurocomputing*, vol. 173, pp. 855–863, 2016.
- [103] L. Yu and H. Liu, “Efficient Feature Selection via Analysis of Relevance and Redundancy”, *Journal of Machine Learning Research* vol. 5, pp. 1205–1224, 2004.
- [104] V. Bolón-Canedo, N. Sánchez-Marroño, and A. Alonso-Betanzos, “A review of feature selection methods on synthetic data”, *Knowledge & Information Syst.*, vol. 34, no. 3, pp. 483–519, 2013.
- [105] A. Jovic and F. Jovic, “Classification of cardiac arrhythmias based on alphabet entropy of heart rate variability time series”, *Biomedical Signal Processing and Control*, vol. 31, pp. 217–230, 2017.
- [106] M. Han and X. Liu, “Feature selection techniques with class separability for multivariate time series”, *Neurocomputing*, vol. 110, pp. 29–34, 2013.
- [107] L. I. U. Tianhong, W. E. I. Haikun, Z. Kanjian, and G. U. O. Weili, “Mutual information based feature selection for multivariate time series forecasting”, in *Proceedings of the 35th Chinese Control Conference*, 2016, pp. 7110–7114.
- [108] Z. Wang, M. Li, and J. Li, “A multi-objective evolutionary algorithm for feature selection based on mutual information with a new redundancy measure”, *Information Science*, vol. 307, pp. 73–88, 2015.

- [109] H. Li, “Accurate and efficient classification based on common principal components analysis for multivariate time series”, *Neurocomputing*, vol. 171, pp. 744–753, 2015.
- [110] R. Taormina and K. W. Chau, “Data-driven input variable selection for rainfall-runoff modeling using binary-coded particle swarm optimization and Extreme Learning Machines”, *Journal of Hydrology*, vol. 529, pp. 1617–1632, 2015.
- [111] M. K. Ebrahimpour and M. Eftekhari, “Ensemble of feature selection methods: A hesitant fuzzy sets approach”, *Applied Soft Computing Journal*, vol. 50, pp. 300–312, 2017.
- [112] G. B. Demisse, T. Tadesse, and Y. Bayissa, “Data Mining Attribute Selection Approach for Drought Modelling: A Case Study for Greater Horn of Africa”, *International Journal of Data Mining & Knowledge Management Processing*, vol. 7, no. 4, pp. 01-16, 2017.
- [113] G. Herman, B. Zhang, Y. Wang, G. Ye, and F. Chen, “Mutual information-based method for selecting informative feature sets”, *Pattern Recognition*, vol. 46, no. 12, pp. 3315–3327, 2013.
- [114] B. Kumari and T. Swarnkar, “Feature Subset Selection in Large Dimensionality Micro array using Wrapper Method: A Review”, *International Journal of Computer Science and Information Technologies*, vol. 2, no. 2, pp. 792–798, 2011.
- [115] N. Clements, S. Sarkar, and W. Wei, “Multiplicative Spatio-Temporal Models”, *Journal of International Energy Policy*, 2014, vol. 3, no. 1, pp. 1–14, 2014.
- [116] R. E. Caraka, D. Candra, and R. Novitasari, “Generalized Spatio Temporal Autoregressive Rainfall-Enso Pattern In East Java Indonesia”, *2018 Indonesia Assoc. Pattern Recognition. International Conference*, pp. 75–79, 2018.
- [117] Edwin Aldrian and Y. S. Djamil, “Spatio-temporal climatic change of rainfall in East Java Indonesia”, *International Journal of Climatology*, vol. 28, no. 12 June 2007, pp. 435–448, 2008.
- [118] A. Abraham, D. Steinberg, and N. S. Philip, “Rainfall Forecasting Using Soft Computing Models and Multivariate Adaptive Regression Splines”, *IEEE SMC Transactions*, vol. 1, pp. 1–6, 2001.
- [119] Z. Beheshti, M. Firouzi, S. M. Shamsuddin, M. Zibarzani, and Z. Yusop, “A new rainfall forecasting model using the CAPSO algorithm and an artificial neural network”, *Neural Comput. Appl.*, vol. 27, no. 8, pp. 2551–2565, 2016.
- [120] S. Makridakis and M. Hibon, “The M3-Competition: results, conclusions and implications”, *International Journal of Forecasting*, vol. 16, pp. 451–476, 2000.
- [121] Info BMKG, "Jangan Abaikan Informasi Cuaca", *BMKG*, 2013.
- [122] K. B. T. and H. L. M. Solomon, S., D.M.Manning, Z.Chen, Marquis, “Climate Change 2007: The Physical Science Basic. Contribution of working group I to the fourth Assesment report of the intergovernmental panel on Climate Change”, *Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA.*, no. October, pp. 1–113, 2010.
- [123] Pusat Database BMKG, <http://dataonline.bmkg.go.id/>

- [124] D. Gunawan and U. A. Linarka, “Penentuan Prediktor Untuk Prediksi Curah Hujan Bulanan”, *Jurnal Meteorologi dan Geofisika*, no. 2, pp. 93–102.
- [125] Giarno, Z. L. Dupe, and M. A. Mustofa, “Kajian awal musim hujan dan awal musim kemarau di indonesia”, *Jurnal Meteorologi dan Geofisika*, vol. 13, no. 1, pp. 1–8, 2012.
- [126] BMKG, *Prakiraan Musim Kemarau 2015 di Indonesia*. Jakarta, 2015.
- [127] D. Syaifullah, “Kajian Sea Surface Temperature (SST), Southern Oscillation Index (SOI) dan Dipole Mode Pada Kegiatan Penerapan Teknologi Modifikasi Cuaca”, *Jurnal Sains Teknologi Modifikasi Cuaca*, vol. 11, no. 1, pp. 1–8, 2010.
- [128] K. Vasmilla, “A Survey on Time Series Data Mining”, *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 2, no. 5, pp. 170–179, 2014.
- [129] A. Saikhu, A. Z. Arifin, and C. Fatichah, “Rainfall Forecasting by Using Autoregressive Integrated Moving Average, Single Input and Multi Input Transfer Function”, in *International Conference on Information & Communication Technology and System (ICTS)*, 2017.
- [130] C. Saikhu, Ahmad, Arifin, Agus Zainal, Fatichah, “Non-Linear Spatio-Temporal Input Selection for Rainfall Forecasting Using Recurrent Neural Networks”, 2018 *International Seminar on Intelligent Technology & Its Application (ISITIA)*, 2018.
- [131] A. Baker and C. Brunsdon, “Non-linearities in drip water hydrology: an example from Stump Cross Caverns, Yorkshire”, *Journal of Hydrology*, vol. 277, no. 3–4, pp. 151–163, 2003.
- [132] R. May, G. Dandy, and H. Maier, “Review of Input Variable Selection Methods for Artificial Neural Networks.”
- [133] A. Saikhu, A. Z. Arifin, and C. Fatichah, “Correlation and Symmetrical Uncertainty based Feature Selection for Multivariate Time Series Classification”, *International Journal of Intelligent Engineering & Systems*, vol. 12, No.3, 2019.
- [134] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection”, *Journal of Machine Learning Research*, vol. 3, no. 3, pp. 1157–1182, 2003.
- [135] C. N. Babu and B. E. Reddy, “A moving-average filter based hybrid ARIMA–ANN model for forecasting time series data”, *Applied Soft Computing*, vol. 23, pp. 27–38, 2014.
- [136] G. P. Reganata, “Inflow and outflow forecasting of currency using multi-input transfer function”, *Proceedings of the IConSSE FSM SWCU*, pp. 53–65, 2015.
- [137] S. Astutik, N. Iriawan, Suhartono, and Sutikno, “Hybrid state-space model and adjusting procedure based on Bayesian approaches for spatio-temporal rainfall disaggregation”, *ICSSBE 2012 - Proceedings, 2012 International Conference on Statistics in Science, Business and Engineering (ICSSBE)*, no. 1, pp. 24–27, 2012.
- [138] B. N. Ruchjana, A. S. Abdullah, I. G. Nyoman, M. Jaya, and E. Hermawan, “R Software for Parameter Estimation of Spatio R Software for Parameter Estimation of Spatio Temporal Model”, no. December, 2016.

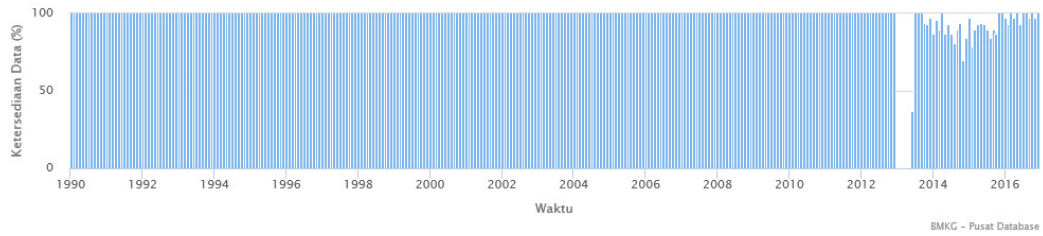
- [139] D. Astuti, B. N. Ruchjana, and Soemartini, “Generalized space time autoregressive with exogenous variable model and its application”, *Journal of Physics: Conference Series*, vol. 893, no. 1, 2017.
- [140] Setiawan, Suhartono, and M. Prastuti, “S-GSTAR-SUR model for seasonal spatio temporal data forecasting”, *Malaysian Journal of Mathematical Science*, vol. 10, pp. 53–65, 2016.
- [141] M. Wegener, “Handbook of Regional Science”, *Handbook of Regional Science*, pp. 741–758, 2014.
- [142] G. P. Zhang, “Time series forecasting using a hybrid ARIMA and neural network model”, *Neurocomputing*, vol. 50, pp. 159–175, 2003.
- [143] G. Ciaburro, B. Venkateswaran, *Neural Networks with R*. Packt Publishing, 2017.
- [144] N. D. Lewis, *Neural Networks For Time Series Forecasting With R: Intuitive Step by Step Blueprint for Beginners*. 2017.
- [145] J. G. De Gooijer and R. J. Hyndman, “25 Years of Time Series Forecasting”, *International Journal of Forecasting*, vol. 22, no. 3, pp. 443–473, 2006.
- [146] L. Wang, Z. Wang, and S. Liu, “An effective multivariate time series classification approach using echo state network and adaptive differential evolution algorithm”, *Expert System with Application*, vol. 43, pp. 237–249, 2016.
- [147] N. Challita, M. Khalil, and P. Beausery, “New technique for feature Selection: combination between Elastic Net and Relief Nicole”, in *Technological Advances in Electrical, Electronics and Computer Engineering (TAECE)*, 2015, pp. 3–8.
- [148] D. Palchak, “An Artificial Neural Network in Short-Term Electrical Load Forecasting of a University Campus: A Case Study”, *Journal of Energy Resource Technology*, vol. 135, no. 3, p. 032001, 2013.
- [149] V. Nourani and S. Mousavi, “Spatiotemporal groundwater level modeling using hybrid artificial intelligence-meshless method”, *Journal of Hydrology*, vol. 536, pp. 10–25, 2016.
- [150] T. Cao, E. P. Lim, Z. H. Zhou, T. B. Ho, D. Cheung, and H. Motoda, “Advances in knowledge discovery and data mining: 19th pacific-asia conference, PAKDD 2015 Ho Chi Minh City, Vietnam, May 19-22, 2015 proceedings, part I”, *Lecture Notes Computer Science*, vol. 9077, pp. 199–211, 2015.

Halaman ini sengaja dikosongkan

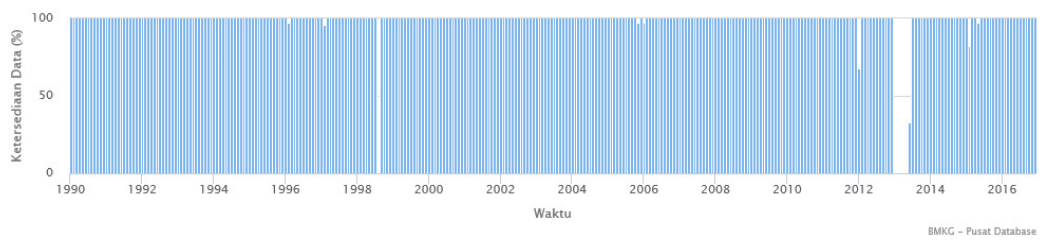
LAMPIRAN I: DATA

I.A. Ketersediaan Data Cuaca dan Indeks Iklim

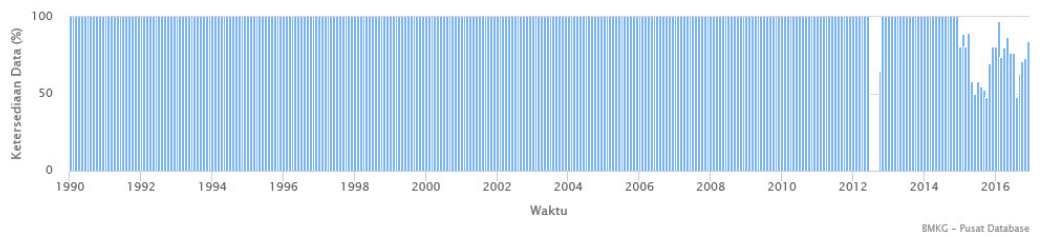
Grafik Ketersediaan Data Curah Hujan
STASIUN METEOROLOGI JUANDA



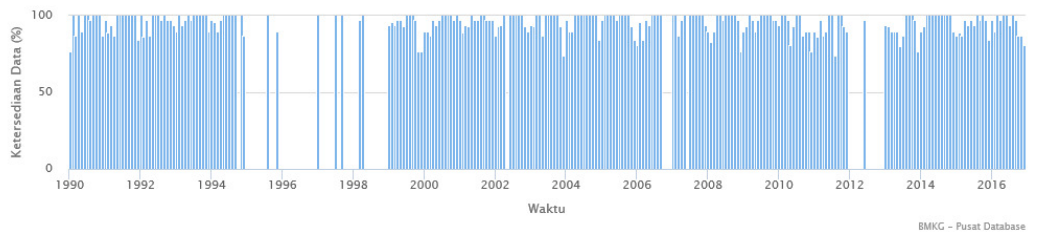
Grafik Ketersediaan Data Kecepatan Angin Rata-rata
STASIUN METEOROLOGI JUANDA



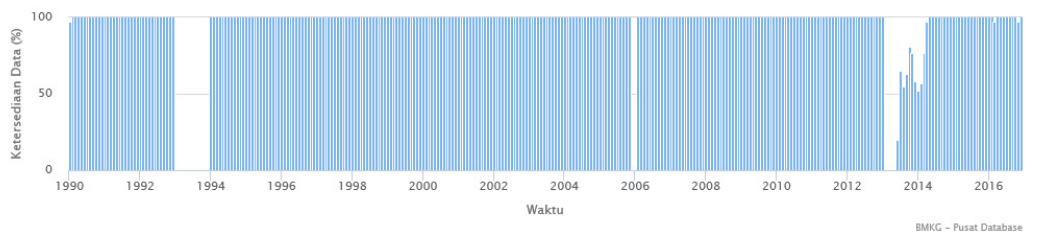
Grafik Ketersediaan Data Curah Hujan
STASIUN METEOROLOGI PERAK I



Grafik Ketersediaan Data Lama Penyinaran Matahari
STASIUN METEOROLOGI PERAK I



Grafik Ketersediaan Data Kelembaban Rata-rata
STASIUN GEOFISIKA KARANG KATES



I.B. Data Studi Kasus 1

Tabel I.B.1 Data Meteorologi dan Indeks Iklim di Juanda

min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
22.4	35.5	27.95	81.42	5.46	3.55	10	2.48	2.68	2.11	0.2
22.6	34.4	27.54	86	4.45	3.29	13	1.98	2.09	1.8	0.21
22.1	34.7	28	84.42	5.52	3.35	15	2.42	1.57	1.36	0.1
22.6	34.1	27.98	85.17	4.22	3.13	15	3.25	1.47	0.98	-0.02
21.8	32.7	27.57	85.87	4.6	3.39	11	3.77	1.66	0.93	0.13
20.2	33.4	26.82	79.4	7.25	3.57	10	4.07	1.5	0.6	-0.17
18.1	32.5	25.8	76.74	7.78	4.03	9	3.77	0.94	-0.06	-0.31
18.3	33.7	26.28	74.32	7.93	4.71	9	2.89	0.77	-0.2	-0.47
19.2	35.2	27.47	68.8	7.88	4.43	9	1.5	0.22	-0.45	-0.73
19.8	36.4	28.78	71.48	6.45	4.26	9	1.07	-0.38	-0.94	-0.83
22.4	35.5	28.22	75.37	4.95	3.9	11	0.53	-0.77	-1.03	-0.85
21.9	34.5	27.74	79.45	5.14	3.39	9	0.18	-0.63	-0.85	-0.84
21.2	33.1	26.49	84.65	3.35	3.65	8	-0.31	-0.48	-0.63	-0.83
21.6	33.5	26.43	85.69	3.53	3.93	13	-0.73	-0.08	-0.16	-0.63
.....

