TUGAS AKHIR – KS184822

KOMBINASI *OVERSAMPLING INSIDE* DAN *OUTSIDE FOLD* PADA METODE *RANDOM FOREST* DAN REGRESI LOGISTIK BINER UNTUK ANALISIS KLASIFIKASI *IMBALANCE* LEADS PT "X"

SABILAH MARGIRIZKI
NRP 062116 4000 0113

Dosen Pembimbing
Dr.rer.pol. Heri Kuswanto, S.Si., M.Si.
Santi Puteri Rahayu, M.Si., Ph.D.

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS SAINS DAN ANALITIKA DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2020

TUGAS AKHIR – KS184822

KOMBINASI *OVERSAMPLING INSIDE* DAN *OUTSIDE FOLD* PADA METODE *RANDOM FOREST* DAN REGRESI LOGISTIK BINER UNTUK ANALISIS KLASIFIKASI *IMBALANCE LEADS* PT "X"

SABILAH MARGIRIZKI
NRP 062116 4000 0113

Dosen Pembimbing
Dr.rer.pol. Heri Kuswanto, S.Si., M.Si.
Santi Puteri Rahayu, M.Si., Ph.D.

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS SAINS DAN ANALITIKA DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2020

(Halaman ini sengaja dikosongkan)

FINAL PROJECT – KS184822

COMBINATION OF INSIDE AND OUTSIDE-FOLD OVERSAMPLING IN RANDOM FOREST AND BINARY LOGISTICS REGRESSION TO ANALYSE PT "X"'S IMBALANCE LEADS CLASSIFICATION

SABILAH MARGIRIZKI
SN 062116 4000 0113

Supervisors
Dr.rer.pol. Heri Kuswanto, S.Si., M.Si.
Santi Puteri Rahayu, M.Si., Ph.D.

UNDERGRADUATE PROGRAMME
DEPARTMENT OF STATISTICS
FACULTY OF SCIENCE AND DATA ANALYTICS
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2020

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

KOMBINASI *OVERSAMPLING INSIDE* DAN *OUTSIDE FOLD* PADA METODE *RANDOM FOREST* DAN REGRESI LOGISTIK BINER UNTUK ANALISIS KLASIFIKASI *IMBALANCE LEADS* PT "X"

TUGAS AKHIR

Diajukan untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Statistika
pada
Program Studi Sarjana Departemen Statistika
Fakultas Sains dan Analitika Data
Institut Teknologi Sepuluh Nopember

Oleh:
Sabilah Margirizki
NRP. 062116 4000 0113

Disetujui oleh Pembimbing:
Dr.rer.pol. Heri Kuswanto, S.Si., M.Si. (.....)
NIP. 19820326 200312 1 004

Santi Puteri Rahayu, M.Si., Ph.D. (.....)
NIP. 19750115 199903 2 003

Mengetahui,
Kepala Departemen Statistika

Dr. Dra. Kartika Fithriasari, M.Si.
NIP. 19691212 199303 2 002

Surabaya, Agustus 2020

(Halaman ini sengaja dikosongkan)

KOMBINASI OVERSAMPLING INSIDE DAN OUTSIDE FOLD PADA METODE RANDOM FOREST DAN REGRESI LOGISTIK BINER UNTUK ANALISIS KLASIFIKASI IMBALANCE LEADS PT "X"

Nama Mahasiswa : Sabilah Margirizki
NRP : 06211640000113
Departemen : Statistika-FSAD-ITS
Dosen Pembimbing: Dr.rer.pol. Heri Kuswanto, S.Si., M.Si.
Santi Puteri Rahayu, M.Si., Ph.D.

Abstrak

Software as a Service (SaaS) merupakan sebuah pelayanan software cloud computing. Persaingan yang dinamis di kalangan start-up membuat pengelola harus bisa mengelola customer yang ada dan yang akan datang. Penelitian ini berfokus pada kasus PT "X" yang merupakan perusahaan business-to-business di bidang SaaS dengan melakukan analisis dan prediksi terhadap karakteristik leads PT "X" pada transaksi berhasil dan gagal. Data yang digunakan dalam penelitian kali ini merupakan data klien PT "X" dari 2018 hingga 2019. Penelitian ini menggunakan metode Random Forest (RF) dan Regresi Logistik Biner (RL) yang dikombinasikan dengan oversampling inside (OIF) dan outside fold (OOF). Secara umum, kombinasi metode OOF pada RF dan RL menghasilkan performa klafisikasi lebih tinggi. Nilai rata-rata AUC, g-mean, dan sensitivity dari metode RF-OOF sebesar 90,63%, 75,59%, dan 82,50%. Sementara itu, dengan metode RL-OOF didapatkan hasil yang lebih tinggi, yaitu sebesar 92,32%, 85,56%, dan 88,02% dengan kenaikan rata-rata sebesar 16,44%, 47,99%, dan 65,86% dari data imbalance. Variabel signifikan hasil RL-OOF yaitu SpecialProject, Industry, IntroduceMonth, TeamLeader, dan Source. Odds ratio terbesar terdapat pada kategori Industry10 yang memiliki kecenderungan 11 kali lebih besar untuk melakukan gagal transaksi dibandingkan Industry1.

Kata Kunci: Imbalance, Oversampling, Random Forest, Regresi Logistik Biner, SaaS.

(Halaman ini sengaja dikosongkan)

COMBINATION INSIDE AND OUTSIDE-FOLD OVERSAMPLING IN RANDOM FOREST AND BINARY LOGISTICS REGRESSION TO ANALYSE PT "X"'S IMBALANCE LEADS CLASSIFICATION

Name : Sabilah Margirizki
SN : 06211640000113
Department : Statistics
Supervisors : Dr.rer.pol. Heri Kuswanto, S.Si., M.Si.
Santi Puteri Rahayu, M.Si., Ph.D.

Abstract

Software as a Service (SaaS) is a cloud computing software service. Dynamic competition among start-ups builds the managers to be able to manage existing and future customers. This study focuses on the case of PT "X" which is a business-to-business company in the SaaS field by analyzing and predicting the characteristics of PT "X"'s leads in successful and failed transactions. The data used in this study is PT "X" leads data from 2018 to 2019. This research uses the Random Forest (RF) and Binary Logistic Regression (RL) method combined with oversampling inside (OIF) and outside fold (OOF). In general, the combination of OOF in RF and RL methods results in higher classification performance compared to the RF-OIF and RL-OIF combination methods. The average values of AUC, g-mean, and sensitivity of the RF-OOF method are 90.63%, 75.59%, and 82.50%. Meanwhile, the RL-OOF method gives higher results, which are 92.32%, 85.56%, and 88.02% with an average increase of 16.44%, 47.99% and 65.86% of imbalance data. Significant variables of RL-OOF results are Special Project, Industry, IntroduceMonth, TeamLeader, and Source. The most significant odds ratio is in the Industry10 category, which has an 11 times greater tendency to fail transactions than Industry1.

Keywords: Binary Logistic Regression, Imbalance, Oversampling, Random Forest, SaaS.

(Halaman ini sengaja dikosongkan)

KATA PENGANTAR

Puji syukur penulis panjatkan atas rahmat dan hidayah yang diberikan oleh Allah SWT sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul **"Kombinasi Oversampling Inside dan Outside Fold Pada Metode Random Forest dan Regresi Logistik Biner untuk Analisis Klasifikasi Imbalance Leads PT "X""** dengan baik dan tepat waktu.

Penulis menyadari bahwa dalam menyelesaikan Tugas Akhir ini penulis telah banyak menerima bantuan dan dukungan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan terima kasih kepada :

1. Kedua orang tua dan keluarga, atas setiap do'a, nasihat, dan dukungan yang senantiasa diberikan.
2. Bapak Dr.rer.pol. Heri Kuswanto, S.Si, M.Si dan Ibu Santi Puteri Rahayu, M.Si., Ph.D. selaku pembimbing, yang telah meluangkan waktu dan dengan sangat sabar memberikan bimbingan, saran, serta dukungan kepada penulis selama proses penulisan Tugas Akhir ini.
3. Bapak Dr.rer.pol. Dedy Dwi Prastyo, S.Si., M.Si. dan Bapak Dr. Suhartono, M.Sc selaku dosen penguji, yang telah memberikan arahan dan saran untuk Tugas Akhir ini.
4. Ibu Dr. Dra. Kartika Fithriasari, M.Si. selaku dosen wali dan Kepala Departemen yang telah memberikan saran dan arahan dalam proses belajar di Departemen Statistika ITS.
5. Seluruh dosen Statistika ITS yang telah memberikan ilmu selama berkuliah di ITS, serta segenap karyawan Departemen Statistika ITS, khususnya Bapak Umam, Bapak Fendi, dan Bapak Anton, yang banyak membantu dalam proses administrasi studi *exchange* dan administrasi lainnya.

6. Mbak Ervina, yang telah menerima aplikasi saya untuk menjadi bagian dari Product Marketing Team, sehingga Tugas Akhir ini dapat terwujud.
7. Mas Adhi Duta Baskara, yang sabarnya belum pernah habis sejak pertama bertemu.
8. Seluruh teman-teman ∑27, Statistika ITS angkatan 2016. Serta teman-temanku, Dora, Jessica, Ika, yang sudah membantu penulis dalam menjalani delapan semester di Statistika ITS.
9. Fitria, Rasyid, Niam, Rozie, Kinanthi, dan Abid, yang berbaik hati membantu penulis ketika kebingungan dan *syntax error*.
10. Teman-teman Bogor-ITS, pengisi kebahagiaan tiada tara selama delapan semester di tanah rantau.
11. Teman-teman Tim Barunastra ITS dan tim robotik lainnya yang super keren, atas pengalaman yang luar biasa dan tidak pernah saya sangka-sangka.
12. Semua teman, relasi dan berbagai pihak yang tidak bisa penulis sebutkan namanya satu persatu yang telah membantu dalam penulisan Tugas Akhir ini maupun selama perkuliahan.

Penulis berharap agar Tugas Akhir ini dapat memberikan manfaat kepada pihak yang terkait.