Tabel I.B.2 Data Meteorologi dan Indeks Iklim di Perak 1

T_min	T_max	T_mean	RH	Solar_Rad	Win_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
23.2	33.8	27.83	81.42	4.726	2.516	8	2.48	2.68	2.11	0.2
22.6	33.2	27.67	82.79	4.557	2.643	8	1.98	2.09	1.8	0.21
23.5	33.2	28.08	83.16	4.923	2.581	8	2.42	1.57	1.36	0.1
23	34.1	28.07	84.53	4.377	2.333	8	3.25	1.47	0.98	-0.02
23	32.9	27.59	85.32	4.368	2.419	8	3.77	1.66	0.93	0.13
21.5	33.2	27.2	77.57	7.12	2.267	5	4.07	1.5	0.6	-0.17
19.5	32.6	26.4	73.77	7.813	2.032	6	3.77	0.94	-0.06	-0.31
20.5	34	26.92	71.23	7.771	2.548	8	2.89	0.77	-0.2	-0.47
21	36.4	28.09	65.6	7.737	2.333	8	1.5	0.22	-0.45	-0.73
22.3	35.8	28.86	71.9	5.639	2.226	10	1.07	-0.38	-0.94	-0.83
22.1	34.9	28.16	77.7	4.523	2.5	10	0.53	-0.77	-1.03	-0.85
22	33.8	27.65	80.42	5.09	2.194	8	0.18	-0.63	-0.85	-0.84
22	32.2	26.69	84.71	4.59	2.71	10	-0.31	-0.48	-0.63	-0.83
23	31.7	26.35	86.04	3.507	2.786	8	-0.73	-0.08	-0.16	-0.63
.....

Tabel I.B.3 Data Meteorologi dan Indeks Iklim di Perak 2

min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
24.8	34.6	28.33	70.81	5.24	3.1	6	2.48	2.68	2.11	0.2
24.1	34.6	27.29	71.43	5	3.36	6	1.98	2.09	1.8	0.21
24.9	34.8	27.05	66.06	5.14	3.03	8	2.42	1.57	1.36	0.1
24.4	34.4	27.64	69.07	5.01	2.7	6	3.25	1.47	0.98	-0.02
24	33.7	27.88	70.52	4.52	3.03	6	3.77	1.66	0.93	0.13
22.9	33.5	27.5	72.07	7.21	3.23	7	4.07	1.5	0.6	-0.17
21.1	32.8	25.68	62.87	7.69	3.39	7	3.77	0.94	-0.06	-0.31
22	33.6	27.07	55.74	7.77	3.81	7	2.89	0.77	-0.2	-0.47
22.9	34.9	28.22	63.8	7.97	3.37	6	1.5	0.22	-0.45	-0.73
24	35.7	24.65	63.9	6.01	2.97	6	1.07	-0.38	-0.94	-0.83
23.2	35	25.94	64.97	5.12	3.4	9	0.53	-0.77	-1.03	-0.85
24.2	35.2	27.38	74.71	5.66	3	7	0.18	-0.63	-0.85	-0.84
23	33.6	27.4	69.81	4.76	3.1	6	-0.31	-0.48	-0.63	-0.83
23.9	32.9	27.08	78.55	3.7	3.9	9	-0.73	-0.08	-0.16	-0.63
.....

I.C. Data Studi Kasus 2

Tabel I.C.1 Data Meteorologi dan Indeks Iklim di Karangates

Rainfall	min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
161	20.4	33.3	26.68	85.65	5.63	1.44	3	-0.99	-1.67	-1.31	-0.59
72	18	39	25.71	79.8	7.13	3.57	5	-1.53	-2.08	-1.6	-0.8
4	16.2	33.3	23.74	79.39	6.63	2.5	6	-1.62	-1.98	-1.64	-0.83
4	19	33.5	25.18	76.84	6.17	3.45	6	-1.61	-1.55	-1.39	-0.82
22	19	34.3	26.19	74.7	6.65	3.87	6	-1.46	-0.9	-0.9	-0.59
143	20.4	34.5	26.41	80.13	4.79	1.58	6	-1.52	-1.53	-1.98	-1.56
182	20.6	32	25.9	83.07	3.52	0	0	-1.21	-1.74	-2.07	-1.62
144	19	33.2	25.18	81.06	4.83	0	0	-1.21	-1.86	-1.98	-1.66
318	20.8	33.8	25.68	85.26	4.18	0	0	-0.48	-1.54	-2.1	-1.85
190	20	32.5	25.19	86	3.21	0	0	-0.06	-0.99	-1.57	-1.47
244	19.4	33	25.15	87.58	4.32	0	0	0.28	-1.18	-1.47	-1.23
146	19.8	32.9	25.62	81.9	5.15	0	0	0.1	-0.88	-1.22	-1.04
232	19.5	33.5	25.61	82.74	5.71	0	0	-1.04	-0.78	-0.8	-0.55
186	19	32.8	25.11	82.5	5.1	0	0	-1.04	-0.16	-0.49	-0.86
.....

Tabel I.C.2 Data Meteorologi dan Indeks Iklim di Karangploso

Rainfall	T_min	T_max	T_mean	RH	Solar_Rad	Win_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
0	13.9	29	21.29	70.06	6.97	2.23	5	0.21	0.41	0.71	0.62
18	14.9	30.9	22.44	69.57	6.47	2.17	5	0.26	0.13	0.35	0.39
0	15.8	32.4	23.97	60.65	6.81	2.26	6	0.25	0.36	0.75	0.92
163	18.2	30	23.33	80.17	3.81	2.07	5	0.35	0.9	1.13	0.77
373	18	29.8	23.55	78.94	4.39	2.35	9	0.34	1.09	1.64	0.98
476	19.2	29.2	22.87	81.65	4.54	2.29	6	0.18	1.17	1.7	0.7
308	19.5	30.6	23.34	79.34	3.33	2.44	9	0.44	0.95	1.56	0.76
402	17.2	29.8	23.17	81.81	4.46	2.13	9	1.2	0.82	1.23	0.73
239	19.4	29.6	23.77	81.13	4.32	2.29	10	1.83	1.14	1.18	0.65
15	17.6	29.8	23.55	75.77	6.3	2.52	9	1.82	1.24	1.05	0.53
10	16.9	29	22.8	71.93	6.7	2.3	9	0.6	0.66	0.59	0.4
21	13	28.9	21.49	74.42	6.52	2.55	10	-0.4	-0.07	0.29	0.57
33	15.8	28.8	22.24	72.87	5.84	2.55	10	-0.19	-0.18	-0.01	0.17
56	16.9	30.4	22.88	77.9	5.39	2.83	10	-0.18	-0.16	-0.13	0.17
110	17.8	30.6	23.49	77.19	5.2	2.71	10	0.1	-0.35	-0.43	0
.....

Tabel I.C.3 Data Meteorologi dan Indeks Iklim di Tretes

Rainfall	min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
299	16.7	26.7	22.13	89.16	1.4	8	2.16	-0.99	-1.67	-1.31	-0.59
142	15.8	26.5	21.49	84.47	4.1	9	2.1	-1.53	-2.08	-1.6	-0.8
54	15	26.4	21.11	81.58	6.8	9	1.97	-1.62	-1.98	-1.64	-0.83
170	16.4	27.3	21.77	78.39	7.2	8	2.52	-1.61	-1.55	-1.39	-0.82
0	16.7	28.3	23.29	69.23	6.9	12	2.43	-1.46	-0.9	-0.9	-0.59
78	18.2	29.3	23.49	74.68	7.1	10	2.26	-1.52	-1.53	-1.98	-1.56
257	17.8	27.8	22.06	86.63	7.4	9	1.9	-1.21	-1.74	-2.07	-1.62
680	16.4	25.6	20.88	91.03	7.6	9	2.42	-1.21	-1.86	-1.98	-1.66
353	17.3	26	21.12	91.94	6.8	9	2.19	-0.48	-1.54	-2.1	-1.85
544	17.8	27.1	20.61	94.14	7.6	20	3.64	-0.06	-0.99	-1.57	-1.47
429	17.3	25.1	20.84	92.84	6.4	8	2.94	0.28	-1.18	-1.47	-1.23
468	17.4	26	21.26	92	6.6	8	2.53	0.1	-0.88	-1.22	-1.04
451	16.9	26.2	21.56	90.26	7.4	7	3.32	-1.04	-0.78	-0.8	-0.55
577	16.3	25.5	20.73	92.5	7	10	3.2	-1.04	-0.16	-0.49	-0.86
2	13.4	29.2	22.54	69.83	3.6	11	4.7	-0.75	-0.34	-0.46	-0.36
.....

I.D. Data Studi Kasus 3

Tabel I.D.1 Data Meteorologi dan Indeks Iklim di Juanda

Rainfall	min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
10	24.4	30.7	26.6	86	0.2	2	6	-0.494	-1.10	-1.08	-0.40
2.5	24.2	31.4	27.5	82	3.2	2	4	-0.417	-1.05	-1.00	-0.41
0	23.8	31.6	26.4	85	0.6	2	6	-0.275	-0.95	-0.97	-0.43
69.5	19.3	30.2	26.3	92	2.4	8	4	0.046	-0.86	-0.92	-0.31
4.8	20.6	30.9	26.4	89	2.8	2	4	0.204	-0.80	-0.85	-0.31
42.8	20.3	31.3	26.8	88	5.8	2	6	0.361	-0.77	-0.85	-0.32
3.8	21.4	33.1	26.8	87	4	2	5	0.29	-0.80	-0.86	-0.33
62.3	23.8	31.1	27.6	86	6.4	3	8	0.07	-0.85	-0.87	-0.30
0.9	24.2	31.4	25.6	88	7.2	1	5	-0.075	-0.89	-0.91	-0.39
34.6	24.2	31.9	27.5	86	6.7	3	6	-0.223	-1.00	-0.95	-0.47
0	22.5	32	27.8	86	5.6	2	5	-0.231	-1.12	-1.04	-0.51
34	21.6	32.5	27.6	84	7	2	4	-0.329	-1.28	-1.16	-0.57
0	23.1	33.4	27.3	84	5.8	3	7	-0.687	-1.30	-1.16	-0.54
4.8	23.8	33.5	27.4	84	4	4	9	-0.709	-1.18	-1.08	-0.48
0.6	24.8	31.1	27	83	0.4	3	6	-0.568	-1.12	-1.16	-0.47
.....

Tabel I.D.2 Data Meteorologi dan Indeks Iklim di Perak 1

Rainfall	T_min	T_max	T_mean	RH	Solar_Rad	Win_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
18	24.8	31	27.3	81	0	2	6	-0.494	-1.10	-1.08	-0.40
0	25	32.8	27.9	78	0.2	3	6	-0.417	-1.05	-1.00	-0.41
17	24.4	33	28.1	78	0.3	3	10	-0.275	-0.95	-0.97	-0.43
13.2	24.2	31	27.3	82	0.1	3	5	0.046	-0.86	-0.92	-0.31
0.2	24.6	32	27.6	82	0.1	1	3	0.204	-0.80	-0.85	-0.31
0.4	24.8	32.4	27.9	80	0.2	2	6	0.361	-0.77	-0.85	-0.32
3.8	25	33.6	28.4	79	0.4	2	5	0.29	-0.80	-0.86	-0.33
14.3	24.6	32	28.1	80	0.2	2	5	0.07	-0.85	-0.87	-0.30
0.2	25.2	32	27.8	83	0.1	2	5	-0.075	-0.89	-0.91	-0.39
3.8	25.7	33.1	28.4	80	0.5	2	5	-0.223	-1.00	-0.95	-0.47
0	25.3	32.8	28.4	83	0.3	2	4	-0.231	-1.12	-1.04	-0.51
3.8	25	33.6	28.2	79	0.5	3	5	-0.329	-1.28	-1.16	-0.57
57.4	24.6	34	27.8	80	0.6	3	8	-0.687	-1.30	-1.16	-0.54
0	24.4	31	27.6	84	0.1	3	6	-0.709	-1.18	-1.08	-0.48
8.1	24	30.8	27.9	77	0	3	6	-0.568	-1.12	-1.16	0.47

Tabel I.B.3 Data Meteorologi dan Indeks Iklim di Perak 2

Rainfall	min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
8.7	23.9	31.3	27.4	82	2.4	1	10	-0.494	-1.10	-1.08	-0.40
0	24.4	33.2	28.4	78	3.3	2	11	-0.417	-1.05	-1.00	-0.41
8.5	24.3	34.2	29	77	3.7	1	13	-0.275	-0.95	-0.97	-0.43
1.2	24.6	31.5	27.7	85	2.5	1	9	0.046	-0.86	-0.92	-0.31
4	24.5	32.4	28.7	80	2.6	1	9	0.204	-0.80	-0.85	-0.31
25.6	25.5	32.8	28.7	78	4.2	1	9	0.361	-0.77	-0.85	-0.32
42.7	24	33.8	28.7	78	6	1	12	0.29	-0.80	-0.86	-0.33
10	24	33	28.7	80	3.5	2	10	0.07	-0.85	-0.87	-0.30
12	24.6	33	28.2	84	2.2	2	8	-0.075	-0.89	-0.91	-0.39
1.2	25.3	32.4	29.1	78	3.8	2	9	-0.223	-1.00	-0.95	-0.47
0	25.8	33.2	28.6	85	4	1	13	-0.231	-1.12	-1.04	-0.51
0	25.6	34.2	29.6	80	5.8	1	10	-0.329	-1.28	-1.16	-0.57
57.2	23.9	35	29.7	76	7	1	13	-0.687	-1.30	-1.16	-0.54
4	23.4	31.2	28.8	80	0.4	2	10	-0.709	-1.18	-1.08	-0.48
5.5	25.2	32	28.6	78	0.3	2	11	-0.568	-1.12	-1.16	-0.47
.....

I.E. Data Studi Kasus 4

Tabel I.E.1 Data Meteorologi dan Indeks Iklim di Karangates

Rainfall	min_T	max_T	mean_T	RH	Solar_Rad	Wind_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
1	23.2	32	26.2	86	4	1	2	0.492	1.33	1.772	1.327
10	23.9	33	26.9	85	6	0	1	0.353	1.287	1.766	1.342
6	24	32.9	27.3	84	7.5	1	3	0.222	1.296	1.768	1.329
10	23.6	31.2	27.2	84	3	1	4	0.163	1.267	1.782	1.357
1	23.8	31.6	26.3	83	7	0	1	0.097	1.225	1.817	1.429
0	23.8	31.8	26.8	83	3.5	0	1	0.149	1.22	1.875	1.48
0	23	31.4	26.4	86	1	0	1	0.133	1.222	1.83	1.438
15	22.6	31.4	25.7	88	3.7	0	1	0.041	1.194	1.763	1.404
12	22.9	30.8	24.2	92	3.5	0	1	0.099	1.196	1.755	1.377
7	22.8	33.1	25.6	88	1.8	0	2	0.123	1.195	1.741	1.364
2	23.6	30.8	25.9	81	3.5	2	3	0.216	1.153	1.702	1.415
20	22.6	31	26.2	85	3.5	2	5	0.304	1.188	1.699	1.433
44	21.8	31	25.7	83	2.5	2	3	0.251	1.232	1.742	1.455
1	21.8	32	25.7	83	6.5	1	3	0.201	1.25	1.768	1.448
6	22.6	32	26.3	76	7	1	2	0.164	1.223	1.7	1.398
.....

Tabel I.E.2 Data Meteorologi dan Indeks Iklim di Karangploso

Rainfall	T_min	T_max	T_mean	RH	Solar_Rad	Win_Speed	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
6	20.6	29.2	24.1	86	4.2	1	3	0.492	1.33	1.772	1.327
34	21.2	30	24.9	81	6.1	2	4	0.353	1.287	1.766	1.342
0	22.4	30.3	25.4	78	4.8	0	4	0.222	1.296	1.768	1.329
1	22.2	28.7	24.8	85	6.1	2	5	0.163	1.267	1.782	1.357
0	21	30	23.5	85	6	2	8	0.097	1.225	1.817	1.429
39	20.4	29	24	83	3.5	1	2	0.149	1.22	1.875	1.48
9	21.4	27.3	23.9	84	0.5	0	3	0.133	1.222	1.83	1.438
0	21	27.4	23.7	83	0.9	1	4	0.041	1.194	1.763	1.404
4	21.3	26.4	22.7	90	2.7	2	4	0.099	1.196	1.755	1.377
31	20.8	27.6	23.6	87	2.5	1	3	0.123	1.195	1.741	1.364
0	21.3	27.9	23.5	88	5	1	4	0.216	1.153	1.702	1.415
9	21.4	29	24	84	4.5	1	4	0.304	1.188	1.699	1.433
3	21.7	29	24.1	85	3.5	2	4	0.251	1.232	1.742	1.455
9	20.8	29.4	24.7	76	0	2	6	0.201	1.25	1.768	1.448
3	21.7	29.4	23.7	78	5.1	2	5	0.164	1.223	1.7	1.398
.....

Tabel I.E.3 Data Meteorologi dan Indeks Iklim di Tretes

Rainfall	min_T	max_T	mean_T	RH	Solar_Rad	Wind_Spee	Max_WS	Nino1_2	Nino3	Nino3_4	Nino4
0	18	24	21.8	95	0	3	0	0.49172	1.33	1.7717	1.327
62	19.6	24	22.6	89	3	4	1	0.35252	1.287	1.76603	1.342
96	16.8	26.4	23.3	88	6	6	2	0.2223	1.296	1.76839	1.329
4	16.8	24.6	21.4	96	1.2	2	0	0.16263	1.267	1.78159	1.357
0	18.8	25	21.1	94	0	4	0	0.09725	1.225	1.81737	1.429
88	18.8	25	21.1	94	1.2	3	1	0.14897	1.22	1.87485	1.48
13	17.8	25.6	20.1	95	1.3	3	1	0.13284	1.222	1.83044	1.438
51	18	24.6	21.2	95	0	3	0	0.04147	1.194	1.76334	1.404
35	18.6	23.2	20.4	94	1.3	3	0	0.0993	1.196	1.75466	1.377
85	19.4	25.5	21.5	90	1	7	0	0.12271	1.195	1.74089	1.364
30	19.2	24.9	21.6	94	0.5	3	0	0.21608	1.153	1.70189	1.415
7	18.8	24.8	21.6	94	0	4	0	0.30351	1.188	1.69932	1.433
57	18	25.4	20.8	96	1	3	0	0.25055	1.232	1.74197	1.455
39	18	26.4	21.6	90	0	7	2	0.20062	1.25	1.76848	1.448
16	15.2	25.8	21.2	91	2.5	5	1	0.16382	1.223	1.6997	1.398
.....

LAMPIRAN II: PROGRAM

II.A. PROGRAM UTAMA

```
## LOAD DATA
juanda <- read.csv("juanda.csv", header=TRUE)
juanda_M<-as.matrix(juanda)
perak1 <- read.csv("perak1.csv", header=TRUE)
perak1_M<-as.matrix(perak1)
perak2 <- read.csv("perak2.csv", header=TRUE)
perak2_M<-as.matrix(perak2)
gab_3lok <- array( c( juanda_M, perak1_M, perak2_M), dim= c( nrow(juanda_M), ncol(juanda_M), 3 ) )
saveRDS(rho_M.status, file= "rho_M.status.partial.rda")
rho_M.status <- readRDS(file= "rho_M.status.partial.rda")

##### MODEL 2 RNN : PCSUF - RNN #####
rho.M.status <- DPCCA_extract(gab_3lok, 7)
PC.Status <-Select_Linear_Relevant(gab_3lok)
SU.Status <-Select_NonLinear_Relevant(gab_3lok)
antar.lok.PC <- Create_feature_predictor_linear(PC.Status)
antar.lok.SU <- create_feature_predictor_nonLinear(SU.Status)
PC.indeks <- create_index_rank_linear(10)
SU.indeks <- create_index_rank_non_linear(10)
rem_lin_NonLin <- remove_redundant_linear_and_nonLinear(10)
Generate_data_as_RNN_input(PC_SU.Status.opt)
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)
#####

##### MODEL 3 RNN : DPCCA - RNN #####
rho.M.status <- DPCCA_extract(gab_3lok, 7)
rho_M.status[1, , ] <- 0
rho_M1.status <- rho_M.status
Generate_data_as_RNN_input(rho_M1.status)
nilai.merit.value <<- Create_feature_based_on_DPCCA(rho_M1.status)
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)
#####

##### MODEL 4 RNN : PCF - RNN #####
rho.M.status <- DPCCA_extract(gab_3lok, 7)
PC.Status <-Select_Linear_Relevant(gab_3lok)
antar.lok.PC <- Create_feature_predictor_linear(PC.Status)
PC.indeks <- create_index_rank_linear(15)
rem_linear <- remove_linear_redundant(10)
antar.lok.RNN <- Generate_data_as_RNN_input(PC.Status.opt)
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)
#####

##### MODEL 5 RNN : SUF - RNN #####
rho.M.status <- DPCCA_extract(gab_3lok, 7)
SU.Status <-Select_NonLinear_Relevant(gab_3lok)
antar.lok.SU <- create_feature_predictor_nonLinear(SU.Status)
SU.indeks <- create_index_rank_non_linear(25)
rem_nonLin <- remove_non_linear_redundant(10)
Generate_data_as_RNN_input_model245(SU.Status.opt)
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)

##### MODEL 6 : RNN : DPCCA - PCSU without remove - RNN #####
rho.M.status <- DPCCA_extract(gab_3lok, 7)
merit.value <- create_feature_without_remove_redundant()
Performance_RNN <- Compute_Model_Performance_GSTAR_RNN()
```

II.B. Program Spatio-Temporal Recurrent Neural Networks

```
## Penggabungan Data
data1 <- read.csv("opt.3juanda-model4.csv")
data2 <- read.csv("opt.3perak1-model4.csv")
data3 <- read.csv("opt.3perak2-model4.csv")
data1 <- data.matrix(data1[,-1])
data2 <- data.matrix(data2[,-1])
data3 <- data.matrix(data3[,-1])

# time series plot
library(ggplot2)
ggplot(data.frame(data1), aes(x = 1:nrow(data2), y = `X.1`)) + geom_line()
ggplot(data.frame(data2), aes(x = 1:nrow(data2), y = `X.1`)) + geom_line()
ggplot(data.frame(data3), aes(x = 1:nrow(data3), y = `X.1`)) + geom_line()
ggplot(data.frame(data1[1:100,]), aes(x = 1:100, y = `X.1`)) + geom_line()
ggplot(data.frame(data2[1:100,]), aes(x = 1:100, y = `X.1`)) + geom_line()
ggplot(data.frame(data3[1:100,]), aes(x = 1:100, y = `X.1`)) + geom_line()

data_tot <- list(data1, data2, data3)
prop <- 0.75
N = nrow(data1)

## Mendefinisikan dimensi hasil training dan testing
result_train_all <- array(data=NA, dim=c(floor(N*prop), 3, 3), dimnames=NULL)
result_test_all <- array(data=NA, dim=c(ceiling((N*(1-prop))), 3, 3), dimnames=NULL)

##Normalisasi dan unscale data
range_data<-function(data, min, max){(data-min)/(max-min)}
unscale_data<-function(x,max_x, min_x)
{x*(max_x-min_x)+min_x}

## mendefinisikan matriks kinerja
korelasi <- matrix(data=NA, 3, 2)
R_sq <- matrix(data=NA, 3, 2)
RMSE <- matrix(data=NA, 3, 2)
AIC_correction <- matrix(data=NA, 3, 1)
jumpred=0

## fungsi2 pengukuran kinerja
RootMSE = function(m, o){
  sqrt(mean((m - o)^2))
}

## fungsi MSE
MSE = function(m, o){
  mean((m-o)^2)
}

## fungsi AIC correction
AICC = function(MSE, n, p)
{
  n*log(MSE)+2*(p+1) + 2*(p+1)*(p+2)/(n-p)
}

## Mengubah menjadi per lokasi
library(rnn)
library(Metrics)
for (lok in 1:3)
{
  data_tot[[lok]][is.na(data_tot[[lok]])] <- 0

  #Normalisasi
  data <-as.matrix(data_tot[[lok]])
```

```

p = ncol(data)
n_train <- floor(prop*N)
min_Y <- min(data[,1])
max_Y <- max(data[,1])
min_Ytest <- min(data[(n_train+1):N,1])
max_Ytest <- max(data[(n_train+1):N,1])
for(kol in 2:p)
{
  min_data <-min(data[,kol])
  max_data <-max(data[,kol])
  data[,kol] <- range_data(data[,kol], min_data, max_data)
}
Y_train<- as.matrix(t(data[1:n_train,1]))
Y_train<- range_data(Y_train, min_Y, max_Y)
X_train <- array(data=NA, dim=c(1, n_train, p-1), dimnames=NULL)
X_train[1, , ] <- data[1:n_train, 2:p]

## create your model and add layers
set.seed(1000)
model<-trainr(Y=Y_train, X=X_train,
  learningrate=0.1, hidden_dim=c(16, 12, 4, 2),
  numepochs=875,
  momentum=0.8, use_bias = F, learningrate_decay = 1,
  seq_to_seq_unsync = F, update_rule = "sgd",
  network_type="rnn",
  epoch_function = c(epoch_print, epoch_annealing), loss_function = loss_L1,
  sigmoid = c("logistic", "Gompertz", "tanh"))

## see your model structure
summary(model)

## plot error by epoch
error<-t(model$error)
rownames(error) <-1:nrow(error)
colnames(error) <- "error"
plot(error)

##evaluasi kinerja model
pred_train <- t(predictr(model,X_train, hidden = FALSE))

## Mempersiapkan data testing model
X_test<-array(data=NA, dim=c(1, N-n_train, p-1), dimnames=NULL)
X_test[1, , ] <- data[(n_train+1):N, 2:p]
Y_test<-as.matrix(data[(n_train+1):N,1])
Y_test <-range_data(Y_test, min_Ytest, max_Ytest)

pred_test <-t(predictr(model, X_test, hidden = F, real_output = T))

# Unscaling data and prediksi
Y_train_actual <-t(unscale_data(Y_train, max_Y, min_Y))
train_actual <-unscale_data(pred_train,max_Y, min_Y)
Y_test_actual<-unscale_data(Y_test, max_Ytest, min_Ytest)
test_actual<-unscale_data(pred_test,max_Ytest, min_Ytest)

err_train <- Y_train_actual - train_actual
err_test <- Y_test_actual - test_actual

result_train<-cbind(Y_train_actual, round(train_actual, 2), round(err_train,2))
result_test<-cbind(Y_test_actual, round(test_actual,2), round(err_test, 2))

# Perhitungan kinerja data actual
r_train_model <- round(cor(Y_train_actual, train_actual),3)
R_sq_train <- (r_train_model)^2

```



```

RMSE_train <- RootMSE(train_actual, Y_train_actual)

r_test_model <- round(cor(Y_test_actual, test_actual),3)
RMSE_test <- RootMSE(test_actual, Y_test_actual)
R_sq_test <- (r_test_model)^2

result_train_all[, ,lok] <- result_train
result_test_all[, ,lok] <- result_test
korelasi[lok,1] <- r_train_model
korelasi[lok,2] <- r_test_model
R_sq[lok, 1] <- R_sq_train
R_sq[lok, 2] <- R_sq_test
RMSE[lok, 1] <- RMSE_train
RMSE[lok, 2] <- RMSE_test

jumpred=jumpred+p

## Perhitungan AIC correction
MSE_value = MSE(Y_train_actual, train_actual)
AIC_correction[lok,1] <- AICC(MSE_value,n_train,p)
}

## Kinerja gabungan seluruh lokasi
correlation <- matrix(data=NA, 1, 2)
RMSE_total<- matrix(data=NA, 1, 2)
R_sq_total <- matrix(data=NA, 1, 2)
AIC_correction_Tot <- matrix(data=NA, 1, 1)

for (kol in 1:2)
{
  correlation [1, kol]<- mean(korelasi[,kol])
  R_sq_total[1, kol] <- mean(R_sq[, kol])
  RMSE_total[1, kol] <- sum(RMSE[, kol])
}
AIC_correction_Tot <- mean(AIC_correction[,1])

print(correlation)
print(R_sq_total)
print(RMSE_total)
print(AIC_correction_Tot)
print(jumpred/3)

#### Plot Aktual vs Training
for (lok in 1:3)
{
  plot(result_train_all[,1, lok], col="blue", type="l",main =c("Training: Data Aktual vs Prediksi", lok),
xlab="bulan ke-", ylab="curah hujan (mm)", lwd = 2)
  lines(result_train_all[,2,lok], type = "l", col = "red", lwd = 1)
  legend("topright", legend=c("aktual", "prediksi"),
  col=c("blue","red"),lty=1:2, cex=1, box.lty=2, box.lwd=2)
}

#### Plot Aktual vs Testing
for (lok in 1:3)
{
  plot(result_test_all[,1, lok], col="blue", type="l",main =c("Testing: Data Aktual vs Prediksi", lok),
xlab="bulan ke-", ylab="curah hujan (mm)", lwd = 2)
  lines(result_test_all[,2,lok], type = "l", col = "red", lwd = 1)
  legend("topright", legend=c("aktual", "prediksi"),
  col=c("blue","red"),lty=1:2, cex=1, box.lty=2, box.lwd=2)
}