Surabaya, Juni 2020

Penulis

DAFTAR ISI

HALAMAN JUDUL ..... **iii**
COVER PAGE ..... **v**
Abstrak ..... **ix**
KATA PENGANTAR ..... **xiii**
DAFTAR ISI ..... **xv**
DAFTAR GAMBAR ..... **xvii**
DAFTAR TABEL ..... **xix**
DAFTAR LAMPIRAN ..... **xxi**
BAB I PENDAHULUAN **1**
1.1 Latar Belakang 1
1.2 Rumusan Masalah 5
1.3 Tujuan 5
1.4 Manfaat 6
1.5 Batasan Masalah 6
BAB II TINJAUAN PUSTAKA **7**
2.1 Regresi Logistik Biner 7
2.1.1 Estimasi Parameter Model Regresi Logistik Biner 9
2.1.2 Odds Ratio 13
2.2 Classification and Regression Tree dan Random Forest 13
2.3 Boruta 19
2.4 Area Under the Receiver Operating Characteristic Curve (AUC-ROC) 20
2.5 Geometric Mean 22
2.6 Synthetic Minority Oversampling Technique 23
2.7 Stratified K-fold Cross Validation 26
2.8 *Marketing*, CRM, dan SaaS 27
2.8.1 Marketing 27
2.8.2 Customer Relationship Management (CRM) 29
2.8.3 Software as a Service (SaaS) 30

BAB III METODOLOGI PENELITIAN 33
3.1 Sumber Data 33
3.2 Variabel Penelitian dan Struktur Data 33
3.3 Langkah Analisis 35
BAB IV ANALISIS DAN PEMBAHASAN 39
4.1 Analisis Karakteristik *Leads* 39
4.2 Klasifikasi *Leads* PT "X" dengan Metode *Random Forest* 43
4.3 Klasifikasi *Leads* PT "X" dengan Metode Regresi Logistik Biner 56
4.3.1 Estimasi Parameter Model *Leads* PT "X" 63
4.3.2 Pengujian Signifikansi Parameter 65
4.3.3 Interpretasi Model *Leads* PT "X" 67
4.4 Perbandingan Performa Klasifikasi Pada Data Imbalance, Oversampling Inside, dan Outside Fold 69
BAB V KESIMPULAN DAN SARAN 71
5.1 Kesimpulan 71
5.2 Saran 72
DAFTAR PUSTAKA 73
LAMPIRAN 77
BIODATA PENULIS 105

DAFTAR GAMBAR

Halaman

Gambar 2. 1 Kurva Fungsi Logit..... 7
Gambar 2. 2 Ilustrasi Struktur Pohon Klasifikasi..... 14
Gambar 2. 3 Ilustrasi Boruta 20
Gambar 2. 4 Ilustrasi Kurva AUC-ROC 21
Gambar 2. 5 Ilustrasi Oversampling Inside dan Outside Fold..... 24
Gambar 2. 6 Ilustrasi Stratified 3-folds Cross Validation ... 27
Gambar 2. 7 Ilustrasi Model CRM..... 30
Gambar 3. 1 Diagram Alir Penelitian..... 38
Gambar 4. 1 Persentase Leads Deal dan Cancelled/ Lost ... 40
Gambar 4. 2 Chart Jumlah Leads Berdasarkan Source 40
Gambar 4. 3 Scatterplot MRR dan Employee 42
Gambar 4. 4 Line Chart Variabel IntroduceMonth 42
Gambar 4. 5 Seleksi Variabel Metode Random Forest 44
Gambar 4. 6 a) Nilai AUC Berdasarkan Fold; b) Rata-rata Nilai AUC Data Testing Metode Random Forest 54
Gambar 4. 7 a) Nilai G-Mean Berdasarkan Fold; b) Rata-rata Nilai G-Mean Data Testing Metode Random Forest 55
Gambar 4. 8 a) Nilai AUC Berdasarkan Fold; b) Rata-rata Nilai AUC Data Testing Metode Regresi Logistik Biner 61
Gambar 4. 9 a) Nilai G-Mean Berdasarkan Fold; b) Rata-rata Nilai G-Mean Data Testing Metode Regresi Logistik Biner..... 62
Gambar 4. 10 Grafik Nilai Rata-rata AUC, G-Mean, Sensitivitas, dan Spesifisitas..... 69

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Halaman

Tabel 2. 1 Ilustrasi Pengukuran Performansi Model dengan Confusion Matrix 21
Tabel 2. 2 Data Simulasi Oversampling..... 25
Tabel 2. 3 Contoh Oversampling Inside Fold 25
Tabel 2. 4 Contoh Oversampling Outside Fold..... 26
Tabel 3. 1 Variabel Penelitian CRM PT "X" 33
Tabel 3. 2 Definisi Operasional Variabel..... 34
Tabel 3. 3 Struktur Data Variabel Penelitian 35
Tabel 4. 1 Nilai Median dan Rata-rata Pada Variabel Employee 41
Tabel 4. 2 Nilai Median dan Rata-rata Pada Variabel MRR 41
Tabel 4. 3 Hasil Impurity dan Z-Score Boruta..... 43
Tabel 4. 4 Jumlah Kemungkinan Pemilah pada Variabel Input..... 45
Tabel 4. 5 Ilustrasi Pemilihan pada Node Special Project.... 45
Tabel 4. 6 Confusion Matrix Fold ke-10 Data Training Imbalance..... 46
Tabel 4. 7 Ketepatan Klasifikasi RF Data Training Imbalance..... 46
Tabel 4. 8 Ketepatan Klasifikasi RF Data Testing Imbalance..... 47
Tabel 4. 9 Data Ilustrasi SMOTE-NC..... 48
Tabel 4. 10 Cross Tabulation Data Training SMOTE-NC... 49
Tabel 4. 11 Jarak VDM Observasi 2538 dan 471 50
Tabel 4. 12 Ketepatan Klasifikasi RF Data Training Oversampling Inside Fold (%) 50
Tabel 4. 13 Ketepatan Klasifikasi RF Data Testing Oversampling Inside Fold (%) 51
Tabel 4. 14 Ketepatan Klasifikasi RF Data Training Oversampling Outside Fold (%)..... 52

Tabel 4. 15 Ketepatan Klasifikasi RF Data Testing Oversampling Outside Fold (%)..... 52
Tabel 4. 16 Ketepatan Klasifikasi Regresi Logistik Biner Data Training Imbalance (%) 56
Tabel 4. 17 Ketepatan Klasifikasi Regresi Logistik Biner Data Testing Imbalance (%)..... 57
Tabel 4. 18 Confusion Matrix Fold 7 Metode Regresi Logistik Biner Pada Data Testing Imbalance ... 57
Tabel 4. 19 Ketepatan Klasifikasi Regresi Logistik Biner Data Training Oversampling Inside Fold (%) .. 58
Tabel 4. 20 Ketepatan Klasifikasi Regresi Logistik Biner Data Testing Oversampling Inside Fold (%).... 58
Tabel 4. 21 Ketepatan Klasifikasi Regresi Logistik Biner Data Training Oversampling Outside Fold (%) 59
Tabel 4. 22 Ketepatan Klasifikasi Regresi Logistik Biner Data Testing Oversampling Outside Fold (%) . 60
Tabel 4. 23 Nilai Estimasi Parameter Oversampling Inside Fold..... 63
Tabel 4. 24 Pengujian Signifikansi Parameter Secara Serentak..... 65
Tabel 4. 25 Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Inside Fold..... 65
Tabel 4. 26 Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Outside Fold 66
Tabel 4. 27 Odds Ratio Model Oversampling Inside Fold... 67

DAFTAR LAMPIRAN

Halaman

Lampiran 1. Data Leads Deal dan Lost/ Cancelled Deal 77
Lampiran 2. Random Forest Plot Average 18 Nodes..... 78
Lampiran 3. Pengambilan Pohon 3 Nodes 79
Lampiran 4. Syntax Random Forest dan Regresi Logistik Biner [Python]..... 80
Lampiran 5. Syntax Model Regresi Logistik Biner [R] 85
Lampiran 6. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Inside Fold Awal 86
Lampiran 7. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Inside Fold Backward 89
Lampiran 8. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Outside Fold Awal..... 91
Lampiran 9. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Outside Fold Backward 94
Lampiran 10. Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Inside Fold 96
Lampiran 11. Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Outside Fold.... 98
Lampiran 12. Odds Ratio Model Oversampling Inside Fold 100
Lampiran 13. Odds Ratio Model Oversampling Outside Fold 102

(Halaman ini sengaja dikosongkan)

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan zaman dan teknologi saat ini membuat persaingan dalam bisnis semakin meningkat. Berbagai strategi *marketing* terus digunakan untuk membuat suatu produk yang memiliki posisi kuat di pasar. Menurut Owomoyela, Olasunkanmi, & Oyeniyi, (2013) pengembangan strategi *marketing* bertujuan untuk membangun dan mempertahankan keunggulannya dalam berkompetisi menghadapi pesaing. Kemampuan manajerial sangat diperlukan dalam mengatasi ambiguitas lingkungan dan ketidakpastian dalam strategi *marketing*. Saat ini sudah bukan menjadi rahasia bahwa pertumbuhan perusahaan *software* dan *online-service* lebih cepat daripada perusahaan sektor lainnya. Teori dari suatu perusahaan keuangan berpendapat bahwa nilai suatu bisnis berasal dari dua sumber. Pertama adalah pertumbuhan. Kedua adalah pengembalian modal yang diinvestasikan atau *return of investment* (ROI) (Kutcher, Nottebohm, & Sprague, 2014). Bagi perusahaan rintisan (*start-up*), pertumbuhan merupakan suatu hal yang lebih penting dibandingkan ROI. Hal ini sangat mudah dijumpai pada system bakar uang yang seringkali dilakukan oleh pelaku bisnis *start-up* saat ini. Menurut Rhenald Kasali, hal ini muncul karena *venture capitalist* mulai masuk dan berinvestasi pada dunia *start-up*. Berbeda halnya dengan korporat, ROI akan menjadi fokus utama dalam pelaksanaan bisnis. Teori dasar ekonomi mengatakan, dimana terdapat permintaan, maka disitu terbentuk suplai. Riset menemukan bahwa pasar *software Supply Chain Management* (SCM) akan mencapai USD 19 miliar pada 2021, hal ini menjadi salah satu peluang bisnis *software* untuk berkembang lebih besar (Blanchard, 2017).

Software as a Service (seterusnya disebut SaaS) merupakan sebuah perangkat lunak berbasis internet sebagai pengelolaan data dan aplikasi (*cloud computing*) yang dimiliki dan dijalankan secara *online* tanpa terbatas jarak oleh satu maupun beberapa *provider* (Smyrnova, 2019). SaaS berbasis *cloud* sehingga tidak membutuhkan perpindahan secara fisik untuk data apapun yang diperlukan. SaaS adalah sebuah model yang dirancang untuk masa depan yang kian populer di kalangan pebisnis. Pada akhirnya, *cloud computing* menjadi sesuatu yang umum digunakan di segmen bisnis dengan tujuan utama pengembangan bisnis di bidang logistik, transportasi, dan ritel. Perusahaan yang saat ini telah menggunakan teknologi SaaS menyatakan bahwa sudah saatnya *cloud computing* bergerak dengan cepat, telah diasumsikan bahwa 80% dari pekerjaan di suatu perusahaan dapat dialihkan ke *cloud* (Smyrnova, 2019). Salah satu alasan SaaS berkembang dengan sangat pesat adalah karena *hyperspecialization*. Menurut Adam Smith, seorang ekonom asal Scotland, *hyperspecialization* merupakan suatu kondisi dimana pembagian divisi dalam pekerjaan menyebabkan biaya yang lebih rendah, efisiensi yang lebih besar, dan kualitas yang lebih tinggi, karena permasalahan yang berbeda diserahkan kepada ahli di bidang tersebut. Hal ini benar adanya pada kasus pabrik pin tahun 1776, yang menjadi pengetahuan dasar di pabrik hingga hari ini. Dengan begitu, pertumbuhan SaaS yang diiringi oleh analisis lebih dalam dari sisi *marketing* adalah hal yang tepat untuk ini. Pada 2019 terjadi peningkatan sebesar 27% terhadap *tools* yang dapat membantu dunia bisnis sehingga secara kumulatif terdapat sekitar 7000 *tools* yang mana pada 2011 hanya terdapat sekitar 150 (Christopher, 2019).