```

II.C. Fungsi-Fungsi Pendukung

```
DPCCA_CC=function(x,y,k){
  xx<- cumsum(x - mean(x))
  yy<- cumsum(y - mean(y))
  slide_win_xx = mat_sliding_window(xx,k)
  slide_win_yy = mat_sliding_window(yy,k)
  x_hat = t(apply(slide_win_xx,1,function(n) (lm(n~seq(1:length(n)))$fitted.values)))
  y_hat = t(apply(slide_win_yy,1,function(n) (lm(n~seq(1:length(n)))$fitted.values)))
  F2_dfa_x = c()
  F2_dfa_y = c()
  for(i in 1:nrow(x_hat)){
    ## Equation 4
    F2_dfa_x = c(F2_dfa_x,mean((xx[i:(i+k-1)]-x_hat[i,])^2))
  }

  mat_sliding_window = function(xx,k){
    ## Function to generate boxes given dataset(xx) and box size (k)
    slide_mat=c()
    for (i in 1:(length(xx)-k+1)){
      slide_mat = rbind(slide_mat,xx[i:(i+k-1)])
    }
    return(slide_mat)
  }

  for(i in 1:nrow(y_hat)){
    ## Equation 4
    F2_dfa_y = c(F2_dfa_y,mean((yy[i:(i+k-1)]-y_hat[i,])^2))
  }
  ## Average detrend variance over all boxes to obtain fluctuation
  F2_dfa_x = mean(F2_dfa_x) ## Equation 3
  F2_dfa_y = mean(F2_dfa_y) ## Equation 3

  ## Get detrended covariance of two profile
  F2_dcca = c()
  for(i in 1:nrow(x_hat)){
    ## Equation 5
    F2_dcca = c(F2_dcca,mean((xx[i:(i+k-1)]-x_hat[i,]) * (yy[i:(i+k-1)]-y_hat[i,])))
  }

  ## Equation 6
  F2_dcca = mean(F2_dcca)

  ## Calculate correlation coefficient partial
  rho = F2_dcca / sqrt(F2_dfa_x * F2_dfa_y) ## Equation 1
  rM <- matrix(c(1, rho, rho, 1), nrow=2, ncol=2, byrow = TRUE)
  inv_rM <- solve(rM)
  rho_partial = -inv_rM[1,2]/((sqrt(inv_rM[1,1])* sqrt(inv_rM[2,2])))
  return(rho_partial)
}

shift<-function(x,shift_by){
  stopifnot(is.numeric(shift_by))
  stopifnot(is.numeric(x))

  if (length(shift_by)>1)
    return(sapply(shift_by,shift, x=x))

  out<-NULL
  abs_shift_by=abs(shift_by)
  if (shift_by > 0 )
    out<-c(tail(x,-abs_shift_by),rep(NA,abs_shift_by))
  else if (shift_by < 0 )
    out<-c(rep(NA,abs_shift_by), head(x,-abs_shift_by))
}
```

```

else
  out<-x
out
}

matrix.index <- function(a, value) {
  idx <- which(data.frame(a)==value)
  col.num <- ceiling(idx/nrow(a))
  row.num <- idx - (col.num-1) * nrow(a)
  return(c(row.num, col.num))
}

mat_sliding_window = function(xx,k){
  ## Function to generate boxes given dataset(xx) and box size (k)
  slide_mat=c()
  for (i in 1:(length(xx)-k+1)){
    slide_mat = rbind(slide_mat,xx[i:(i+k-1)])
  }
  return(slide_mat)
}

trace<-function(A)
{return(ifelse(is.square(A),sum (diag(A)),NA));
}

is.square<-function(A)
{return(is.matrix(A)&&(nrow(A)== ncol(A)));
}

cbind.fill<-function(...){
  nm <- list(...)
  nm<-lapply(nm, as.matrix)
  n <- max(sapply(nm, nrow))
  do.call(cbind, lapply(nm, function (x)
    rbind(x, matrix(, n-nrow(x), ncol(x)))))
}

## fungsi rootMSE
RootMSE = function(m, o){
  sqrt(mean((m - o)^2))
}

## fungsi MSE
MSE = function(m, o){
  mean((m-o)^2)
}

## fungsi AIC correction
AICC = function(n, p)
{
  AICc = -n*log(MSE)+2(p+1) + 2(p+1)(p+2)/(n-p)
}

DPCCA_extract <- function(gab_3lok, k) {
rho_M <- array(data = NA, dim=c(13, 12, 3, 3), dimnames = NULL)
for(lok1 in 1:3) {
  Rainfall <- gab_3lok[,1,lok1]
  for(col in 2:ncol(gab_3lok))
  {
    for(lok2 in 1:3) {
      {
        Prediktor <- gab_3lok[,col,lok2]
        x <- Prediktor

```

```

y <- Rainfall
rho_M[1, col, lok1, lok2] <- DPCCA_CC(x,y,k)

for(t in 1:12)
{
x <- Prediktor[-(((length(Prediktor)+1)-t):length(Prediktor))]
y <- Rainfall[-(1:t)]
rho_M[t+1,col, lok1, lok2] <- DPCCA_CC(x,y,k)
}
## Perhitungan dpcca untuk Pasangan Rainfall
Prediktor <- gab_3lok[,1,lok2]
x <- Prediktor
y <- Rainfall
if(identical(x,y)==TRUE) {rho_M[1, 1, lok1, lok2] <-1}
else {rho_M[1, 1, lok1, lok2] <- DPCCA_CC(x,y,k)}

for(t in 1:12)
{
x <- Prediktor[-(((length(Prediktor)+1)-t):length(Prediktor))]
y <- Rainfall[-(1:t)]
rho_M[t+1,1,lok1, lok2] <- DPCCA_CC(x,y,k)
}
rownames(rho_M) <- paste('lag ke', 1:13)
colnames(rho_M) <- paste('var', 1:12)
}
}
}
rho_M <<- rho_M
rho_M.status <<- ifelse(abs(rho_M[ , , ]) < 0.04811, 0, 1)
return(rho_M.status)
saveRDS(rho_M.status, file= "rho_M.status.partial.rda")
}

Select_Linear_Relevant <- function(gab_3lok)
{
##menghitung korelasi (PC) untuk seleksi Relevan
PC_M <- array(data = NA, dim=c(13, 12, 3, 3), dimnames = NULL)
for(lok1 in 1:3) {
Rainfall <- gab_3lok[,1,lok1]
for(col in 2:ncol(gab_3lok))
{
for(lok2 in 1:3) {
{
Prediktor <- gab_3lok[ ,col,lok2]
if (rho_M.status[1,col, lok1, lok2]==0) {PC_M[1,col,lok1,lok2] <- 0}
else {
x <- Prediktor
y <- Rainfall
PC_M[1,col,lok1,lok2] <- cor(x,y)
}
}
}
}
}
for(t in 1:12)
{
if (rho_M.status[t+1, col, lok1, lok2]==0) {PC_M[t+1, col, lok1, lok2] <-0}
else
{
x <- Prediktor[-(((length(Prediktor)+1)-t):length(Prediktor))]
y <- Rainfall[-(1:t)]
PC_M[t+1,col, lok1, lok2] <- cor(x,y)
}
}
}
}
}

```

```

}

##Perhitungan PC untuk Pasangan Rainfall
for(lok2 in 1:3) {
  if(rho_M.status[1, 1, lok1, lok2]==0){PC_M[1, 1, lok1, lok2] <- 0}
  else {
    Prediktor <- gab_3lok[,1,lok2]
    x <- Prediktor
    y <- Rainfall
    {PC_M[1, 1, lok1, lok2] <- cor(x, y)}

    for(t in 1:12)
    {
      x <- Prediktor[-((length(Prediktor)+1)-t):length(Prediktor)]
      y <- Rainfall[-(1:t)]
      PC_M[t+1,1,lok1, lok2] <- cor(x, y)
    }
  }
}
PC_M <<- PC_M
PC.Status <<- ifelse(abs(PC_M[, , ]) < 0.113, 0, 1)
return(PC.Status)
}

Select_NonLinear_Relevant <- function(gab_3lok) {
## SU dan Status_SU untuk memilih relevan non linier
library(FSelector)

SU_M <- array(data = NA, dim=c(13, 12, 3, 3), dimnames = NULL)

##menghitung Symmetrical Unc. (SU) untuk seleksi Relevan
for(lok1 in 1:3) {
  Rainfall <- gab_3lok[,1,lok1]
  for(col in 2:ncol(gab_3lok))
  {
    for(lok2 in 1:3) {
      {
        Prediktor <- gab_3lok[,col,lok2]
        if (rho_M.status[1,col, lok1, lok2]==0) {SU_M[1,col, lok1, lok2] <- 0}
        else {
          x <- Prediktor
          y <- Rainfall
          x <- data.frame(x)
          y <- data.frame(y)
          z <- data.frame(c(x, y))
          colnames(z)<-c("x", "y")
          SU_M[1,col, lok1, lok2] <- as.numeric(symmetrical.uncertainty(x~, z))
        }
      }

      for(t in 1:12)
      {
        if (rho_M.status[t+1, col, lok1, lok2]==0) {SU_M[t+1, col, lok1, lok2] <-0}
        else
        {
          x <- Prediktor[-((length(Prediktor)+1)-t):length(Prediktor)]
          y <- Rainfall[-(1:t)]
          x <- data.frame(x)
          y <- data.frame(y)
          z <- data.frame(c(x, y))
          colnames(z)<-c("x", "y")
          SU_M[t+1,col, lok1, lok2] <- as.numeric(symmetrical.uncertainty(x~, z))
        }
      }
    }
  }
}

```

```

    }
  }
}
##Perhitungan SU untuk Pasangan Rainfall
for(lok2 in 1:3) {
  if(rho_M.status[1, 1, lok1, lok2]==0){SU_M[1, 1, lok1, lok2] <- 0}
  else{
    Prediktor <- gab_3lok[,1,lok2]
    x <- Prediktor
    y <- Rainfall
    x <- data.frame(x)
    y <- data.frame(y)
    z <- data.frame(c(x, y))
    SU_M[1, 1, lok1, lok2] <- as.numeric(symmetrical.uncertainty(x~., z))

    for(t in 1:12)
    {
      x <- Prediktor[-(((length(Prediktor)+1)-t):length(Prediktor))]
      y <- Rainfall[-(1:t)]
      x <- data.frame(x)
      y <- data.frame(y)
      z <- data.frame(c(x, y))
      SU_M[t+1,1,lok1, lok2] <- as.numeric(symmetrical.uncertainty(x~., z))
    }
  }
}
rownames(rho_M.status) <- paste('lag ke', 1:13)
colnames(rho_M.status) <- paste('var', 1:12)
}
SU_M <<- SU_M
SU.Status <<- ifelse(abs(SU_M[, , ]) < 0.1, 0, 1)
return(SU.Status)
}
Create_feature_predictor_linear <- function(PC.Status) {
## Time-lags linier var predictor
antar.lok.PC<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3, 3),dimnames=NULL)
lokasi<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3),dimnames=NULL)
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    y <- NULL
    for(col in 1:12)
    {
      count_t<-13
      x_lag<-matrix(data=NA,nrow(gab_3lok),count_t,dimnames=NULL)
      t <- 1
      repeat {
        if(t==1){
          x_lag[,t]<- as.matrix(gab_3lok[ ,col,lok1])
          if(PC.Status[t,col,lok1,lok2]==0)
            {x_lag[,t]<- NA}
        }
        else
        {
          lag <- (t-1)
          x_lag[,t]<- as.matrix(shift(gab_3lok[ ,col,lok1], -lag))
          if(PC.Status[t,col,lok1,lok2]==0){
            x_lag[,t]<- NA}
        }
      }
      colnames(x_lag) <- (0:(count_t-1))
      colnames(x_lag) <- paste(col,"~t-",colnames(x_lag),sep="")
      t= t+1
      if (t > 13){

```

```

        break
      }
    }
    y[[col]] <- x_lag
  }
  z <- do.call(cbind, y)
  lokasi[, , lok1] <- z
}
antar.lok.PC[, , , lok2] <- lokasi
}
return(antar.lok.PC <<- antar.lok.PC)
}

create_feature_predictor_nonLinear <- function(SU.Status) {
## create fitur time lags non linier
antar.lok.SU<-array(data=NA, dim=c(nrow(gab_3lok),ncol=156, 3, 3),dimnames=NULL)
lokasi<-array(data=NA, dim=c(nrow(gab_3lok), ncol=156,3),dimnames=NULL)
matriks.SU <- array(data=NA, dim=c(nrow=156, ncol=156, 3, 3),dimnames=NULL)

for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    y <- NULL
    for (col in 1:12)
    {
      count_t<-13
      x_lag<-matrix(data=NA,nrow(gab_3lok),count_t,dimnames=NULL)
      t <- 1
      repeat {
        if(t==1){x_lag[,t]<- as.matrix(gab_3lok[,col,lok1])
        if(SU.Status[t,col,lok1,lok2]==0){x_lag[,t]<- NA}}
        else{
          lag <- (t-1)
          x_lag[,t]<- as.matrix(shift(gab_3lok[,col, lok1], -lag))
          if(SU.Status[t,col,lok1,lok2]==0){x_lag[,t]<- NA}}
        colnames(x_lag) <- (0:(count_t-1))
        colnames(x_lag) <- paste(col,"~t-",colnames(x_lag),sep="")
        t= t+1
        if (t > 13){
          break
        }
      }
      y[[col]] <- x_lag
    }
    z <- do.call(cbind, y)
    lokasi[, , lok1] <- z
  }
  antar.lok.SU[, , , lok2] <- lokasi
}
return(antar.lok.SU <<- antar.lok.SU)
}

create_index_rank_linear <- function(max_rank)
{
##Hitung matriks korelasi dan ranking prediktor linier
r <- array(data=NA, dim=c(nrow=156 , ncol=156,3, 3),dimnames=NULL)
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    r[, ,lok1, lok2] <- cor(antar.lok.PC[, , lok1, lok2],use='pairwise.complete.obs')
  }
}
}

```

```

    round(r, digits=3)
  }
r <<- r
}

## Ranking PC_M
maks_rankPC <- max_rank
PC2.rank<-array(data=NA, dim=c(nrow(PC_M), ncol(PC_M),3),dimnames=NULL)
PC.rank<-array(data=NA, dim=c(nrow(PC_M), ncol(PC_M),3,3),dimnames=NULL)
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    d.rank <- matrix(rank(-(abs(PC_M[, ,lok1,lok2]))), nrow = 13, ncol = 12)
    d.rank[d.rank[, , ]>maks_rankPC]=0
    PC2.rank[, ,lok1] <- d.rank
  }
  PC.rank[, , , lok2] <- PC2.rank
}
PC.rank <<- PC.rank

## Memberi indeks prediktor relevan berdasar PC
PC.indeks <- array(data=NA, dim=c(maks_rankPC,2,3,3),dimnames=NULL)
PC2.indeks <- array(data=NA, dim=c(maks_rankPC,2,3),dimnames=NULL)
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    ind <- NULL
    ## maks <- sum(PC.Status[, ,lok1, lok2]==1)
    for (rank in 1:maks_rankPC) ## harusnya maks s/d total PC.Status=1
    {
      indeks <- as.matrix(matrix.index(PC.rank[, ,lok1, lok2], rank))
      indeks <- t(indeks)
      ind[[rank]] <- indeks
    }
    ind2 <- do.call(rbind, ind)
    PC2.indeks[, , lok1] <- ind2
  }
  PC.indeks[, , , lok2] <- PC2.indeks
}
return (PC.indeks <<- PC.indeks)
}

create_index_rank_non_linear <- function(max_rank)
{
  matriks.SU <- array(data=NA, dim=c(nrow=156, ncol=156, 3, 3),dimnames=NULL)
  ## Menghitung SU var prediktor
  for (lok2 in 1:3)
  {
    for (lok1 in 1:3)
    {
      for (kol1 in 1:156)
      {
        Var1 <- antar.lok.SU[, kol1,lok1,lok2]
        for (kol2 in 1:156)
        {
          x <- as.data.frame(antar.lok.SU[, kol2,lok1,lok2])
          y <- as.data.frame(Var1)
          z <-as.data.frame(c(x, y))
          colnames(z)<-c("x", "y")
          ifelse((all(is.na(z$x))||all(is.na(z$y))),next,
            matriks.SU[kol1,kol2,lok1,lok2]<-as.numeric(symmetrical.uncertainty(x~, z)))
        }
      }
    }
  }
}

```



```

    }
  }
}
}
matriks.SU <<- matriks.SU

# Ranking SU_M
maks_rankSU <- max_rank
SU2.rank<-array(data=NA, dim=c(nrow(SU_M), ncol(SU_M),3),dimnames=NULL)
SU.rank<-array(data=NA, dim=c(nrow(SU_M), ncol(SU_M),3,3),dimnames=NULL)
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    d.rank <- matrix(rank(-(abs(SU_M[ , ,lok1,lok2]))), nrow = 13, ncol = 12)
    d.rank[d.rank[ , ]>maks_rankSU]=0
    SU2.rank[ , ,lok1] <- d.rank
  }
  SU.rank[ , , ,lok2] <- SU2.rank
}
SU.rank <<- SU.rank

## Memberi indeks prediktor relevan berdasar SU
SU.indeks <- array(data=NA, dim=c(maks_rankSU,2,3,3),dimnames=NULL)
SU2.indeks <- array(data=NA, dim=c(maks_rankSU,2,3),dimnames=NULL)
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    ind <- NULL
    ## maks <- sum(SU.Status[ , ,lok1,lok2]==1)
    for (rank in 1:maks_rankSU) ## harusnya maks s/d total SU.Status=1
    {
      indeks <- as.matrix(matrix.index(SU.rank[ , ,lok1,lok2], rank))
      indeks <- t(indeks)
      ind[[rank]] <- indeks
    }
    ind2 <- do.call(rbind, ind)
    SU2.indeks[ , ,lok1] <- ind2
  }
  SU.indeks[ , , ,lok2] <- SU2.indeks
}
SU.indeks <<- SU.indeks
return(SU.indeks <<-SU.indeks)
}

##### Remove Redundan berdasar alpha-beta (Model 2)
##-----
## Membandingkan matriks korelasi, SU dengan alpha-beta
remove_redundant_linear_and_nonLinear <- function(max_rank) {
max_rank <- 2
lokasi<-array(data=NA, dim=c(nrow(gab_3lok), ncol=156,3),dimnames=NULL)
maks_rankSU <- max_rank
maks_rankPC <- max_rank
trace1.merit.value <-0
alpha <- 0.2
beta <- 0.2
delta <- 0.05
alpha.opt <- 0
beta.opt <- 0

while (alpha < 0.5)
{

```

```

while (beta <0.5)
{
  ##remove PC.rank
  for (lok2 in 1:3)
  {
    for(lok1 in 1:3)
    {
      for (rank in 1:(maks_rankPC-1))
      {
        x1_bar <- PC.indeks[rank,1,lok1,lok2]
        x1_kol <- PC.indeks[rank,2,lok1,lok2]
        x1 <-x1_bar+((x1_kol-1)*13)
        for (rank2 in 2:maks_rankPC)
        {x2_bar <- PC.indeks[rank2, 1, lok1, lok2]
        x2_kol <- PC.indeks[rank2, 2, lok1, lok2]
        x2 <-x2_bar+((x2_kol-1)*13)
        ifelse(abs(r[x1,x2,lok1,lok2])>alpha,PC.rank[x2_bar,x2_kol,lok1,lok2]<-
NA,PC.rank[x2_bar,x2_kol,lok1,lok2]<-1)
        }
      }
    }
  }

  ## remove SU.rank
  for (lok2 in 1:3)
  {
    for (lok1 in 1:3)
    {
      for (rank in 1:(maks_rankSU-1))
      {
        x1_bar <- SU.indeks[rank,1,lok1,lok2]
        x1_kol <- SU.indeks[rank,2,lok1,lok2]
        x1 <-x1_bar+((x1_kol-1)*13)
        for (rank2 in 2:maks_rankSU)
        {x2_bar <- SU.indeks[rank2, 1, lok1, lok2]
        x2_kol <- SU.indeks[rank2, 2, lok1, lok2]
        x2 <-x2_bar+((x2_kol-1)*13)
        ifelse(abs(matriks.SU[x1,x2,lok1,lok2])>beta,SU.rank[x2_bar,x2_kol,lok1,lok2]<-
NA,SU.rank[x2_bar,x2_kol,lok1,lok2]<-1)
        }
      }
    }
  }

  ## Mengubah kode PC_SU.status
  PC.rank1 <-ifelse(PC.rank!=0,1,0)
  SU.rank1 <-ifelse(SU.rank!=0,1,0)
  PC.rank1[is.na(PC.rank1)]=0
  SU.rank1[is.na(SU.rank1)]=0
  merit.SU <- array(data=NA, dim=c(nrow=156 , ncol=156,3, 3),dimnames=NULL)
  PC_SU.Status <-PC.rank1+SU.rank1

  ## buat array time lags kembali untuk antar.lok.PC_SU
  antar.lok.PC_SU<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3, 3),dimnames=NULL)
  antar.lok.PC_SU <- create_feature_predictor_nonLinear(PC_SU.Status)
  merit.SU <- compute_matrix_SU_for_PC(antar.lok.PC_SU)
  merit.value <- compute_Matrix_Merit_Value(SU.Status, merit.SU)
  trace.merit.value <- abs(trace(merit.value))
  if (trace1.merit.value < trace.merit.value)
  {trace1.merit.value<-trace.merit.value
  PC_SU.Status.opt<-PC_SU.Status
  alpha.opt <- alpha
  beta.opt <- beta

```

```

    } else {
      trace1.merit.value
      PC_SU.Status.opt
      alpha
      beta}
      beta<-beta+delta
    }
    alpha<-alpha+delta
  }
  jum.pred.opt <- 0
  for (lok2 in 1:3)
  {
    for (lok1 in 1:3)
    {
      jum.pred.opt<-length(which(PC_SU.Status.opt[ , ,lok1,lok2]==1))+length(which(PC_SU.Status.opt[ , ,lok1,lok2]==2))-1
    }
  }
  PC_SU.Status.opt[1,1, , ] <- 0
  return(remove_lin_nonLin <<-list(trace1.merit.value, alpha.opt, beta.opt, PC_SU.Status.opt, jum.pred.opt, matriks.merit))
}

```

FUNGSI MENGHITUNG MATRIKS SU UNTUK MERIT VALUE DARI AUGMENTED MAT ANTAR_LOK_PC

```

compute_matrix_SU_for_PC <- function(antar.lok) {
  merit.SU <- array(data=NA, dim=c(nrow=156 , ncol=156,3, 3),dimnames=NULL)
  for (lok2 in 1:3)
  {
    for (lok1 in 1:3)
    {
      for (kol1 in 1:156)
      {
        Var1 <-antar.lok[ ,kol1,lok1,lok2]
        for (kol2 in 1:156)
        {
          x <- as.data.frame(antar.lok.PC[ ,kol2,lok1,lok2])
          y <- as.data.frame(Var1)
          z <- as.data.frame(c(x, y))
          colnames(z)<-c("x", "y")
          ifelse((all(is.na(z$x))|(all(is.na(z$y))),next,
            merit.SU[kol1,kol2,lok1,lok2]<-as.numeric(symmetrical.uncertainty(x~, z)))
        }
      }
    }
  }
  return(merit.SU <<-merit.SU)
}

```

FUNGSI MENGHITUNG MATRIKS SU UNTUK MERIT VALUE DARI AUGMENTED MAT ANTAR_LOK_PC

```

compute_matrix_SU_for_SU <- function(antar.lok) {
  merit.SU <- array(data=NA, dim=c(nrow=156 , ncol=156,3, 3),dimnames=NULL)
  for (lok2 in 1:3)
  {
    for (lok1 in 1:3)
    {
      for (kol1 in 1:156)
      {
        Var1 <-antar.lok[ ,kol1,lok1,lok2]
        for (kol2 in 1:156)
        {
          x <- as.data.frame(antar.lok.SU[ ,kol2,lok1,lok2])

```

```

    y <- as.data.frame(Var1)
    z <- as.data.frame(c(x, y))
    colnames(z)<-c("x", "y")
    ifelse((all(is.na(z$x))||all(is.na(z$y))),next,
           merit.SU[kol1,kol2,lok1,lok2]<-as.numeric(symmetrical.uncertainty(x~., z)))
  }
}
}
return(merit.SU <<-merit.SU)
}

## Menghitung Merit Value
compute_Matrix_Merit_Value <- function(PC.Status, merit.SU)
{
num_merit <- matrix(data=NA,nrow=3, ncol=3, dimnames=NULL)
merit.value <- matrix(data=NA,nrow=3, ncol=3, dimnames=NULL)

for (lok2 in 1:3)
{
for (lok1 in 1:3)
{
jum_pred <- length(which(PC.Status[ , ,lok1,lok2]==1))+length(which(PC.Status[ , ,lok1,lok2] == 2))-1
merit.SU[is.na(merit.SU)]=0
for (kol2 in 2:ncol(merit.SU))
{
merit.SU[1,1,lok1,lok2]<-abs(merit.SU[1,1,lok1,lok2])+abs(merit.SU[1,kol2,lok1,lok2])
numerator <- as.numeric(merit.SU[1,1,lok1,lok2])
num_merit[lok1,lok2] <- as.numeric(numerator)

for (kol1 in kol2:nrow(merit.SU))
{
merit.SU[2,2,lok1,lok2]<-merit.SU[2,2,lok1,lok2]+merit.SU[kol1,kol2,lok1,lok2]
}
}
denominator <- merit.SU[2, 2, lok1, lok2]
merit.value[lok1,
lok2]<-num_merit[lok1,lok2]/(sqrt(jum_pred+jum_pred*(jum_pred-
1)*(denominator/jum_pred)))
}
}
matriks.merit <<- merit.value
}

##### remove redundant linier (Model 4)
## Membandingkan matriks korelasi dengan alpha
remove_linear_redundant <- function(max_rank)
{
lokasi<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3),dimnames=NULL)
maks_rankPC <- max_rank
trace1.merit.value <-0
alpha <- 0.2
delta <- 0.05
alpha.opt <- 0
while (alpha < 0.5)
{
##remove PC.rank
for (lok2 in 1:3)
{
for(lok1 in 1:3)
{
for (rank in 1:(maks_rankPC-1))
{
x1_bar <- PC.indeks[rank,1,lok1,lok2]
x1_kol <- PC.indeks[rank,2,lok1,lok2]

```

```

    x1 <-x1_bar+((x1_kol-1)*13)
    for (rank2 in 2:maks_rankPC)
    {x2_bar <- PC.indeks[rank2, 1, lok1, lok2]
    x2_kol <- PC.indeks[rank2, 2, lok1, lok2]
    x2 <-x2_bar+((x2_kol-1)*13)
    ifelse(abs(r[x1,x2,lok1,lok2])>alpha,PC.rank[x2_bar,x2_kol,lok1,lok2]<-
NA,PC.rank[x2_bar,x2_kol,lok1,lok2]<-1)
    }
  }
}

## Mengubah kode PC.Status
PC.rank1 <-ifelse(PC.rank!=0,1,0)
PC.rank1[is.na(PC.rank1)]=0
PC.Status <-PC.rank1

## buat array time lags kembali untuk antar.lok.PC
antar.lok.PC<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3, 3),dimnames=NULL)
antar.lok.PC <- Create_feature_predictor_linear(PC.Status)
merit.SU <- compute_matrix_SU_for_PC(antar.lok.PC)
merit.value <- compute_Matrix_Merit_Value(PC.Status, merit.SU)
trace.merit.value <- abs(trace(merit.value))
if (trace1.merit.value < trace.merit.value)
{trace1.merit.value<-trace.merit.value
  PC.Status.opt<-PC.Status
  alpha.opt <- alpha
} else {
  trace1.merit.value
  PC.Status.opt
  alpha}
alpha<-alpha+delta
}
jum.pred.opt <- 0
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    jum.pred.opt <- jum.pred.opt + length(which(PC.Status.opt[ , ,lok1,lok2]==1))-1
  }
}
PC.Status.opt <<- PC.Status.opt
PC.Status.opt[1,1, , ] <- 0
return(remove_lin <<-list(trace1.merit.value, alpha.opt, PC.Status.opt, jum.pred.opt, matriks.merit))
}

## Membandingkan matriks SU dengan beta (model 5)
remove_non_linear_redundant <- function(max_rank)
{
lokasi<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3),dimnames=NULL)
maks_rankSU <- max_rank
trace1.merit.value <-0
beta <- 0.2
delta <- 0.05
beta.opt <- 0

while (beta < 0.5)
{

## remove SU.rank
for (lok2 in 1:3)
{
  for (lok1 in 1:3)

```

```

{
  for (rank in 1:(maks_rankSU-1))
  {
    x1_bar <- SU.indeks[rank,1,lok1,lok2]
    x1_kol <- SU.indeks[rank,2,lok1,lok2]
    x1 <-x1_bar+((x1_kol-1)*13)
    for (rank2 in 2:maks_rankSU)
    {x2_bar <- SU.indeks[rank2, 1, lok1, lok2]
    x2_kol <- SU.indeks[rank2, 2, lok1, lok2]
    x2 <-x2_bar+((x2_kol-1)*13)
    ifelse(abs(matriks.SU[x1,x2,lok1,lok2])>beta,SU.rank[x2_bar,x2_kol,lok1,lok2]<-
NA,SU.rank[x2_bar,x2_kol,lok1,lok2]<-1)
    }
  }
}

## Mengubah kode SU.Status
SU.rank1 <-ifelse(SU.rank!=0,1,0)
SU.rank1[is.na(SU.rank1)]=0
SU.Status <-SU.rank1

## buat array time lags kembali untuk antar.lok.SU
antar.lok.SU<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3, 3),dimnames=NULL)

antar.lok.SU <- create_feature_predictor_nonLinear(SU.Status)
merit.SU <- compute_matrix_SU_for_SU(antar.lok.SU)
merit.value <- compute_Matrix_Merit_Value(SU.Status, merit.SU)
trace.merit.value <- abs(trace(merit.value))
if (trace1.merit.value < trace.merit.value)
{trace1.merit.value<-trace.merit.value
  SU.Status.opt<-SU.Status
  beta.opt <- beta
} else {
  trace1.merit.value
  SU.Status.opt
  beta}
beta<-beta+delta
}
jum.pred.opt <- 0
for (lok2 in 1:3)
{
  for (lok1 in 1:3)
  {
    jum.pred.opt <- jum.pred.opt + length(which(SU.Status.opt[ , ,lok1,lok2]== 1))-1
  }
}
SU.Status.opt[1,1, , ] <- 0
return(remove_nonLin <<-list(trace1.merit.value, beta.opt, SU.Status.opt, jum.pred.opt, matriks.merit))
}

## Generate data berdasar PC_SU.Status.opt menggunakan RNN (Model 2, 4, 5)
## JANGAN LUPA assign parameter dari list data remove_lin/non_lin
Generate_data_as_RNN_input_model245 <- function(PC_SU.Status.opt)
{
  antar.lok.PC_SU.opt <- Create_feature_predictor_linear(PC_SU.Status.opt)

  antar_lok.RNN.11 <<- antar.lok.PC_SU.opt[ , ,1,1]
  antar_lok.RNN.21 <<- antar.lok.PC_SU.opt[ , ,2,1]
  antar_lok.RNN.31 <<- antar.lok.PC_SU.opt[ , ,3,1]

  antar_lok.RNN.12 <<- antar.lok.PC_SU.opt[ , ,1,2]
  antar_lok.RNN.22 <<- antar.lok.PC_SU.opt[ , ,2,2]