Saat ini SaaS mencakup beberapa kategori fungsional yang diantaranya adalah *Account-Based Marketing* (ABM), *Customer Relationship Management* (CRM), *Enterprise*

Resource Planning (ERP), surat elektronik, manajemen proyek, *e-commerce*, manajemen data, dan sebagainya. Berbagai cara dilakukan untuk mendapatkan perhatian pasar, perseorangan atau bisnis, yang membutuhkan SaaS sesuai dengan bidangnya. Perseorangan maupun suatu bisnis yang pada akhirnya berpotensi menjadi klien disebut *lead(s).* Dengan melakukan *leads analysis,* suatu perusahaan dapat melacak informasi tentang status dan efektivitas iklan dan promosi dalam menghasilkan penjualan. Pada akhirnya, hal ini akan memungkinkan perusahaan untuk memprioritaskan karakteristik *lead* tertentu yang mengarah pada penggunaan produk.

Penelitian mengenai CRM menggunakan metode *machine learning* sudah dilakukan sebelumnya oleh (Sabbeh, 2018). Penelitian tersebut menganalisis karakteristik *customer* untuk memberikan keunggulan yang harus dipertahankan oleh perusahaan tersebut guna meningkatkan *customer retention rate,* penelitian ini dilakukan dengan 10 metode *machine learning*. *Random forest* merupakan metode terbaik dari semua yang digunakan yaitu dengan nilai akurasi sebesar 96,39%. Selain itu, penelitian oleh Nabavi & Jafari (2013) mengenai prediksi *customer churn* pada Solico Food Industries Group untuk memprediksi pelanggan berhenti menggunakan produknya (*customer churn*) dengan standar model CRISP-DM dan diperoleh hasil bahwa metode *Random Forest* memiliki tingkat akurasi terbaik yaitu sebesar 76,64% dibanding metode lainnya. Penelitian oleh Starzyczná, Pellešová & Stoklasa (2017) menggunakan regresi logistik biner untuk mengetahui hubungan antara CRM dan *Small and Medium Enterprises* (seterusnya disebut SME) yang hasilnya adalah meningkatkan jumlah *loyal customer* yang berarti meningkatkan kepuasan pelanggan dan kepercayaan terhadap perusahaan lebih berdampak pada manfaat ekonomi sistem.

Penelitian kali ini akan berfokus pada kasus PT "X", yang merupakan perusahaan perangkat lunak yang menjual produknya ke perusahaan lain atau biasa disebut *business-to-business* (seterusnya disebut B2B) di bidang SaaS. Saat ini, PT "X" sedang berkembang pesat dalam menyediakan perangkat lunak untuk SME. Pada 2020 ini terdapat ribuan perusahaan yang telah mempercayai PT "X" dan menggunakan produknya serta terdapat setidaknya 10 perusahaan sejenis di Indonesia yang menjadi pesaing PT "X" dalam menjalankan bisnisnya. Maka dari itu untuk tetap dapat menjadi yang terdepan dalam mendapatkan klien yang berpotensi diperlukan strategi yang tepat dalam memasarkan produk yang dimiliki. PT "X" memiliki produk berbasis berlangganan dalam periode waktu tertentu. Pada penelitian ini akan dilakukan analisis dan prediksi terhadap karakteristik klien pada PT "X" untuk memprediksi apakah *lead*s termasuk dalam kategori *deal* atau *lost*. Data yang digunakan dalam penelitian kali ini merupakan data *leads* PT "X" dari 2018 hingga 2019 dengan perbandingan kedua kategori variabel respon yaitu 1:6 (*imbalance*). Penelitian ini akan menggunakan metode *Random Forest* dan regresi logistik biner dengan *oversampling* menggunakan SMOTE untuk menangani *imbalance* pada kategori variabel respon. Hasil penelitian ini diharapkan dapat memberikan informasi kepada PT "X" dalam mendapatkan karakteristik *leads* tertentu yang berpotensi untuk *deal*, sehingga dapat menjadi dasar dalam berbagai kebijakan agar mendapatkan metode yang paling efisien dalam memperluas periklanan produk sehingga lebih tepat sasaran, dan memberikan *treatment* tertentu dalam bentuk promosi kepada klien yang tepat.

1.2 Rumusan Masalah

Hilangnya *leads* pada proses transaksi tentunya merugikan penggerak bisnis dari segi waktu dan menjadikan tidak tercapainya target. Selain itu, prediksi dengan klasifikasi di saat kelompok data tidak seimbang akan menghasilkan prediksi yang tidak sesuai harapan. Oleh karena itu, peneliti menarik rumusan masalah mengenai bagaimana karakteristik *leads* pada PT "X" berdasarkan analisis statistika deskriptif, faktor-faktor yang berpengaruh terhadap klasifikasi *leads* dan prediksinya menggunakan *Random Forest* dengan *oversampling inside* dan *outside fold*, variabel yang berpengaruh signifikan terkait prediksi *leads* menggunakan regresi logistik biner dengan *oversampling inside* dan *outside fold*, dan perbandingan *oversampling inside fold* dan *oversampling outside fold.*

1.3 Tujuan

Berdasarkan rumusan masalah yang telah disusun, tujuan penelitian ini secara umum adalah untuk menganalisis perilaku konsumen pada PT "X" terkait status penggunaan produk *deal* atau *lost*. Secara khusus, tujuan yang ingin dicapai adalah sebagai berikut.

1. Memperoleh hasil karakteristik *leads* PT "X" dengan statistika deskriptif.
2. Mendapatkan *important variable* terkait prediksi klasifikasi *leads* PT "X" dan ketepatan prediksinya menggunakan metode *Random Forest* yang dikombinasikan dengan metode *oversampling inside fold* dan *oversampling outside fold*.
3. Mendapatkan variabel yang berpengaruh signifikan terkait prediksi klasifikasi *leads* PT "X" dan ketepatan prediksinya dengan metode regresi logistik biner yang

dikombinasikan dengan metode *oversampling inside fold* dan *oversampling outside fold*.

4. Mendapatkan hasil perbandingan *oversampling inside fold* dan *oversampling outside fold*.

1.4 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat bagi seluruh pihak yang terlibat secara langsung maupun tidak langsung, diantaranya sebagai berikut.

1. Bagi Perusahaan
   Hasil riset dapat digunakan untuk meningkatkan efisiensi operasional, serta pendapatan perusahaan dengan mengetahui target klien yang lebih potensial.
2. Bagi Peneliti
   Memberikan ilmu pengetahuan dan wawasan mengenai *Customer Relationship Management* (CRM), mengetahui pengaruh studi karakteristik klien terhadap keputusan berlangganan suatu produk, serta penerapan metode klasifikasi pada kasus nyata.

1.5 Batasan Masalah

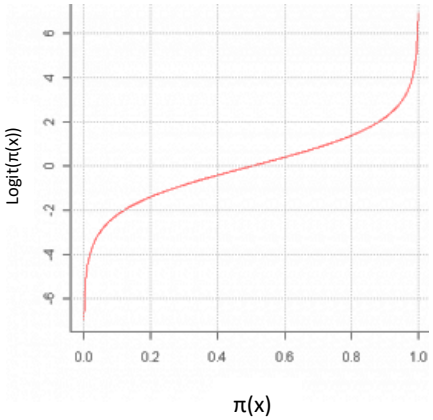
Batasan masalah dalam penelitian ini adalah data leads Customer Relationship Management (CRM) yang digunakan dari PT “X” tercatat sejak 2018 hingga Desember 2019. Variabel respon merupakan *leads* yang melakukan transaksi sebagai bentuk persetujuan berlangganan produk PT “X” dan *leads* yang melakukan pembatalan di tengah proses transaksi. Variabel input yang digunakan disesuaikan dengan variabel yang ada pada platform CRM yang digunakan PT “X”.

BAB II
TINJAUAN PUSTAKA

Bab ini membahas mengenai regresi logistik biner, *classification and regression tree, Random Forest, Area Under the Receiver Operating Curve* (AUC-ROC), *k-fold cross validation*, SMOTE *marketing, Customer Relationship Management* (CRM), dan *Software as a Service* (SaaS).

2.1 Regresi Logistik Biner

Regresi logistik merupakan salah satu bagian dari analisis regresi yang digunakan untuk memprediksi probabilitas kejadian suatu peristiwa dengan mencocokkan data pada fungsi logit kurva logistik.



Gambar 2. 1 Kurva Fungsi Logit

Adapun fungsi logit dari Gambar 2.1 adalah sebagai berikut.

$$logit(\pi(x)) = \ln\left(\frac{\pi(x)}{1-\pi(x)}\right) \quad (2.1)$$

dengan $\pi(x)$ merupakan nilai probabilitas suatu kejadian.

Seperti analisis regresi pada umumnya, metode ini menggunakan satu atau beberapa variabel bebas dengan satu variabel kategorikal tak bebas dan pada penelitian ini bersifat

biner atau dikotomi. Regresi logistik juga digunakan secara luas pada bidang kedokteran, ilmu sosial dan bahkan pada bidang pemasaran, seperti prediksi kecenderungan pelanggan untuk membeli suatu produk atau berhenti berlangganan. Regresi logistik biner tidak memerlukan asumsi normalitas, heteroskedastisitas dan autokorelasi, dikarenakan variabel terikat yang terdapat pada regresi logistik merupakan variabel *dummy* (0 dan 1), sehingga residualnya tidak memerlukan ketiga pengujian tersebut. Asumsi multikolinearitas hanya melibatkan variabel-variabel bebas, maka masih perlu untuk dilakukan pemeriksaan multikolinearitas tahap awal dengan uji kesesuaian (*goodness of fit test*) yang kemudian dilanjutkan dengan pengujian hipotesis guna melihat variabel bebas mana saja yang signifikan dan dapat tetap digunakan dalam penelitian. Selanjutnya di antara variabel bebas yang signifikan, dapat dibentuk suatu matriks korelasi, dan apabila tidak terdapat variabel bebas yang saling memiliki korelasi yang tinggi (>0,7 (Dormann, et al., 2012)), maka dapat disimpulkan bahwa tidak terdapat gangguan multikolinearitas pada model penelitian (Hosmer & Lemeshow, 2000).