```

```

antar_lok.RNN.32 <<- antar.lok.PC_SU.opt[ , ,3,2]

antar_lok.RNN.13 <<- antar.lok.PC_SU.opt[ , ,1,3]
antar_lok.RNN.23 <<- antar.lok.PC_SU.opt[ , ,2,3]
antar_lok.RNN.33 <<- antar.lok.PC_SU.opt[ , ,3,3]

antar_lok.RNN.11 <- antar_lok.RNN.11[, apply(antar_lok.RNN.11, 2, function(x) !all(is.na(x)))]
antar_lok.RNN.21 <- antar_lok.RNN.21[, apply(antar_lok.RNN.21, 2, function(x) !all(is.na(x)))]
antar_lok.RNN.31 <- antar_lok.RNN.31[, apply(antar_lok.RNN.31, 2, function(x) !all(is.na(x)))]

antar_lok.RNN.12 <- antar_lok.RNN.12[, apply(antar_lok.RNN.12, 2, function(x) !all(is.na(x)))]
antar_lok.RNN.22 <- antar_lok.RNN.22[, apply(antar_lok.RNN.22, 2, function(x) !all(is.na(x)))]
antar_lok.RNN.32 <- antar_lok.RNN.32[, apply(antar_lok.RNN.32, 2, function(x) !all(is.na(x)))]

antar_lok.RNN.13 <- antar_lok.RNN.13[, apply(antar_lok.RNN.13, 2, function(x) !all(is.na(x)))]
antar_lok.RNN.23 <- antar_lok.RNN.23[, apply(antar_lok.RNN.23, 2, function(x) !all(is.na(x)))]
antar_lok.RNN.33 <- antar_lok.RNN.33[, apply(antar_lok.RNN.33, 2, function(x) !all(is.na(x)))]

opt.3juanda <<- data.frame(cbind(antar_lok.RNN.11, antar_lok.RNN.21, antar_lok.RNN.31))
opt.3perak1 <<- data.frame(cbind(antar_lok.RNN.12, antar_lok.RNN.22, antar_lok.RNN.32))
opt.3perak2 <<- data.frame(cbind(antar_lok.RNN.13, antar_lok.RNN.23, antar_lok.RNN.33))

opt.3juanda[ ,1] <- juanda_M[ ,1]
opt.3perak1[ ,1] <- perak1_M[ ,1]
opt.3perak2[ ,1] <- perak2_M[ ,1]

write.csv(opt.3juanda, file="opt.3juanda-model2-rank10.csv")
write.csv(opt.3perak1, file="opt.3perak1-model2-rank10.csv")
write.csv(opt.3perak2, file="opt.3perak2-model2-rank10.csv")
}

## Generate data hanya berdasar DPCCA (Model 3 : DPCCA + RNN)
Generate_data_as_RNN_input_based_on_DPCCA <- function(rho_M.status)
{
  antar.lok <- Create_feature_predictor_linear(rho_M.status)
  antar_lok.RNN.11 <<- antar.lok[ , ,1,1]
  antar_lok.RNN.21 <<- antar.lok[ , ,2,1]
  antar_lok.RNN.31 <<- antar.lok[ , ,3,1]

  antar_lok.RNN.12 <<- antar.lok[ , ,1,2]
  antar_lok.RNN.22 <<- antar.lok[ , ,2,2]
  antar_lok.RNN.23 <<- antar.lok[ , ,3,2]

  antar_lok.RNN.31 <<- antar.lok[ , ,1,3]
  antar_lok.RNN.32 <<- antar.lok[ , ,2,3]
  antar_lok.RNN.33 <<- antar.lok[ , ,3,3]

  antar_lok.RNN.11 <- antar_lok.RNN.11[, apply(antar_lok.RNN.11, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.21 <- antar_lok.RNN.21[, apply(antar_lok.RNN.21, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.31 <- antar_lok.RNN.31[, apply(antar_lok.RNN.31, 2, function(x) !all(is.na(x)))]

  antar_lok.RNN.12 <- antar_lok.RNN.12[, apply(antar_lok.RNN.12, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.22 <- antar_lok.RNN.22[, apply(antar_lok.RNN.22, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.32 <- antar_lok.RNN.32[, apply(antar_lok.RNN.32, 2, function(x) !all(is.na(x)))]

  antar_lok.RNN.13 <- antar_lok.RNN.13[, apply(antar_lok.RNN.13, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.23 <- antar_lok.RNN.23[, apply(antar_lok.RNN.23, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.33 <- antar_lok.RNN.33[, apply(antar_lok.RNN.33, 2, function(x) !all(is.na(x)))]

  opt.3juanda <<- data.frame(cbind(antar_lok.RNN.11, antar_lok.RNN.21, antar_lok.RNN.31))
  opt.3perak1 <<- data.frame(cbind(antar_lok.RNN.12, antar_lok.RNN.22, antar_lok.RNN.32))
  opt.3perak2 <<- data.frame(cbind(antar_lok.RNN.13, antar_lok.RNN.23, antar_lok.RNN.33))
}

```

```

opt.3juanda[,1] <- juanda_M[,1]
opt.3perak1[,1] <- perak1_M[,1]
opt.3perak2[,1] <- perak2_M[,1]

write.csv(opt.3juanda, file="opt.3juanda-model3REV.csv")
write.csv(opt.3perak1, file="opt.3perak1-model3REV.csv")
write.csv(opt.3perak2, file="opt.3perak2-model3REV.csv")
}

##PEMBENTUKAN FEATURE TIME-LAGS-hanya berdasar DPCCA (MDOEL 3:tanpa PCSUF)
Create_feature_based_on_DPCCA <- function(rho_M.status)
{
  antar.lok <- Create_feature_predictor_linear(rho_M.status)
  for (lok1 in 1:3)
  {
    for(lok2 in 1:3)
    {antar.lok[,1,lok1, lok2] <-gab_3lok[,1,lok1]}
  }
}

##Perhitungan merit-value untuk input berdasar DPCCA
merit.SU <- compute_matrix_SU_for_SU(antar.lok)
merit.value <- compute_Matrix_Merit_Value(rho_M.status, merit.SU)
trace.merit.value <-trace(merit.value)
trace.merit.value <-abs(trace.merit.value)
return(trace.merit.value)
}

## model 6
create_feature_without_remove_redundant<-function()
{
  PC_SU.Status<-array(data=NA, dim=c(nrow(PC.Status) , ncol(PC.Status),3, 3),dimnames=NULL)
  PC_SU.Status <- PC.Status + SU.Status
  PC_SU.Status[1, , , ] <- 0
  antar.lok.PC_SU<-array(data=NA, dim=c(nrow(gab_3lok) , ncol=156,3, 3),dimnames=NULL)
  antar.lok.PC_SU <- Create_feature_predictor_linear(PC_SU.Status)
  merit.SU <- compute_matrix_SU_for_PC(antar.lok.PC_SU)
  merit.value <- compute_Matrix_Merit_Value(PC_SU.Status, merit.SU)
  antar.lok.PC_SU.opt <- Create_feature_predictor_linear(PC_SU.Status)

  antar_lok.RNN.11 <<- antar.lok[, ,1,1]
  antar_lok.RNN.21 <<- antar.lok[, ,2,1]
  antar_lok.RNN.31 <<- antar.lok[, ,3,1]

  antar_lok.RNN.21 <<- antar.lok[, ,1,2]
  antar_lok.RNN.22 <<- antar.lok[, ,2,2]
  antar_lok.RNN.23 <<- antar.lok[, ,3,2]

  antar_lok.RNN.31 <<- antar.lok[, ,1,3]
  antar_lok.RNN.32 <<- antar.lok[, ,2,3]
  antar_lok.RNN.33 <<- antar.lok[, ,3,3]

  antar_lok.RNN.11 <- antar_lok.RNN.11[, apply(antar_lok.RNN.11, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.12 <- antar_lok.RNN.12[, apply(antar_lok.RNN.12, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.13 <- antar_lok.RNN.13[, apply(antar_lok.RNN.13, 2, function(x) !all(is.na(x)))]

  antar_lok.RNN.21 <- antar_lok.RNN.21[, apply(antar_lok.RNN.21, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.22 <- antar_lok.RNN.22[, apply(antar_lok.RNN.22, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.23 <- antar_lok.RNN.23[, apply(antar_lok.RNN.32, 2, function(x) !all(is.na(x)))]

  antar_lok.RNN.31 <- antar_lok.RNN.31[, apply(antar_lok.RNN.31, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.32 <- antar_lok.RNN.32[, apply(antar_lok.RNN.32, 2, function(x) !all(is.na(x)))]
  antar_lok.RNN.33 <- antar_lok.RNN.33[, apply(antar_lok.RNN.33, 2, function(x) !all(is.na(x)))]
}

```



```
opt.3juanda <- data.frame(cbind(antar_lok.RNN.11, antar_lok.RNN.12, antar_lok.RNN.13))
opt.3perak1 <- data.frame(cbind(antar_lok.RNN.21, antar_lok.RNN.22, antar_lok.RNN.23))
opt.3perak2 <- data.frame(cbind(antar_lok.RNN.31, antar_lok.RNN.32, antar_lok.RNN.33))

opt.3juanda[,1] <- juanda_M[,1]
opt.3perak1[,1] <- perak1_M[,1]
opt.3perak2[,1] <- perak2_M[,1]

write.csv(opt.3juanda, file="opt.3juanda-model6Rev.csv")
write.csv(opt.3perak1, file="opt.3perak1-model6Rev.csv")
write.csv(opt.3perak2, file="opt.3perak2-model6Rev.csv")
}
```

II.D. Generalized Spatio-Temporal Autoregressive (GSTAR)

(Modifikasi dari Sumber [138] : R Software for Parameter Estimation of Spatio Temporal Model)

```
GSTAR<-function(Data,W,p){
  Series<-ts(Data)
  k<-ncol(Data) # Number Location
  n<-nrow(Data) # Number Time
  zt<-(stack(as.data.frame(t(Data)))[,1]) # Zt with Lag

  # Function Create Lag Data
  #####
  MD<-function(zt){
    M<-matrix(0,(n*k),k)
    z=0
    for( i in 1:(n)){
      for(j in 1:k){
        z<-z+1
        M[z,j]<-zt[z]
      }
    }
    M
  }
  #####
  M1<-MD(zt)
  MA<-matrix(0,(n*k-k*p),k*p)
  MAW<-matrix(0,(n*k-k*p),k*p)
  W1<-kronecker(diag(n), W)
  ztw<-W1%*%zt
  M2<-MD(ztw)
  zt<-as.matrix(zt)
  ZT<-zt[-(1:(k*p)),]
  for( i in 1:p){
    MA[(1:(n*k-k*p)),(k*(i-1)+1):(k*((i-1)+1))] <- M1[(k*(p-i)+1):(n*k-(k*i)),]
    MAW[(1:(n*k-k*p)),(k*(i-1)+1):(k*((i-1)+1))] <- M2[(k*(p-i)+1):(n*k-(k*i)),]
  }
  XT<-MA
  WXT<-MAW
  GSTAR<-data.frame(ZT,XT,WXT)
  GSTARfit<-lm(ZT~.-1,data=GSTAR)
  fit <-summary(GSTARfit)
  # Function Create Lag Data_diff
  #####
  MD<-function(zt){
    M<-matrix(0,(n*k),k)
    z=0
    for( i in 1:(n)){
      for(j in 1:k){
        z<-z+1
        M[z,j]<-zt[z]
      }
    }
    M
  }
  #####
  M1<-MD(zt)
  MA<-matrix(0,(n*k-k*p),k*p)
  MAW<-matrix(0,(n*k-k*p),k*p)
  W1<-kronecker(diag(n), W)
  ztw<-W1%*%zt
  M2<-MD(ztw)
  zt<-as.matrix(zt)
  ZT<-zt[-(1:(k*p)),]
```

```

for (i in 1:p){
  MA[(1:(n*k-k*p)),(k*(i-1)+1):(k*((i-1)+1))] <- M1[(k*(p-i)+1):(n*k-(k*i)),]
  MAW[(1:(n*k-k*p)),(k*(i-1)+1):(k*((i-1)+1))] <- M2[(k*(p-i)+1):(n*k-(k*i)),]
}
XT<-MA
WXT<-MAW
GSTAR<-data.frame(ZT,XT,WXT)
GSTARfit<-lm(ZT~.-1,data=GSTAR)
MSE<-(fit$sigma)^2
MAE<-sum(abs(GSTARfit$residuals))/(n*k-k*p)
MAPE<-sum(abs(GSTARfit$residuals)/ZT)/(n*k-k*p)
MADP<-sum(abs(GSTARfit$residuals))/sum(ZT)
AIC<-AIC(GSTARfit)
R2<-fit$r.squared
R2a<-fit$adj.r.squared
Coef<-GSTARfit$coefficients
Error<-data.frame(MSE=MSE, MAE=MAE, MAPE=MAPE,
  MADP=MADP, AIC=AIC, Rsquare=R2, AdjRSquare=R2a)
Coefficient<-data.frame(Coef)
Zt<-as.data.frame(matrix(ZT,n-p,k,byrow=T))
Zhat1<-as.data.frame(matrix(GSTARfit$fitted.values,(n-p),k,byrow = T))
Residual<-as.data.frame(matrix(GSTARfit$residuals,(n-p),k,byrow = T))
Result<-data.frame(Zt,Zhat1, Residual)
Result<-ts(Result)
#windows()
#plot(Series,plot.type="single", lty=1:3, col = 4:2,xlab="Time",main="Data Series Plot")
#windows()
#plot(Result,plot.type="multiple", lty=1:3, col = 4:2,main="Predictive vs Observed")
#windows()
#plot(Result,plot.type="single", lty=1:3, col = 4:2,main="Predictive vs Observed")
Residual<-ts(Residual)
#windows()
#plot(Residual,plot.type="single", lty=1:3, col =4:2,main="Residual")
Summary<-list(GOF=Error,Fit=fit)
return(Summary)
}

#' Select the Best p
#' @param p
#' @return p
#' @export

Model<-function(p){
  Error<-matrix(0,p,7)
  for (i in 1:p){
    Error[i,]<-as.matrix(GSTAR(Data, W,i)[[1]])
  }
  colnames(Error)<-c("MSE", "MAE", "MAPE", "MADP","AIC", "Rsquare", "AdjRSquare")
  Result<-list(Error=Error)
  return(Result)
}

#### FUNGSI GSTAR UNTUK KASUS 3 LOKASI STASIUN CUACA
Data_Tot <- read.csv("Rainfall3Lokasi-GSTAR.txt", header=TRUE)
Data_Aslis <- as.matrix(Data_Tot)
Data_Tot <- as.matrix(Data_Tot)
dif <- 12
Data_Tot <- as.matrix(diff(Data_Tot, lag=dif, differences=1))
n_train <- floor(nrow(Data_Aslis)*0.75)
Data <- as.matrix(Data_Tot[1:(n_train-dif), ])

### Stasioneritas dan Diagnosa model
library(tseries)
for (lok in 1:3)