Menurut Hosmer dan Lemeshow (2000) tujuan melakukan analisis data kategori menggunakan regresi logistik adalah mendapatkan model untuk menjelaskan hubungan antara keluaran dari variabel respons (Y) dengan variabel-variabel prediktornya (X). Variabel respons dalam regresi logistik berupa kategori atau kualitatif, sedangkan variabel prediktornya dapat berupa kualitatif dan kuantitatif. Jika variabel Y merupakan variabel biner atau dikotomi dalam artian variabel respons terdiri dari dua kategori yaitu "sukses" (Y = 1) atau "gagal" (Y = 0), maka variabel Y mengikuti sebaran Bernoulli yang memiliki fungsi densitas peluang:

$$f(y_i) = \pi(x_i)^{y_i}(1-\pi(x_i))^{1-y_i} \qquad (2.2)$$

dengan $y_i = 0, 1$ dan $i = 1, 2 \ldots n$

Sehingga diperoleh:

Untuk $y_i = 0$, maka

$$f(0) = \pi(x_i)^0(1 - \pi(x_i))^{1-0} = 1 - \pi(x_i) \qquad (2.3)$$

untuk $y_i = 1$, maka

$$f(1) = \pi(x_i)^1(1 - \pi(x_i))^{1-1} = \pi(x_i) \qquad (2.4)$$

Misalkan probabilitas dari variabel respons Y untuk nilai x yang diberikan, dinotasikan sebagai (x). Model umum (x) dinotasikan sebagai berikut:

$$\pi(x) = \frac{\exp(\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p)}{1+\exp(\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots+\beta_p x_p)} \qquad (2.5)$$

Persamaan (2.5) disebut fungsi regresi logistik yang menunjukkan hubungan antara variabel prediktor dan probabilitas yang tidak linear, sehingga untuk mendapatkan hubungan yang linear dilakukan transformasi yang sering disebut dengan transformasi logit. Bentuk logit dari (x) dinyatakan sebagai $g(x)$, yaitu:

$$logit[\pi(x)] = g(x) = \ln\left(\frac{\pi(x)}{1-\pi(x)}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p. \qquad (2.6)$$

Persamaan (2.6) merupakan bentuk fungsi hubungan model regresi logistik yang disebut model regresi logistik berganda (Hosmer & Lemeshow, 2000).

2.1.1 Estimasi Parameter Model Regresi Logistik Biner

Estimasi parameter dalam regresi logistik dilakukan dengan metode *Maximum Likelihood Estimation* (MLE).

Metode *maximum likelihood* mengestimasi parameter $\boldsymbol{\beta}$ dengan cara memaksimalkan fungsi *likelihood* (Agresti, 2012). Apabila diambil sampel *random* sebanyak n dengan $i = 1, 2, ..., n$, fungsi likelihood yang dimaksimumkan dapat dinyatakan dalam persamaan (2.6)

$$l(\boldsymbol{\beta}) = \prod_{i=1}^{n} \pi(x_i)^{y_i} [1 - \pi(x_i)]^{1-y_i} \tag{2.7}$$

Kemudian persamaan (2.7) digunakan untuk menyusun *ln* fungsi *likelihood* seperti pada persamaan (2.8)

$$L(\boldsymbol{\beta}) = \ln\left[l(\boldsymbol{\beta})\right] = \sum_{i=1}^{n} \left\{ y_i \ln\left[\pi(x_i)\right] + (1 - y_i) \ln\left[1 - \pi(x_i)\right] \right\} \tag{2.8}$$

dengan $\boldsymbol{\beta} = [\beta_0 \;\; \beta_1 \;\; \ldots \;\; \beta_p]^T$ dan

$\mathbf{X} = [\mathbf{1} \;\; \mathbf{X}_{1i} \;\; \mathbf{X}_{2i} \;\; \ldots \;\; \mathbf{X}_{pi}]^T$

Kemudian dengan melakukan differensial terhadap persamaan (2.8), maka dapat dihitung parameter-parameter regresi logistik biner dengan persamaan (2.9).

$$\begin{aligned} \frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}'} &= \frac{\partial \sum_{i=1}^{n} [y_i \mathbf{X'}_i \boldsymbol{\beta} - \ln(1 - \exp(\mathbf{X'}_i \boldsymbol{\beta}))]}{\partial \boldsymbol{\beta}'} \\ &= \sum_{i=1}^{n} \left[y_i \mathbf{X'}_i - \frac{\mathbf{X}_i \exp(\mathbf{X'}_i \boldsymbol{\beta})}{1 + \exp(\mathbf{X'}_i \boldsymbol{\beta})} \right] \\ &= \sum_{i=1}^{n} [y_i - \pi(\boldsymbol{X}_i)] \boldsymbol{X}_i = 0 \end{aligned} \tag{2.9}$$

Persamaan (2.9) bukan merupakan persamaan *closed form* sehingga diperlukan metode iterasi numerik untuk memperoleh

$\boldsymbol{\beta}$ yang konvergen. Metode iterasi Newton Raphson digunakan untuk menyelesaikan persamaan non-linear, dengan rumus iterasi seperti pada persamaan (2.10).

$$\boldsymbol{\beta}^{(\mathbf{b+1})} = \boldsymbol{\beta}^{(\mathbf{b})} - \mathbf{H}^{-1}(\boldsymbol{\beta}^{(\mathbf{b})})\mathbf{g}(\boldsymbol{\beta}^{(\mathbf{b})}) \qquad (2.10)$$

Langkah-langkah iterasi Newton Raphson adalah sebagai berikut.

1. Menentukan nilai awal estimasi parameter $\widehat{\boldsymbol{\beta}}$(b),
2. Membentuk vektor gradien g dan matriks Hessian **H**,
3. Memasukkan nilai $\widehat{\boldsymbol{\beta}}$(b) pada elemen g dan H sehingga diperoleh $\mathbf{g}(\widehat{\boldsymbol{\beta}}(b))$ dan $\mathbf{H}(\widehat{\boldsymbol{\beta}}(b))$,
4. Iterasi dimulai b = 0 menggunakan persamaan (2.10). Nilai $\widehat{\boldsymbol{\beta}}$(b) adalah sekumpulan penaksir parameter yang konvergen pada iterasi ke- b,
5. Apabila belum diperoleh estimasi parameter yang konvergen, maka langkah (3) diulang kembali hingga nilai $\|\widehat{\boldsymbol{\beta}}\mathbf{(b+1)} - \widehat{\boldsymbol{\beta}}\mathbf{(b)}\| \leq \varepsilon$, dengan ε merupakan bilangan yang sangat kecil. Hasil estimasi yang diperoleh adalah $\widehat{\boldsymbol{\beta}}$(b+1) pada iterasi terakhir.

Menurut Hosmer dan Lameshow (2000), model yang telah diperoleh perlu diuji signifikansinya dengan melakukan pengujian statistik. Pengujian dilakukan melalui uji serentak dan uji parsial yang akan dijelaskan sebagai berikut.

1. Uji Serentak
   Uji serentak dilakukan untuk memeriksa makna koefisien β secara keseluruhan.
   Hipotesis:
   H_0: $\beta_1 = \beta_2 = \ldots = \beta_p = 0$
   H_1: Minimal ada satu $\beta_j \neq 0$; j = 1, 2... p

Statistik uji:

$$G^2 = -2\,ln\frac{\left(\frac{n_0}{n}\right)^{n_0}\left(\frac{n_1}{n}\right)^{n_1}}{\sum_{i=1}^{n}\hat{\pi}_i^{\,y_i}(1-\hat{\pi}_i)^{(1-y_i)}} \tag{2.11}$$

Keterangan
n_1: Banyaknya observasi yang berkategori 1
n_0: Banyaknya observasi yang berkategori 0
n : Banyaknya observasi ($n_1 + n_0$)
Daerah penolakan:
Tolak H_0 jika $G^2 > \chi^2_{(p-1,\alpha)}$.

2. Uji Parsial
Uji parsial digunakan untuk memeriksa kemaknaan β secara individu.
H_0: $\beta_j = 0$; $j = 1, 2, \ldots p$
H_1: $\beta_j \neq 0$
Statistik uji:

$$W = \frac{\widehat{\beta_j}}{SE(\widehat{\beta_j})} \tag{2.12}$$

Rasio yang dihasilkan dari statistik uji dibawah hipotesis H_0 akan mengikuti sebaran normal standar. Sehingga, untuk memperoleh keputusan dilakukan perbandingan dengan distribusi normal standar (*Z*). Kriteria penolakan (tolak H_0) jika nilai $|W| > Z_{\frac{\alpha}{2}}$ dan dapat diperoleh melalui persamaan berikut.

$$W^2 = \frac{\widehat{\beta}_j^{\,2}}{SE(\widehat{\beta}_j)^2} \tag{2.13}$$

Statistik uji tersebut mengikuti distribusi Chi-Squared, sehingga tolak H_0 jika $W^2 > \chi^2_{(p-1,\alpha)}$.

2.1.2 Odds Ratio

Odds Ratio menunjukkan perbandingan peluang munculnya suatu kejadian dengan peluang tidak munculnya kejadian tersebut (Wulandari, Salamah, & Susilaningrum, 2009). Rasio peluang bagi prediktor diartikan sebagai jumlah relatif dimana peluang hasil meningkat (rasio peluang > 1) atau turun (rasio peluang < 1) ketika nilai variabel prediktor meningkat sebesar 1 unit. Menurut Hosmer dan Lemeshow perhitungan Odds Ratio adalah sebagai berikut.

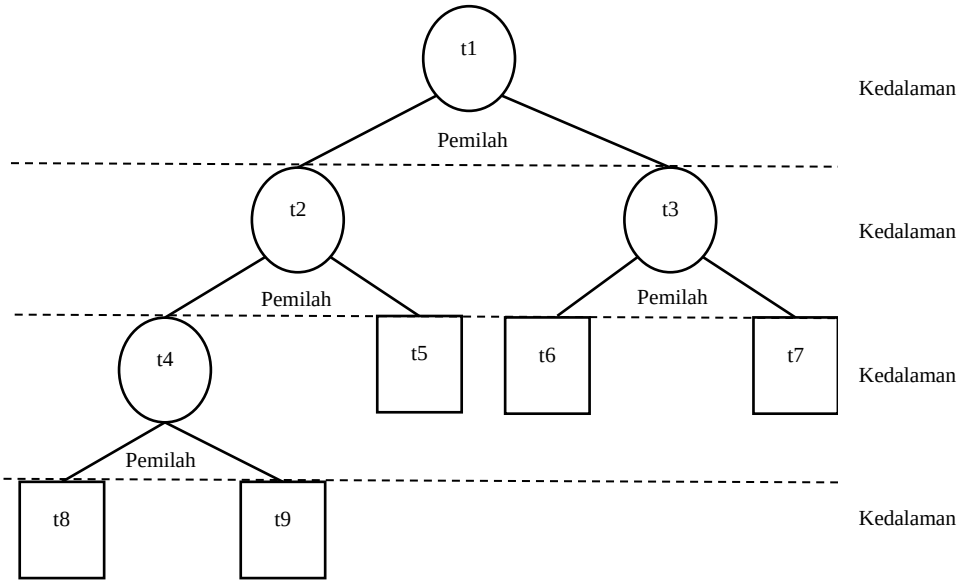
$$OR = \frac{\frac{\pi(1)}{[1-\pi(1)]}}{\frac{\pi(0)}{[1-\pi(0)]}} = \frac{\pi(1)[1-\pi(0)]}{\pi(0)[1-\pi(1)]} = \frac{e^{\beta_0+\beta_1}}{e^{\beta_0}} = e^{\beta_1} \qquad (2.14)$$

Odds ratio memperkirakan seberapa besar kemungkinan (atau ketidakmungkinan) hasilnya merupakan variabel respon dengan kode 0 atau 1. Sebagai contoh, ketika variabel respon *y* merupakan keputusan klien untuk menggunakan produk (0) atau tidak (1) dan variabel input *x* merupakan harga produk, maka apabila nilai odds ratio = 2 memiliki arti kemungkinan klien tidak menggunakan produk dua kali lebih besar daripada menggunakan produk dengan peningkatan setiap satu satuan harga.

2.2 Classification and Regression Tree dan Random Forest

Classification and Regression Tree (CART) merupakan suatu model pohon yang digunakan dalam mengklasifikasi ketika variabel target kategorikal. CART membangun model dengan pertanyaan kategoris seperti “ya” atau “tidak” (nominal), namun bisa juga untuk ordinal yang bersifat

klasifikasi (Williams, 2011). CART akan menghasilkan pohon klasifikasi dengan variabel respon kategorik dan menghasilkan pohon regresi jika variabel respon kontinu (Breiman L. , 2001). Model pohon regresi menghasilkan *m* himpunan hasil numerik yang dihitung secara matematis dengan menguji hubungan antar variabel target dan prediktor. Formula ini kemudian diterapkan pada pengamatan baru untuk memprediksi kemungkinan hasil. CART tidak hanya menghasilkan klasifikasi namun juga memberikan estimasi probabilitas kesalahan dalam estimasi.



Gambar 2. 2 Ilustrasi Struktur Pohon Klasifikasi

Sumber: (Breiman, Friedman, Olshen, & Stone, 1993).

Ilustrasi struktur pohon klasifikasi yang ditunjukkan pada gambar 2.2 merupakan awalan yang mengandung seluruh

data dengan notasi t1. Pada lingkaran berikutnya, atau disebut *internal node*, dinotasikan dengan t2, t3, dan t4. *Terminal node* atau simpul akhir dinotasikan dengan t5, t6, t7, t8, dan t9 yang mana setelahnya tidak ada lagi pemilahan. Setiap *node* memiliki kedalaman tertentu, dimulai dari t1 pada *node* 1, t2 dan t3 pada *node 2*, dan seterusnya.

Pembentukan pohon klasifikaasi diawali dengan menentukan variable dan nilainya untuk dijadikan pemilah utama. Proses pembentukan pohon klasifikasi terdiri dari tiga tahap yaitu pemilihan pemilah, penentuan *terminal node*, dan penandaan label kelas (Breiman, Friedman, Olshen, & Stone, 1993).

1. Pemilihan pemilah (*Classifier*)

   Pemilahan dilakukan pada sampel data *training* dengan aturan *goodness of split*, dimana sampel pada data training yang digunakan masih bersifat heterogen. Pemilihan pemilah tergantung pada jenis variabel respon. Hasil proses pemilahan harus lebih homogen dibandingkan simpul induknya. Tingkat keheterogenan simpul tersebut dapat diukur menggunakan nilai *impurity* atau $imp(t)$. Fungsi heterogenitas yang umum digunakan adalah Indeks Gini yang dituliskan dalam persamaan berikut.

   $$imp(t) = \sum_{j=1}^{i} p(j|t)p(i|t)\,, i \neq j \qquad (2.15)$$

   Keterangan:

   $p(j|t)$ = proporsi kelas *j* pada simpul *t*

   $p(i|t)$ = proporsi kelas *i* pada simpul *t*

   Kriteria *goodness of split* $(\phi(s,t))$ untuk mengevaluasi pemilah *s* pada simpul *t* didefinisikan sebagai berikut.

   $$\big(\phi(s,t)\big) = imp(st) = imp(t) - p_L imp(t_L) - p_R imp(t_R) \qquad (2.16)$$

Keterangan:

$imp(t)$ = fungsi heterogenitas pada simpul *t*

p_L = proposi pengamatan simpul kiri

p_R = proposi pengamatan simpul kanan

$imp(t_L)$ = fungsi heterogenitas pada simpul anak kiri

$imp(t_R)$ = fungsi heterogenitas pada simpul anak kanan

$$\Delta_i(s^*, t_1) = max\Delta_i(s, t) \qquad (2.17)$$

Pemilah yang menghasilkan nilai *goodness of split* tertinggi merupakan pemilah terbaik karena dapat mengurangi heterogenitas lebih tinggi. Langkah pemilihan pemilah yang telah dibahas sebelumnya dilakukan secara berulang untuk menentukan variabel yang digunakan sebagai pemilah pada setiap *node*, mulai dari *root node*, hingga *internal node*. Pengembangan pohon dilakukan dengan pencarian pemilah yang mungkin pada simpul t_j yang kemudian akan dipilah menjadi t_2 dan t_3 oleh pemilah s^* dan begitu seterusnya.

2. Penentuan *terminal node*

   Suatu simpul *t* dikatakan sebagai *terminal node* ketika tidak terdapat penurunan heterogenitas yang signifikan, atau hanya terdapat satu pengamatan di setiap simpul anak, atau terdapat batasan minimum *n* pengamatan di setiap simpul anak yang dihasilkan (Breiman, Friedman, Olshen, & Stone, 1993).

3. Penandaan label kelas

   Penandaan label kelas pada simpul terminal berdasarkan aturan jumlah terbanyak seperti pada persamaan (2.17).

$$p(j_0|t) = \max\ p(j|t) = \max\frac{N_j(t)}{N(t)} \qquad (2.18)$$

   dengan $N_j(t)$ merupakan banyaknya amatan kelas *j* pada *terminal nodes t* dan *N*(t) merupakan jumlah total pengamatan dalam *terminal node t.*

Setelah pohon klasifikasi terbentuk, perlu juga dilakukan *pruning* atau pemangkasan pohon karena semakin banyak

pemilahan yang dilakukan mengakibatkan makin kecilnya tingkat kesalahan prediksi atau dengan kata lain nilai prediksi melebihi nilai yang sebenarnya (*overfitting*). Pemangkasan pohon dilakukan dengan menentukan *cost complexity minimum* yang dihitung dengan formula sebagai berikut (Breiman, Friedman, Olshen, & Stone, 1993).

$$R_a(T) = R(T) + a|T| \qquad (2.19)$$

Keterangan:

$R(T)$ = ukuran kompleksitas suatu pohon T pada kompleksitas a

(T) = penduga pengganti (*resubtitution estimate*) pohon atau ukuran kesalahan klasifikasi pohon T

a = parameter *cost complexity* bagi penambah satu simpul terminal pada pohon T

$|T|$ = banyaknya simpul terminal pada pohon

Setelah dilakukan pemangkasan, agar dicapai ukuran pohon yang optimum maka perlu dilakukan penentuan pohon klasifikasi optimal dengan ukuran yang lebih sederhana dan nilai penduga pengganti yang kecil. Ukuran pohon yang terlalu besar akan menyebabkan nilai kompleksitas yang tinggi. *Test sample estimate* merupakan penduga pengganti yang sering digunakan jika ukuran pengamatan besar. Prosedur ini diterapkan dengan membagi sampel L menjadi dua bagian, yaitu L_1 (training) dan L_2 (testing). Pengamatan L_1 digunakan untuk membentuk pohon T, sedangkan pengamatan L_2 digunakan untuk menduga (T). N_1 merupakan jumlah pengamatan L_1 dan N_2 jumlah pengamatan L_2. $X(.)$ bernilai 0 jika pernyataan dalam tanda kurung salah dan bernilai 1 jika pernyataan dalam tanda kurung benar. Penduga sampel uji dihitung menggunakan persamaan 2.18.

$$R^{ts}(T_t) = \frac{1}{N_2}\sum_{(x_n,j_n)\in L_2}^{N} X(d(x_n) \neq j_n) \qquad (2.20)$$

Keterangan:
$R^{ts}(T_t)$ = total proporsi kesalahan test sample estimate
N_2 = jumlah pengamatan dari data training L_2
Persamaan ini digunakan ketika ingin menduga proporsi kesalahan yang dihasilkan dari proses pembentukan pohon klasifikasi, sehingga pohon klasifikasi optimal yang dipilih adalah pohon T_t yang memiliki nilai penduga sampel uji minimum atau $R^{ts}(T_t) = min_t R^{ts}(T_t)$.

Metode *Random Forest* merupakan perluasan dari CART. Pembuatan kelompok (pohon) baru dari beberapa model *Decision Tree* yang dirancang untuk menghasilkan perhitungan yang lebih baik dalam akurasi klasifikasi. Pembuatan kelompok-kelompok baru tersebut seringkali dihasilkan secara acak yang kemudian memengaruhi pembuatan kelompok baru berikutnya (Breiman L. , 2001). Pada kelompok data yang terdiri atas *n* observasi dan *p* variable prediktor, prosedur untuk melakukan *Random Forest* adalah sebagai berikut (Breiman L. , 2001).

1. Tahap *bootstrap*
   Lakukan penarikan sampel acak berukuran *n* dengan pengembalian pada kelompok data.
2. Dengan sampel *bootstrap*, kelompok baru dibentuk hingga mencapai jumlah maksimum. Hal ini dilakukan dengan menerapkan *random feature selection* pada setiap proses pemilihan pemilah, yaitu *m* variable prediktor dipilih secara acak dimana $m < p$, lalu pemilah terbaik dipilih berdasarkan banyaknya *m* variable predictor tersebut.
3. Ulangi langkah 1 dan 2 sebanyak *k* kali hingga terbentuk sebuah hutan yang terdiri atas *k* pohon.

Agar dapat mengklasifikasikan kelompok yang baru, setiap *Decision Tree* menyediakan klasifikasi untuk input data berupa data sampel dari dataset asli yang kemudian

diklasifikasi kembali oleh kumpulan kelompok tersebut (*Random Forest*) dan memilih prediksi yang paling banyak dipilih sebagai hasil akhir (Mao & Wang, 2013).