```

```

{
  ADF_data_asli <- adf.test(Data_Asli[,lok], k=24)
  Test_ADF <- adf.test(Data[,lok], k=24)
  print(ADF_data_asli)
  print(Test_ADF)
  AutoCorrelation <- acf(Data[,lok], plot = FALSE)
  plot(AutoCorrelation, main = "Curah Hujan di 3 lokasi Stasiun Cuaca", lwd=3, ci.type = "ma")
  Partial_acf <- pacf(Data[,lok], plot=FALSE)
  plot(Partial_acf, main = "CUran Hujan di 3 lokasi Stasiun Cuaca", lwd=3)
}
W<-matrix(c(0,0.5,0.5,0.5,0,0.5,0.5,0.5,0),3,3)

## Perhitungan weight berdasar jarak (inverse distance weight)
r1 <- 17.79
r2 <- 19.49
r3 <- 2.0

W <- matrix(c(0,r2/(r1+r2), r1/(r1+r2), r3/(r1+r3), 0, r1/(r1+r3), r3/(r2+r1), r2/(r2+r3),0),3,3)

## Perhitungan weight berdasar normalisasi korelasi
r12 <- 0.6896245
r13 <- 0.7749152
r23 <- 0.8109261

W <- matrix(c(0,r12/(r12+r13), r13/(r12+r13), r12/(r12+r13), 0, r23/(r12+r23), r13/(r13+r12),
r23/(r12+r23),0),3,3)

p <- 24
gstar <- GSTAR(Data, W, p)

Train_Result_temp <- array(data=NA, dim=c(nrow(Result), 3, 3), dimnames=NULL)
Train_Result_GSTAR <- array(data=NA, dim=c(n_train, 3, 3), dimnames=NULL)
for (lok in 1:3){
  for (col in 1:3){
    Train_Result_temp[,col,lok] <- Result[, (lok+(col-1)*3)]
  }
}
## result adalah hasil training gstar berdimensi jumlah_train x 9
## kolom 1-3 = data, kolom 4-6=prediksi, kol 7-9= error

### Tambah Data unt Initial
Tambah <- array(data=NA, dim=c(dif+p, 3, 3), dimnames=NULL)
for (lok in 1:3){
  Tambah[, 1,lok] <- Data_Asli[1:(dif+p),lok]
  Tambah[, 2,lok] <- Data_Asli[1:(dif+p),lok]
}

## Invers Difference dari Train_Result_GSTAR
for (lok in 1:3) {
  Train_Result_GSTAR[(p+1):n_train,1,lok]<-diffinv(Train_Result_temp[,1,lok], lag=dif, differences = 1,
xi=Tambah[(p+1):(dif+p),1,lok])
  Train_Result_GSTAR[1:p,1,lok] <- Tambah[1:p,1,lok]
  Train_Result_GSTAR[(p+1):n_train,2,lok]<-diffinv(Train_Result_temp[,2,lok], lag=dif, differences = 1,
xi=Tambah[(p+1):(dif+p),2,lok])
  Train_Result_GSTAR[1:p,2,lok] <- Tambah[1:p,2,lok]
  Train_Result_GSTAR[,3,lok] <- Train_Result_GSTAR[,1,lok] - Train_Result_GSTAR[,2,lok]
}

## Train_Result_GSTAR adalah set training yang lengkap

#Menyiapkan Data Testing
n_validation <- nrow(Data_Asli)-n_train
Data_Test <- as.matrix(Data_Tot[(n_train+1):nrow(Data_Tot), ])

```

```

## transformasi koef model ke matrix
Coefficient <- as.data.frame(Coefficient)
Coef_par <- array(data=NA, dim=c(3, 2, p), dimnames=NULL)
Matrix_par1 <- array(data=NA, dim=c(3, 3, p), dimnames=NULL)
Matrix_par <- array(data=NA, dim=c(3, 3, p, 2), dimnames=NULL)
for (kol in 1:2)
{
  for (i in 1:p)
  {
    Coef_par[,1,i] <- Coefficient[((i-1)*3+1):(i*3),1]
    Coef_par[,2,i] <- Coefficient[((i-1)*3+(3*p+1)):(i*3+3*p),1]
    for (j in 1:3)
    {
      Matrix_par1[j, j, i] <- Coef_par[j,kol, i]
    }
  }
  Matrix_par[, , kol] <- Matrix_par1
}
Matrix_par[is.na(Matrix_par)]=0

##Proses prediksi 1 step ke depan
Data_prediksi <- matrix(data=NA, (n_validation-dif), 3, dimnames=NULL)
for (i in (p+1):(n_validation-dif))
{
  temp <- matrix(data=0, 3, 1)
  for (j in 1:p)
  {
    temp <- temp + Matrix_par[, , j, 1]%% t(Data_Test[i-j, ]) + Matrix_par[, , j, 2]%% W %%
    t(Data_Test[i-j, ])
  }
  Data_prediksi[i, ]<-t(temp)
}

## Data prediksi 1 step, periode 1 s/d p
for (i in 1:p)
{
  Data_prediksi[i, ]<- Data_Tot[n_train+i, ]
}

##Data_prediksi[is.na(Data_prediksi)]=0

## MELAKUKAN INVERS DIFF dATuntuk perhitungan kinerja
Valid_Result_GSTAR <- array(data=NA, dim=c(n_validation, 3, 3), dimnames=NULL)
for (lok in 1:3)
{
  Valid_Result_GSTAR[,1,lok]<-diffinv(Data_Test[, ,lok], lag=dif, differences = 1,
xi=Data_Asl[(n_train+1):(n_train+dif),lok])
  Valid_Result_GSTAR[,2,lok]<-diffinv(Data_prediksi[, ,lok], lag=dif, differences = 1,
xi=Data_Asl[(n_train+1):(n_train+dif),lok])
  Valid_Result_GSTAR[,3,lok] <- Valid_Result_GSTAR[,1,lok]-Valid_Result_GSTAR[,2,lok]
}

## Perhitungan Kinerja Training dan Validasi per lokasi KESELURUHAN DATA
korelasi <- matrix(data=NA, 3, 2)
R_sq <- matrix(data=NA, 3, 2)
RMSE <- matrix(data=NA, 3, 2)

for (lok in 1:3)
{korelasi[lok, 1] <- cor(Train_Result_GSTAR[,1,lok], Train_Result_GSTAR[,2,lok])
R_sq[lok, 1] <- (korelasi[lok,1])^2
}

```

```

RMSE[lok, 1] <- RootMSE(Train_Result_GSTAR[,1,lok], Train_Result_GSTAR[,2, lok])
}
for (lok in 1:3)
{
  korelasi[lok, 2] <- cor(Valid_Result_GSTAR[(p+dif+1):n_validation,1, lok],
Valid_Result_GSTAR[(p+dif+1):n_validation,2,lok])
  R_sq[lok, 2] <- (korelasi[lok,2])^2
  RMSE[lok, 2] <- RootMSE(Valid_Result_GSTAR[(p+dif+1):n_validation,1, lok],
Valid_Result_GSTAR[(p+dif+1):n_validation ,2, lok])
}

## Perhitungan Kinerja Gabungan
correlation <- matrix(data=NA, 1, 2)
RMSE_total<- matrix(data=NA, 1, 2)
R_sq_total <- matrix(data=NA, 1, 2)

for (kol in 1:2)
{
  correlation [1, kol]<- mean(korelasi[,kol])
  R_sq_total[1, kol] <- mean(R_sq[, kol])
  RMSE_total[1, kol] <- sum(RMSE[, kol])
}

## Perhitungan AIC correction
## fungsi AIC correction
AICC = function(MSE, n, p)
{
  n*log(MSE)+2*(p+1) + 2*(p+1)*(p+2)/(n-p)
}

AIC_correction <-matrix(data=NA, 3,1)
N <- n_train
for (lok in 1:3)
{
  MSE_value = (RootMSE(Train_Result_GSTAR[,1,lok], Train_Result_GSTAR[,2,lok]))^2
  AIC_correction[lok,1] <- AICC(MSE_value,N, p)
}
AIC_correction <- (AIC_correction[1,1]+AIC_correction[2,1]+AIC_correction[3,1])/3
print(correlation)
print(R_sq_total)
print(RMSE_total)
print(AIC_correction)

#### Plot Aktual vs Training Dari Data Differencing
for (lok in 1:3)
{
  plot(Train_Result_GSTAR[(p+1):n_train,1, lok], col="blue", type="l",main =c("Training GSTAR: Data
Aktual vs Prediksi", lok), xlab="bulan ke-", ylab="curah hujan (mm)", lwd = 2)
  lines(Train_Result_GSTAR[(p+1):n_train,2,lok], type = "l", col = "red", lwd = 1)
  legend("topleft", legend=c("aktual", "prediksi"),
        col=c("blue","red"),lty=1:0.5, cex=0.6, box.lty=1, box.lwd=0.5)
}

#### Plot Aktual vs Testing
for (lok in 1:3)
{
  plot(Valid_Result_GSTAR[(p+dif+1):n_validation,1, lok], col="blue", type="l",main =c("Testing GSTAR:
Data Aktual vs Prediksi", lok), xlab="bulan ke-", ylab="curah hujan (mm)", lwd = 2)
  lines(Valid_Result_GSTAR[(p+dif+1):n_validation,2,lok], type = "l", col = "red", lwd = 1)
  legend("topleft", legend=c("aktual", "prediksi"),
        col=c("blue","red"),lty=1:0.5, cex=0.6, box.lty=1, box.lwd=0.5)
}

```

```
## Analisis Residual
for (lok in 1:3)
{
AutoCorrelation <- acf(Train_Result_temp[,3,lok], plot = FALSE, lag.max = 24)
plot(AutoCorrelation, main = c("Error dari Model GSTAR", lok), lwd=3, ci.type = "ma")
}
```

II.E. Main-Program Hybrid GSTAR + ANN

```
## LOAD DATA
juanda <- read.csv("juanda.csv", header=TRUE)
juanda_M<-as.matrix(juanda)
perak1 <- read.csv("perak1.csv", header=TRUE)
perak1_M<-as.matrix(perak1)
perak2 <- read.csv("perak2.csv", header=TRUE)
perak2_M<-as.matrix(perak2)
gab_3lok <- array( c( juanda_M , perak1_M, perak2_M) , dim = c( nrow(juanda_M) , ncol(juanda_M), 3 ) )

saveRDS(rho_M.status, file="rho_M.status.hybrid.rda")
rho_M.status <- readRDS(file= "rho_M.status.hybrid.rda")

library(FSelector)

## pilih train_set GSTAR dan menyimpan error GSTAR di Train_Result_GSTAR1
gab_3lok_full <- gab_3lok
train_set <- floor(nrow(gab_3lok_full)*0.75)
gab_3lok <- gab_3lok_full[1:train_set, , ]
for (lok in 1:3){
  gab_3lok[,1, lok] <- Train_Result_GSTAR1[,3, lok]
}

## FUNgsi menggabungkan error train dan validasi dari GSTAR setelah dpcca di gab_3lok
gab_3lok <- gab_3lok_full
for (lok in 1:3){
  gab_3lok[1:train_set,1,lok] <- Train_Result_GSTAR1[,3, lok]
  gab_3lok[(train_set+1):nrow(gab_3lok_full),1, lok] <- Valid_Result_GSTAR1[,3,lok]
}

##### MODEL 8: hybrid lengkap #####
rho_M.status <- DPCCA_extract(gab_3lok, 7)
PC.Status <-Select_Linear_Relevant(gab_3lok)
SU.Status <-Select_NonLinear_Relevant(gab_3lok)
antar.lok.PC <- Create_feature_predictor_linear(PC.Status)
antar.lok.SU <- create_feature_predictor_nonLinear(SU.Status)
PC.indeks <- create_index_rank_linear(60)
SU.indeks <- create_index_rank_non_linear(30)
rem_lin_NonLin <- remove_redundant_linear_and_nonLinear(30)
Generate_data_as_RNN_input(PC_SU.Status.opt)
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)
Compute_Model_Performance_GSTAR_RNN()
#####

##### MODEL 9 : hybrid : DPCCA + ANN #####
rho_M.status <- DPCCA_extract(gab_3lok, 7)
merit.value <- Create_feature_based_on_DPCCA(rho_M.status)
Generate_data_as_RNN_input_based_on_DPCCA(rho_M.status)
Compute_Model_Performance_GSTAR_RNN()
#####

##### MODEL 10 : hybrid : PCF + ANN #####
rho_M.status <- DPCCA_extract(gab_3lok, 7)
PC.Status <-Select_Linear_Relevant(gab_3lok)
antar.lok.PC <- Create_feature_predictor_linear(PC.Status)
PC.indeks <- create_index_rank_linear(60)
rem_linear <- remove_linear_redundant(60)
antar.lok.RNN <- Generate_data_as_RNN_input(PC.Status.opt)
ANN_Spatio_Temporal
Compute_Model_Performance_GSTAR_RNN()
#####
```



```
##### MODEL 11 : hybrid : SUF + ANN #####  
rho_M.status <- DPCCA_extract(gab_3lok, 7)  
SU.Status <- Select_NonLinear_Relevant(gab_3lok)  
antar.lok.SU <- create_feature_predictor_nonLinear(SU.Status)  
SU.indeks <- create_index_rank_non_linear(5)  
rem_nonLin <- remove_non_linear_redundant(30)  
Generate_data_as_RNN_input(SU.Status.opt)  
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)  
#####
```

```
##### MODEL 12 : hybrid : DPCCA + PCSU without remove + ANN #####  
rho.M.status <- DPCCA_extract(gab_3lok, 5)  
merit.value <- create_feature_without_remove_redundant()  
Generate_data_as_RNN_input()  
Performance_RNN <- RNN_spatio-temporal(antar.lok.RNN)  
#####
```