2.3 Boruta

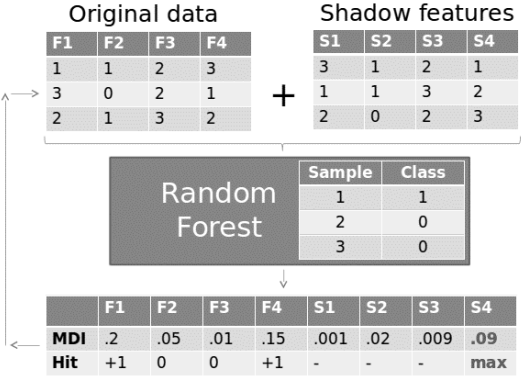
Boruta merupakan algoritma untuk melakukan seleksi variabel. Boruta menjalankan beberapa model *Random Forest* untuk memperoleh variabel yang signifikan secara statistik antara variabel yang relevan dan variabel yang tidak berpengaruh (Kursa, Jankowski, & Rudnicki, Boruta – A System for Feature Selection, 2010). Boruta menggunakan *shadow*, yang merupakan salinan variabel asli dengan nilai acak, dalam melakukan iterasi. Variabel yang secara signifikan memiliki nilai di atas *shadow* akan dikategorikan ke dalam variabel yang penting dan tetap masuk pada iterasi berikutnya. Sebaliknya, variabel dengan nilai di bawah *shadow* akan dihapus pada iterasi berikutnya. Iterasi berhenti ketika seluruh variabel sudah mendapatkan nilainya atau jika sudah sampai jumlah iterasi maksimum yang ditentukan sebelumnya (Kursa, Boruta for Those In A Hurry, 2020). Adapun tahapan dan ilustrasi pada Boruta sebagai berikut.

1. Lakukan penggandaan pada data dan mengacak nilainya pada kolom yang berbeda. Dataset hasil penggandaan disebut *shadow*. Pada *shadow dataset* akan dilakukan pemilahan seperti pada algoritma *Random Forest* untuk mendapatkan gambaran *feature* yang dianggap penting dari nilai *Mean Decrease Accuracy* (MDA) atau *Mean Decrease Impurity* (MDI), dengan menghitung nilai rata-rata dari persamaan (2.15). Nilai yang tinggi mengindikasikan variabel tersebut baik atau penting.
2. Algoritma akan membandingkan tingkat kepentingan variabel tersebut dengan menghitung nilai Z. Variabel

asli yang memiliki nilai Z lebih besar dari nilai Z maksimum pada *shadow features* akan diberikan nilai *Hit* +1. Artinya, variabel tersebut dapat dikatakan penting atau pun *tentative*. Tergantung dari besarnya nilai.

3. Iterasi terus dilakukan hingga seluruh variabel sudah mendapatkan nilainya masing-masing dan diketahui tingkat kepentingannya.

Adapun ilustrasi Boruta sebagai berikut.

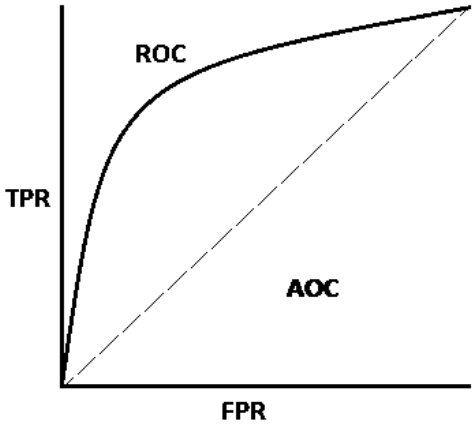


Gambar 2. 3 Ilustrasi Boruta

2.4 Area Under the Receiver Operating Characteristic Curve (AUC-ROC)

Kurva AUC-ROC adalah suatu pengukur kinerja untuk klasifikasi pada berbagai batas (*threshold*). ROC atau *receiver operating curve* adalah kurva probabilitas dan AUC (*area under curve*) merupakan luas bagian bawah dari kurva ROC. Hal ini memberikan informasi seberapa kapabel suatu model tersebut dalam mengklasifikasikan kelas.

Pada plot kurva ROC terdapat sumbu *x* dan *y* dimana *False Positive Rate* (FPR) berada di sumbu *x* dan *True Positive Rate* (TPR) di sumbu *y*. Hal ini diilustrasikan sebagai berikut.



Gambar 2. 4 Ilustrasi Kurva AUC-ROC
Sumber: (Narkhede, 2018)

Metode pengukuran kinerja untuk model klasifikasi dapat disimpulkan menggunakan *confusion matrix*. Jumlah prediksi yang benar dan salah dihitung dan dirangkum pada tabel berdasarkan masing-masing kelas. *Confusion matrix* memberikan informasi pada kesalahan model klasifikasi.

Tabel 2. 1 Ilustrasi Pengukuran Performansi Model dengan *Confusion Matrix*

Actual Values	*Predicted Values*	
	Negative (0)	*Positive (1)*
Negative (0)	*TN*	*FP*
Positive (1)	*FN*	*TP*

TPR (*sensitivity)*, *specificity*, FPR, dan akurasi total didefinisikan sebagai berikut (Narkhede, 2018).

$$TPR = \frac{TP}{TP+FN} \tag{2.21}$$

$$Specificity = \frac{TN}{TN+FP} \tag{2.22}$$

$$FPR = 1 - Specificity \quad (2.23)$$

$$Akurasi\ Total = \frac{TP+TN}{TP+FP+TN+FN} \quad (2.24)$$

Keterangan:
TP = *true positive*, observasi positif dan hasil prediksi positif.
TN = *true negative*, observasi negatif dan hasil prediksi negatif.
FP = *false positive*, observasi negatif, hasil prediksi positif.
FN = *false negative*, observasi positif, hasil prediksi negatif.

Semakin tinggi nilai AUC dan akurasi menandakan semakin baik model tersebut dalam memprediksi 0 sebagai 0 dan 1 sebagai 1 (Narkhede, 2018). AUC dapat diperoleh dengan menghitung rata-rata pada seluruh kombinasi AUC *one-vs-one* dan memiliki fungsi seperti AUC pada umumnya (Fawcett, 2006). Model AUC yang baik didefinisikan dengan nilai mendekati 1, sementara model yang buruk mendekati 0. Ketika nilai AUC sama dengan 0,5 hal ini menandakan model tersebut tidak memiliki kapasitas untuk mengklasifikasi (Narkhede, 2018). AUC pada klasifikasi biner dihiting dengan rumus sebagai berikut (Wray, Yang, Goddard, & Visscher, 2010).

$$AUC = \frac{1+TPR-FPR}{2} \quad (2.25)$$

2.5 Geometric Mean

Geometric Mean (G-Mean) adalah suatu metrik untuk mengukur keseimbangan antara kinerja klasifikasi pada kelas mayor dan minor. G-Mean yang rendah mengindikasikan performa yang buruk dalam klasifikasi kasus positif Ketika klasifikasi kelas negative diklasifikasikan dengan benar (Akosa, 2017). Adapun nilai G-Mean dihitung dengan rumus sebagai berikut.

$$G - Mean = \sqrt{sensitivity \times specificity} \quad (2.26)$$

2.6 Synthetic Minority Oversampling Technique

Synthetic Minority Oversampling Technique merupakan salah satu cara mengatasi imbalance class dengan konsep membuat data *synthetic* dengan berulang pada minority class (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). Pada data kontinu, SMOTE menambah data buatan dengan mencari nilai median kelas minor. Sementara pada kelas nominal, SMOTE menambah data buatan dengan *k-nearest neighbor* agar jumlah observasi kelas minor seimbang dengan jumlah observasi kelas mayor. *Nearest neighbor* pada kelas nominal dihitung dengan Modified Value Difference Metric (MVDM), sebuah konsep yang diusulkan oleh Cost dan Salzberg pada 1993. Jarak antar kategori dalam suatu variabel independen dapat dihitung dengan persamaan berikut.

$$\delta(V_1, V_2) = \sum_{i=1}^{h} \left| \frac{C_{1i}}{C_1} - \frac{C_{2i}}{C_2} \right|^s \qquad (2.27)$$

dimana :

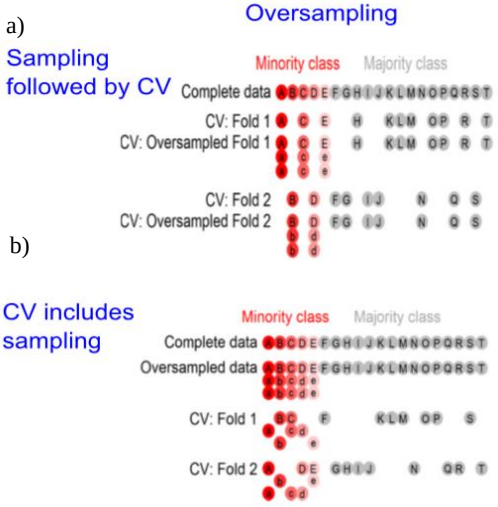
$\delta(V_1, V_2)$ = jarak kategori V_1 dan V_2
C_{1i} = jumlah V_1 yang masuk kelas respon *i*
C_{2i} = jumlah V_2 yang masuk kelas respon *i*
C_1 = jumlah V_1
C_2 = jumlah V_2
s = konstan (1 *by default*)
h = jumlah kelas pada variabel respon

Jarak antar observasi dihitung dengan persamaan berikut.

$$\Delta(X, Y) = w_x w_y \sum_{b=1}^{p} \delta(x_b, y_b)^r \qquad (2.28)$$

$\Delta(X, Y)$ = jarak antara observasi *X* dan *Y*
$w_x w_y$ = pembobot (dapat diabaikan)
p = jumlah variabel independent
$\delta(x_b, y_b)$ = jarak antara kategori *x* dan *y* pada variabel indenpenden ke-*b*
r = 1: jarak Manhattan; 2: jarak Euclidean

Penggunaan *oversampling* dan *k-fold cross validation* secara bersamaan seringkali dilakukan dengan melakukan *oversampling* terlebih dahulu kemudian pembagian data *training testing,* yang seterusnya disebut *oversampling outside fold* (OOF). Hal ini mengakibatkan seluruh data kelas minor akan tersintetis tanpa pembagian pada data *training testing.* Sehingga, pada OOF akan terbentuk data sintetis yang lebih heterogen karena *oversampling* dilakukan pada dataset awal secara menyeluruh. Adapun cara *oversampling inside fold* (OIF) adalah dengan melakukan pemisahan *fold* terlebih dahulu dan kemudian *oversampling* data training pada tiap-tiap *fold* (Blagus & Lusa, 2015). Prosedur yang dilakukan pada metode OIF hanya akan mensintetis data *training* pada masing-masing *fold,* menyebabkan terbentuknya data yang lebih homogen pada tiap *fold*. Hal ini tercantum pada ilustrasi sebagai berikut.



Gambar 2. 5 Ilustrasi Oversampling a) Inside dan b) Outside Fold

Adapun simulasi Gambar 2.5 terdapat pada Tabel 2.2 hingga 2.4.

Tabel 2. 2 Data Simulasi Oversampling

Data ke-	Var Input 1	Var Input 2	Var Input 3	Var Input 4	Var Output	Jarak VDM Data ke-8
1	0	10	12	9	0	2
2	0	10	1	9	0	2,7346
3	0	10	12	6	0	1
4	0	10	12	9	0	2
5	1	10	12	9	0	3
6	0	10	12	1	1	2
7	0	10	12	5	1	2
8	0	9	12	6	1	0

Pada Tabel 2.2 merupakan contoh data yang akan digunakan untuk melakukan *oversampling inside fold* dan *oversampling outside fold*. Kolom 6 merupakan jarak VDM masing-masing data terhadap data ke-8 yang dihitung menggunakan persamaan (2.28). Pada Tabel 2.3 berikut ini merupakan contoh apabila melakukan *oversampling inside fold*.

Tabel 2. 3 Contoh *Oversampling Inside Fold*

Fold	Data ke-	Var Input 1	Var Input 2	Var Input 3	Var Input 4	Var Output
1	1	0	10	12	9	0
	4	0	10	12	9	0
	6	0	10	12	1	1
	7	0	10	12	5	1
2	2	0	10	1	9	0
	3	0	10	12	6	0
	5	1	10	12	9	0
	8	0	9	12	6	1
	Sintetis	0	9	1	6	1
	Sintetis	1	9	12	6	1

Berdasarkan Tabel 2.3, pada *fold* 2 terbentuk dua data sintetis untuk mengimbangi jumlah data kelas 0 pada kategori output. Data sintetis yang terbentuk memiliki nilai yang tidak jauh berbeda pada setiap variabelnya dengan data ke-8 saja tanpa melihat data ke-6 dan ke-7. Hal ini yang dimaksud dengan pembentukan data yang cenderung lebih homogen pada *oversampling inside fold*.

Tabel 2. 4 Contoh Oversampling Outside Fold

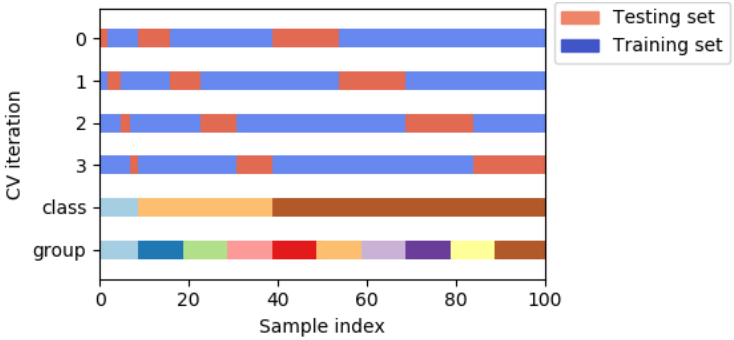
Data ke-	Var Input 1	Var Input 2	Var Input 3	Var Input 4	Var Output
1	0	10	12	9	0
2	0	10	1	9	0
3	0	10	12	6	0
4	0	10	12	9	0
5	1	10	12	9	0
6	0	10	12	1	1
7	0	10	12	5	1
8	0	9	12	6	1
Sintetis	1	9	12	6	1
Sintetis	0	10	1	5	1

Sementara itu, pada Tabel 2.4 merupakan contoh apabila dilakukan oversampling outside fold. Data sintetis yang terbentuk dapat menyerupai data ke-6, ke-7, atau ke-8.

2.7 Stratified K-fold Cross Validation

Metode validasi yang digunakan adalah *Stratified K-fold Cross Validation*. Pada *stratified k-fold cross validation* data sampel dibagi secara acak menjadi sejumlah *k* bagian dengan proporsi setiap kategori data pada tiap *fold* sama banyak dan dilakukan pengulangan sebanyak *k* kali. Nilai *k* yang sering digunakan adalah 10 karena merupakan nilai yang paling

memadai untuk mendapatkan perkiraan kesalahan terkecil (Berthold & Hand, 2003).



Gambar 2. 6 Ilustrasi *Stratified 3-folds Cross Validation*

Pada Gambar 2.6, *bar* oranye mewakilan banyaknya data testing dan *bar* biru mewakilkan data training. Sementara itu, pada bagian *class* yang diwakilkan oleh biru muda, *peach*, dan cokelat, memberikan arti banyaknya data dengan kategori tersebut. Sehingga, dapat disimpulkan bahwa pengambilan data *training* dan *testing* dengan *stratified k-fold cross validation* disesuaikan dengan banyaknya data pada masing-masing *class*. Semakin banyak data tersebut, maka akan semakin banyak data yang terambil untuk menjadi *training* dan *testing*.

2.8 *Marketing*, CRM, dan SaaS

Sub-bab ini menjelaskan mengenai tinjauan non-statistik yaitu *marketing*, *Customer Relationship Management* (*CRM*), dan *Software as a Service* (SaaS).

2.8.1 Marketing

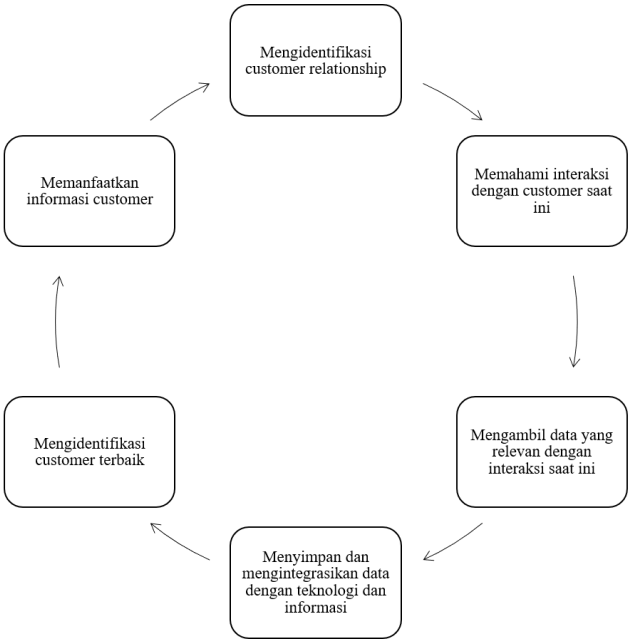
Marketing atau pemasaran merupakan suatu kegiatan dan proses dalam menyampaikan dan bertukar penawaran yang

memiliki nilai bagi pelanggan, klien, mitra, dan masyarakat pada umumnya (Lamb, et al., 2002). *Marketing* memiliki dua sisi. Pertama, berfokus pada filosofi, sikap, perspektif, dan orientasi manajemen yang bertujuan untuk kepuasan pelanggan (*customer satisfaction).* Kedua, merupakan kegiatan dan serangkaian proses yang digunakan untuk menerapkan filosofi yang dimiliki sehingga, *marketing* merupakan bagian yang tidak terlepas dari departemen lainnya dalam suatu organisasi. Kegiatan marketing di dalamnya termasuk dalam periklanan, data penjualan, dan pengiriman produk ke konsumen. Beberapa kegiatan *marketing* dilakukan oleh afiliasi atas nama perusahaan. Seorang profesional yang bekerja di bidang *marketing* dan promosi dalam suatu perusahaan berusaha untuk mendapatkan perhatian audiens yang potensial melalui iklan. Suatu promosi ditargetkan untuk audiens tertentu yang dapat melibatkan selebriti, slogan menarik, kemasan yang mengesankan, maupun desain grafis yang terlihat di media secara keseluruhan (Twin, 2019).

Secara kontras, perusahaan yang berfokus pada penjualan akan berupaya untuk meningkatkan volume penjualan melalui promosi yang intensif. Sebaliknya, perusahaan yang berorientasi pada pasar akan menganggap promosi hanyalah satu dari empat dasaran *marketing mix* yang terdiri dari: produk, harga, promosi, dan tempat (atau distribusi). Selain itu, perusahaan yang berfokus pada pasar mempercayai bahwa *marketing* bukan hanya tanggung jawab satu departemen, melainkan melibatkan seluruh sumber daya di perusahaan tersebut untuk menciptakan, membentuk komunikasi, dan memberikan layanan yang unggul (Lamb, et al., 2002).

2.8.2 Customer Relationship Management (CRM)

Customer Relationship Management (seterusnya akan disebut *CRM*) merupakan suatu strategi bisnis yang dirancang untuk mengoptimalkan profitabilitas, penghasilan, dan kepuasan pelanggan dengan berfokus pada kelompok pelanggan tepat sasaran (Lamb, et al., 2002). Perusahaan yang menganut sistem *CRM* pada umumnya memiliki filosofi berfokus pada pelanggan. Dengan begitu, perusahaan dapat menyesuaikan produk dan layanan yang dimiliki melalui interaksi antara pelanggan dan perusahaan. Sekilas *CRM* terlihat seperti strategi yang sederhana, bahkan layanan pelanggan (*customer service*) merupakan bagian dari proses *CRM*. Hal tersebut hanya sebagian kecil dari pendekatan yang sepenuhnya terintegrasi untuk membangun hubungan pelanggan (Lamb, et al., 2002). *CRM* sering digambarkan sebagai siklus yang berulang dalam membangun hubungan pelanggan seperti yang ditampilkan pada gambar berikut.



Gambar 2. 7 Ilustrasi Model CRM
Sumber: (Lamb, et al., 2002)

Dalam memulai siklus *CRM*, perusahaan harus mengidentifikasi terlebih dahulu bagaimana hubungan perusahaan dan pelanggan dengan mempelajari siapa yang akan jadi pelanggan dari produk yang dibuat, di mana pelanggan berada, bahkan informasi yang lebih rinci terkait produk yang saat ini sedang marak digunakan oleh pelanggan.

2.8.3 Software as a Service (SaaS)

Software as a Service (seterusnya disebut *SaaS*) merupakan perangkat lunak berbasis internet sebagai pengelolaan data dan aplikasi (*cloud computing*) yang dimiliki dan dijalankan secara *online* tanpa terbatas jarak oleh satu maupun beberapa *provider* (Smyrnova, 2019). *SaaS* dapat berupa aplikasi bagi pengguna. Idealnya, SaaS digunakan oleh

perusahaan yang tidak ingin melakukan perawatan infrastuktur, *platforms*, dan perangkat lunak (Red Hat, n.d.). Contoh *SaaS* yang termasuk layanan *interface* konsumen yaitu Google Docs dan Microsoft 365.

(Halaman ini sengaja dikosongkan)

BAB III
METODOLOGI PENELITIAN

3.1 Sumber Data

Data yang digunakan pada penelitian ini merupakan data sekunder dari database PT "X" mengenai CRM selama 2018 dan 2019. Data diakses pada 7 Januari 2020 melalui *platform* yang digunakan PT "X" untuk menyimpan data secara general. Unit sampling yang digunakan dalam penelitian ini adalah *leads*, (merupakan suatu badan usaha) PT "X" sebanyak 10.808 *leads*.

3.2 Variabel Penelitian dan Struktur Data

Variabel penelitian yang digunakan pada penelitian ini merupakan *leads* yang terekam pada sistem CRM PT "X" baik melalui sumber digital maupun non-digital.

Tabel 3. 1 Variabel Penelitian CRM PT "X"

Variabel	Nama Variabel	Skala Data
Y	*Deal Status*	Nominal
X_1	*Special Project*	Nominal
X_2	*Number of Employee*	Rasio
X_3	*Industry*	Nominal
X_4	*Introduce Month*	Nominal
X_5	*Team Leader*	Nominal
X_6	*MRR*	Rasio
X_7	*Contract Length*	Nominal
X_8	*Source*	Nominal

Penjelasan setiap variabel disajikan dalam tabel berikut.