LAMPIRAN III : OUTPUT PROGAM STUDI KASUS 1

```

> rho_M
, , 1, 1
    var 1 var 2 var 3 var 4 var 5 var 6 var 7 var 8 var 9 var 10
lag ke 1  1.000 0.479 0.063 -0.108 0.788 -0.744 -0.290 0.430 -0.070 -0.062
lag ke 2  0.669 0.492 0.421 0.214 0.481 -0.689 -0.298 0.315 -0.078 -0.053
lag ke 3  0.231 0.408 0.679 0.457 0.019 -0.434 -0.230 0.160 -0.067 -0.027
lag ke 4 -0.133 0.158 0.681 0.435 -0.404 -0.079 -0.028 0.001 -0.041 0.003
lag ke 5 -0.406 -0.207 0.404 0.159 -0.628 0.257 0.209 -0.082 -0.007 0.037
lag ke 6 -0.558 -0.530 -0.003 -0.139 -0.642 0.502 0.348 -0.186 0.052 0.069
lag ke 7 -0.595 -0.651 -0.303 -0.277 -0.506 0.606 0.336 -0.298 0.107 0.103
lag ke 8 -0.510 -0.523 -0.448 -0.226 -0.322 0.596 0.249 -0.338 0.133 0.127
lag ke 9 -0.320 -0.250 -0.482 -0.112 -0.079 0.456 0.092 -0.298 0.113 0.129
lag ke 10 -0.053 0.038 -0.429 -0.058 0.213 0.187 -0.075 -0.136 0.085 0.091
lag ke 11 0.238 0.247 -0.335 -0.109 0.466 -0.146 -0.158 0.125 0.046 0.037
lag ke 12 0.532 0.383 -0.165 -0.156 0.636 -0.481 -0.153 0.332 0.000 -0.025
lag ke 13 0.703 0.447 0.073 -0.053 0.639 -0.683 -0.214 0.399 -0.060 -0.098
    var 11 var 12
lag ke 1 -0.066 -0.077
lag ke 2 -0.023 -0.029
lag ke 3 0.024 0.029
lag ke 4 0.056 0.070
lag ke 5 0.068 0.074
lag ke 6 0.075 0.068
lag ke 7 0.076 0.038
lag ke 8 0.073 -0.010
lag ke 9 0.074 -0.026
lag ke 10 0.055 -0.025
lag ke 11 0.019 -0.011
lag ke 12 -0.035 -0.010
lag ke 13 -0.096 -0.036
, , 2, 1
    var 1 var 2 var 3 var 4 var 5 var 6 var 7 var 8 var 9 var 10
lag ke 1  0.747 0.305 0.157 -0.165 0.527 -0.648 -0.145 0.404 -0.083 -0.089
lag ke 2  0.451 0.230 0.354 0.019 0.258 -0.521 -0.127 0.266 -0.077 -0.089
lag ke 3  0.061 0.198 0.508 0.278 -0.126 -0.265 -0.077 0.132 -0.069 -0.074
lag ke 4 -0.307 0.031 0.505 0.408 -0.492 0.118 0.002 -0.136 -0.047 -0.045
lag ke 5 -0.549 -0.263 0.276 0.207 -0.675 0.438 0.149 -0.258 -0.009 0.002
lag ke 6 -0.591 -0.499 -0.115 -0.125 -0.596 0.586 0.333 -0.269 0.053 0.048
lag ke 7 -0.459 -0.597 -0.447 -0.353 -0.318 0.557 0.323 -0.231 0.102 0.074
lag ke 8 -0.260 -0.424 -0.568 -0.346 -0.043 0.436 0.205 -0.176 0.122 0.092
lag ke 9 -0.020 -0.099 -0.470 -0.135 0.208 0.209 -0.069 -0.166 0.084 0.103
lag ke 10 0.192 0.274 -0.240 0.061 0.390 -0.101 -0.264 -0.076 0.036 0.072
lag ke 11 0.361 0.447 -0.053 0.116 0.451 -0.314 -0.258 0.134 0.012 0.038
lag ke 12 0.491 0.444 0.076 0.010 0.463 -0.497 -0.118 0.356 -0.001 0.002
lag ke 13 0.539 0.330 0.177 -0.069 0.423 -0.580 -0.098 0.421 -0.031 -0.039
    var 11 var 12
lag ke 1 -0.095 -0.061
lag ke 2 -0.077 -0.058
lag ke 3 -0.034 -0.007
lag ke 4 0.007 0.054
lag ke 5 0.036 0.076
lag ke 6 0.049 0.066
lag ke 7 0.049 0.035
lag ke 8 0.049 -0.016
lag ke 9 0.066 -0.021
lag ke 10 0.058 0.003
lag ke 11 0.034 0.002
lag ke 12 -0.011 -0.016
lag ke 13 -0.043 -0.019
.....

```

> PC_SU.Status.opt

,, 1, 1

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]
[1,]	0	0	0	0	0	0	0	0	0	0	0	0
[2,]	2	2	2	2	2	2	1	1	0	0	0	0
[3,]	1	2	2	2	0	1	0	0	0	0	0	0
[4,]	0	0	2	1	1	0	0	0	0	0	0	0
[5,]	1	0	1	0	2	0	0	0	0	0	0	0
[6,]	2	1	0	0	2	2	0	0	0	0	0	0
[7,]	2	2	1	0	2	2	0	1	0	0	0	0
[8,]	2	2	2	0	2	2	0	1	0	0	0	0
[9,]	1	1	2	0	0	2	0	1	0	0	0	0
[10,]	0	0	2	0	0	0	0	0	0	0	0	0
[11,]	1	0	1	1	2	0	0	0	0	0	0	0
[12,]	2	1	0	0	2	2	0	1	0	0	0	0
[13,]	2	2	0	0	2	2	0	1	0	0	0	0

,, 2, 1

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]
[1,]	0	0	0	0	0	0	0	0	0	0	0	0
[2,]	2	1	2	0	0	2	0	0	0	0	0	0
[3,]	0	1	2	1	0	1	0	0	0	0	0	0
[4,]	1	0	2	1	2	0	0	0	0	0	0	0
[5,]	2	1	1	0	2	2	0	1	0	0	0	0
[6,]	2	2	0	0	2	2	1	1	0	0	0	0
[7,]	2	2	2	1	1	2	0	1	0	0	0	0
[8,]	1	1	2	1	0	2	0	0	0	0	0	0
[9,]	0	0	2	0	0	2	0	0	0	0	0	0
[10,]	0	1	1	1	1	0	0	0	0	0	0	0
[11,]	2	2	0	0	2	2	0	0	0	0	0	0
[12,]	2	2	0	0	2	2	0	1	0	0	0	0
[13,]	2	2	0	0	2	2	0	1	0	0	0	0

,, 3, 1

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]	[,12]
[1,]	0	0	0	0	0	0	0	0	0	0	0	0
[2,]	2	2	2	2	2	2	0	0	0	0	0	0
[3,]	0	1	2	1	0	1	0	0	0	0	0	0
[4,]	0	0	2	1	2	0	0	0	0	0	0	0
[5,]	2	1	1	0	2	1	0	0	0	0	0	0
[6,]	2	2	0	0	2	2	1	1	0	0	0	0
[7,]	2	2	2	0	2	2	0	1	0	0	0	0
[8,]	2	1	2	0	0	2	0	1	0	0	0	0
[9,]	0	0	2	0	0	2	0	1	0	0	0	0
[10,]	0	0	2	0	1	0	0	0	0	0	0	0
[11,]	2	1	1	0	2	1	0	0	0	0	0	0
[12,]	2	2	0	0	2	2	0	0	0	0	0	0
[13,]	2	2	0	0	2	2	0	1	0	0	0	0

.....

```
> PC.Status.opt
```

```
., 1, 1
```

```
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]  
[1,]  1  0  0  0  1  1  0  0  0  0  0  0  
[2,]  1  0  0  0  0  1  0  0  0  0  0  0  
[3,]  0  0  1  0  0  0  0  0  0  0  0  0  
[4,]  0  0  1  0  0  0  0  0  0  0  0  0  
[5,]  0  0  0  0  0  0  0  0  0  0  0  0  
[6,]  0  0  0  0  0  0  0  0  0  0  0  0  
[7,]  1  1  0  0  0  1  0  0  0  0  0  0  
[8,]  0  0  0  0  0  1  0  0  0  0  0  0  
[9,]  0  0  0  0  0  0  0  0  0  0  0  0  
[10,] 0  0  0  0  0  0  0  0  0  0  0  0  
[11,] 0  0  0  0  0  0  0  0  0  0  0  0  
[12,] 0  0  0  0  1  0  0  0  0  0  0  0  
[13,] 1  0  0  0  1  1  0  0  0  0  0  0
```

```
., 2, 1
```

```
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]  
[1,]  1  0  0  0  0  1  0  0  0  0  0  0  
[2,]  0  0  0  0  0  0  0  0  0  0  0  0  
[3,]  0  0  1  0  0  0  0  0  0  0  0  0  
[4,]  0  0  0  0  0  0  0  0  0  0  0  0  
[5,]  1  0  0  0  0  0  0  0  0  0  0  0  
[6,]  0  0  0  0  0  1  0  0  0  0  0  0  
[7,]  0  0  0  0  0  1  0  0  0  0  0  0  
[8,]  0  0  0  0  0  0  0  0  0  0  0  0  
[9,]  0  0  1  0  0  0  0  0  0  0  0  0  
[10,] 0  0  0  0  0  0  0  0  0  0  0  0  
[11,] 0  0  0  0  0  0  0  0  0  0  0  0  
[12,] 1  0  0  0  0  1  0  0  0  0  0  0  
[13,] 0  0  0  0  0  1  0  0  0  0  0  0
```

```
., 3, 1
```

```
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]  
[1,]  1  0  0  0  0  1  0  0  0  0  0  0  
[2,]  0  0  0  0  0  1  0  0  0  0  0  0  
[3,]  0  0  1  0  0  0  0  0  0  0  0  0  
[4,]  0  0  0  0  0  0  0  0  0  0  0  0  
[5,]  0  0  0  0  1  0  0  0  0  0  0  0  
[6,]  1  0  0  0  1  0  0  0  0  0  0  0  
[7,]  1  0  0  0  0  1  0  0  0  0  0  0  
[8,]  0  0  0  0  0  0  0  0  0  0  0  0  
[9,]  0  0  0  0  0  0  0  0  0  0  0  0  
[10,] 0  0  0  0  0  0  0  0  0  0  0  0  
[11,] 0  0  0  0  0  0  0  0  0  0  0  0  
[12,] 1  0  0  0  1  1  0  0  0  0  0  0  
[13,] 1  0  0  0  0  1  0  0  0  0  0  0
```

```
.....
```

```

> SU.Status.opt
, , 1, 1
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]  0  0  0  0  0  0  0  0  0  0  0  0
[2,]  1  1  1  1  1  1  0  0  0  0  0  0
[3,]  0  1  1  1  0  0  0  0  0  0  0  0
[4,]  0  0  1  0  0  0  0  0  0  0  0  0
[5,]  0  0  0  0  1  0  0  0  0  0  0  0
[6,]  1  0  0  0  1  1  0  0  0  0  0  0
[7,]  1  1  0  0  1  1  0  0  0  0  0  0
[8,]  1  1  1  0  1  1  0  0  0  0  0  0
[9,]  0  0  1  0  0  1  0  0  0  0  0  0
[10,] 0  0  1  0  0  0  0  0  0  0  0  0
[11,] 0  0  0  0  1  0  0  0  0  0  0  0
[12,] 1  0  0  0  1  1  0  0  0  0  0  0
[13,] 1  1  0  0  1  1  0  0  0  0  0  0

```

```

, , 2, 1
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]  0  0  0  0  0  0  0  0  0  0  0  0
[2,]  1  0  1  0  0  1  0  0  0  0  0  0
[3,]  0  0  1  0  0  0  0  0  0  0  0  0
[4,]  0  0  1  0  1  0  0  0  0  0  0  0
[5,]  0  0  0  0  0  1  0  0  0  0  0  0
[6,]  1  1  0  0  1  1  0  0  0  0  0  0
[7,]  1  1  1  0  0  1  0  0  0  0  0  0
[8,]  0  0  1  0  0  1  0  0  0  0  0  0
[9,]  0  0  1  0  0  1  0  0  0  0  0  0
[10,] 0  0  0  1  0  0  0  0  0  0  0  0
[11,] 1  1  0  0  1  1  0  0  0  0  0  0
[12,] 1  1  0  0  1  1  0  0  0  0  0  0
[13,] 1  1  0  0  1  1  0  0  0  0  0  0

```

```

, , 3, 1
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
[1,]  0  0  0  0  0  0  0  0  0  0  0  0
[2,]  1  1  1  1  1  1  0  0  0  0  0  0
[3,]  0  0  1  0  0  0  0  0  0  0  0  0
[4,]  0  0  1  0  1  0  0  0  0  0  0  0
[5,]  1  0  0  0  1  0  0  0  0  0  0  0
[6,]  1  1  0  0  1  1  0  0  0  0  0  0
[7,]  1  1  1  0  1  1  0  0  0  0  0  0
[8,]  0  0  1  0  0  0  0  0  0  0  0  0
[9,]  0  0  1  0  0  0  0  0  0  0  0  0
[10,] 0  0  1  0  0  0  0  0  0  0  0  0
[11,] 1  0  0  0  1  0  0  0  0  0  0  0
[12,] 1  1  0  0  1  1  0  0  0  0  0  0
[13,] 1  1  0  0  1  1  0  0  0  0  0  0

```

```

.....
> rem_lin_NonLin[[1]]
[1] 0.7034765
> rem_linear[[1]]
[1] 0.5430044
> rem_nonLin[[1]]
[1] 0.8059053

```

```

#### GSTAR
> print(correlation)
      [,1] [,2]
[1,] 0.7638326 0.7278418
> print(R_sq_total)
      [,1] [,2]
[1,] 0.5861797 0.535996
> print(RMSE_total)
      [,1] [,2]
[1,] 249.9896 287.6947

### ST-RNN
> print(correlation)
      [,1] [,2]
[1,] 0.8556667 0.8453333
> print(R_sq_total)
      [,1] [,2]
[1,] 0.7322463 0.7151153
> print(RMSE_total)
      [,1] [,2]
[1,] 211.1341 279.1012
> print(AIC_correction_Tot)
[1] 2641.432
> print(jumpred/3)
[1] 22.33333

## hybrid GSTAR-RNN
> Compute_Model_Performance_GSTAR_RNN()
      [,1] [,2]
[1,] 0.9996766 0.6084153
      [,1] [,2]
[1,] 0.9993533 0.3701718
      [,1] [,2]
[1,] 9.393136 346.6256
[1] 883.6844

```

BIOGRAFI PENULIS



Nama : Ahmad Saikhu
Tempat / Tanggal Lahir : Gresik / 18 Juli 1971
Pekerjaan : Dosen
Pangkat / Gol : Pembina Tk.I / IV-B
Jabatan Fungsional : Lektor Kepala
Masa Kerja : 14 Tahun
Alamat Kantor : Kampus ITS Sukolilo, 60111
Alamat Rumah : Griya Kartika A-27 Sedati, Sidoarjo
Institusi : Fakultas Teknologi Informasi & Komunikasi
Institut Teknologi Sepuluh Nopember
Email : saikhu@if.its.ac.id

A. Riwayat Pendidikan

- SDN Kawisanyar II, Kebomas Gresik (1977-1983)
- SMP Negeri 1 Gresik (1983-1986)
- SMA Negeri 1 Gresik (1986-1989)
- S1 Statistika ITS (1989-1994)
- S2 Teknik Informatika ITS (1997-2000)

B. Riwayat Pekerjaan

- Staf Production Planning & Control, PT. Asia Tile : 1994
- Dosen STIKOM : 1995-2005
- Dosen ITS : 2005-Sekarang

C. Publikasi Ilmiah selama studi program Doktor

✓ Seminar Internasional

- Saikhu, A., Arifin, A.Z., & Faticah, C. (2017). *Rainfall Forecasting by Using Autoregressive Integrated Moving Average, Single Input and Multi Input Transfer Function*. In 2017 ICTS. Surabaya: IEEE. <https://ieeexplore.ieee.org/document/8265651>.
- Saikhu, A., Arifin, A.Z., & Faticah, C. (2018). *Non-Linear Spatio-Temporal Input Selection for Rainfall Forecasting Using Recurrent Neural Networks*. In 2018 ISITIA. Bali: IEEE. <https://ieeexplore.ieee.org/document/8710864>.

✓ Jurnal Internasional

- Saikhu, A., Arifin, A.Z., & Faticah, C. (2019). *Correlation and Symmetrical Uncertainty-Based Feature Selection for Multivariate Time Series Classification*. International Journal of Intelligent Engineering & Systems, Volume 12, Issue 3, Jun 2019 (Published paper).
<http://www.inass.org/ContentsPapers2019-3.html>.
- Saikhu, A., Arifin, A.Z., & Faticah, C. (2019). *Rainfall Forecasting in the Season Zone based on Spatiotemporal and Simultaneous Predictors Using Generalized Space-Time Autoregressive and Artificial Neural Networks*.
(Draft untuk Egyptian Informatics Journal)