Tabel 3. 2 Definisi Operasional Variabel

Nama Variabel	Deskripsi	Rincian
Deal Status (Y)	*Deal* status menjelaskan status transaksi dari *leads.*	0: *Deal* 1: *Cancelled/ Lost Deal*
Special Project (X_1)	*Special Project* menjelaskan ada atau tidaknya *custom* atau permintaan khusus dari calon klien terhadap produk yang ditawarkan (modifikasi pada *software*).	TRUE: terdapat custom pada produk FALSE: Tidak terdapat custom
Number of Employee (X_2)	Jumlah karyawan pada perusahaan leads.	-
Industry (X_3)	Variabel *industry* menjelaskan sektor bisnis *leads* PT "X".	Industry_1 hingga Industry_14
Introduce Month (X_4)	Bulan PT "X" melakukan *pitching* pada *leads*.	Januari, Februari, Maret, April, Mei, Juni, Juli, Agustus, September, Oktober, November, Desember
Team Leader (X_5)	Pemimpin grup yang terdiri atas beberapa *sales*.	TeamLeader_1 hingga TeamLeader_10
MRR (X_6)	*Monthly Recurring Revenue*, pendapatan yang dihasilkan dari produk yang digunakan klien.	Dalam satuan ribu rupiah.
Contract Length (X_7)	Lamanya kontrak berlangganan akan	-

Tabel 3. 2 Definisi Operasional Variabel (Lanjutan)

Nama Variabel	Deskripsi	Rincian
	berlangsung dalam satuan bulan.	
Lead Source (X_8)	*Lead Source* merupakan variabel yang menjelaskan sumber *leads* didapatkan.	Source_1 hingga Source_12

Berdasarkan variabel-variabel yang digunakan, adapun struktur data dari variabel di atas adalah sebagai berikut.

Tabel 3. 3 Struktur Data Variabel Penelitian

Variabel Respon (Y)	Variabel Prediktor (X)			
	X_1	X_2	⋯	X_8
Y_1	$X_{1,1}$	$X_{1,2}$	…	$X_{1,8}$
Y_2	$X_{2,1}$	$X_{2,2}$	…	$X_{2,8}$
Y_3	$X_{3,1}$	$X_{3,2}$	…	$X_{3,8}$
⋮	⋮	⋮	⋱	⋮
Y_n	$X_{n,1}$	$X_{n,2}$	…	$X_{n,8}$

3.3 Langkah Analisis

Langkah analisis digunakan untuk menggambarkan tahap-tahap penelitian yang akan dilakukan secara urut. Langkah analisis yang digunakan pada penelitian ini adalah sebagai berikut.

1. Melakukan *pre-processing* dan eksplorasi data sebagai berikut:
   a. Pemisahan data yang akan digunakan untuk analisis yaitu hanya klien yang sudah sampai tahap *deal* dan *cancelled/ lost deal*.
   b. Penghapusan pada *duplicate data*.

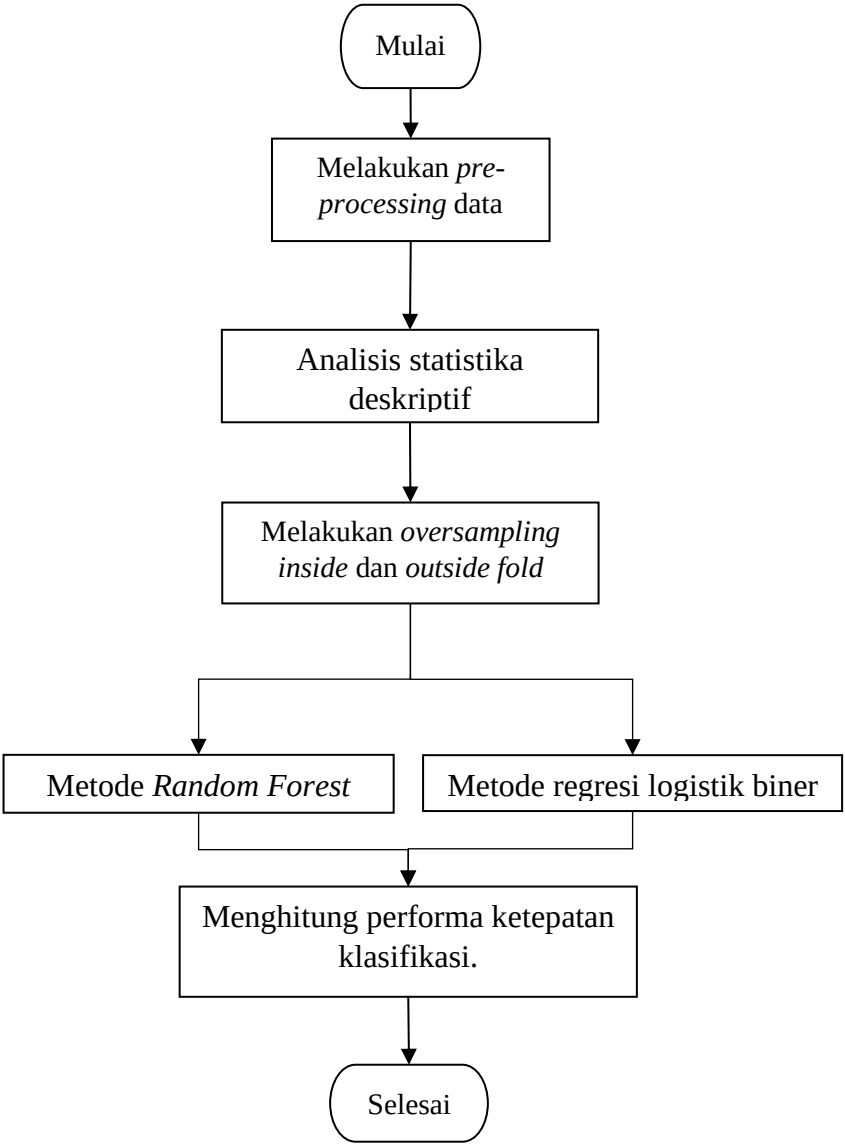
c. Imputasi pada *cell* kosong menggunakan *mean* dan modus.
d. Kode ulang variable input dan respon.
e. Melakukan analisis statistika deskriptif pada variabel input dan respon.

2. Selanjutnya, melakukan *feature selection* dengan algotirma Boruta dan klasifikasi menggunakan metode *Random Forest* dengan *oversampling inside fold* dan *oversampling outside fold*. Adapun beberapa tahapan dalam melakukan klasifikasi hingga mendapatkan hasil ketepatan prediksi adalah sebagai berikut.
   a. Membagi data menjadi sepuluh bagian sesuai aturan *stratified k-fold cross validation*.
   b. Melakukan *balancing* menggunakan metode SMOTE pada data *training* di dalam *fold* untuk *oversampling inside fold* dan melakukan *oversampling* sebelum membagi data menjadi sepuluh bagian untuk *oversampling outside fold.*
   c. Melakukan klasifikasi dengan metode *Random Forest*.
   d. Menghitung performa ketepatan klasifikasi akurasi, AUC, g-mean, sensitivitas, dan spesivisitas.
3. Sementara itu, untuk mendapatkan hasil klasifikasi menggunakan regresi logistik biner adalah sebagai berikut.
   a. Melakukan pengodean ulang pada variabel kategorik sebagai variabel *dummy* pada variabel prediktor.
   b. Membagi data menjadi sepuluh bagian sesuai aturan *stratified k-fold cross validation*.
   c. Melakukan *balancing* menggunakan metode SMOTE pada data *training* di dalam *fold* untuk *oversampling inside fold* dan melakukan

oversampling sebelum membagi data menjadi sepuluh bagian untuk *oversampling outside fold.*

d. Melakukan klasifikasi dengan metode regresi logistik biner.
e. Menghitung performa ketepatan klasifikasi akurasi, AUC, g-mean, sensitivitas, dan spesivisitas.
f. Melakukan estimasi parameter.
g. Melakukan uji parsial dan uji serentak.
h. Menghitung *odds ratio* untuk interpretasi.

4. Melakukan perbandingan akurasi klasifikasi.
   a. Membandingkan hasil ketepatan klasifikasi *oversampling inside fold* dan *oversampling outside fold* pada *Random Forest* dan regresi logistik biner.
   b. Membandingkan hasil ketepatan klasifikasi kelompok data *imbalance* dan *oversampling inside fold* dan *oversampling outside fold* pada *Random Forest* dan regresi logistik biner.



Gambar 3. 1 Diagram Alir Penelitian

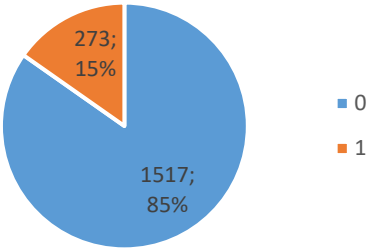
BAB IV
ANALISIS DAN PEMBAHASAN

Bab analisis dan pembahasan ini membahas mengenai deskripsi karakteristik data, faktor-faktor yang diduga memengaruhi keputusan *deal* atau *cancelled deal* suatu perusahaan klien PT "X", klasifikasi klien PT "X" yang melakukan *deal* atau *cancelled deal* menggunakan metode andom forest dan regresi logistik biner.

4.1 Analisis Karakteristik *Leads*

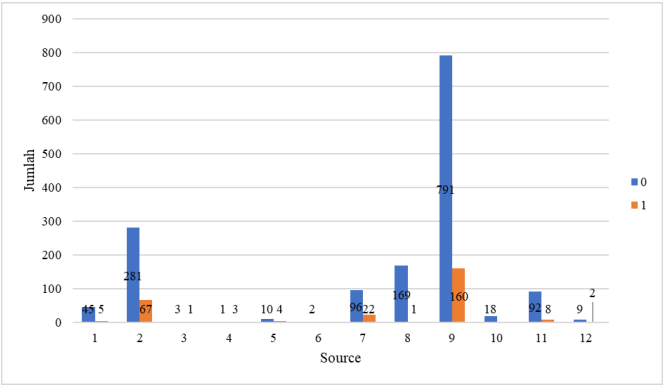
Data yang digunakan merupakan data klien PT "X" yang melakukan transaksi berlangganan produk (*deal*) dan pembatalan berlangganan produk (*cancelled/ lost deal*). Data dipilih berdasarkan *stage* yang sudah melalui proses tanda tangan kontrak kemudian melanjutkan transaksi dan melakukan pembatalan atau tidak ada respon (*lost*) setelah melakukan tanda tangan. Penyaringan data berdasarkan *stage* menyisakan sebanyak 1790 data.

Analisis karakteristik data dilakukan untuk melihat gambaran data. Eksplorasi data tahap awal dilakukan pada beberapa variabel awal seperti variabel respon dan karakter perusahaan *client*.



Gambar 4. 1 Persentase *Leads* Deal dan Cancelled/ Lost

Pada analisis data klien per Januari 2020, terdapat 15% klien yang melakukan pembatalan deal terhadap produk PT "X" dan 85% tetap melanjutkan transaksi untuk menggunakan produk. Walaupun klien yang melakukan transaksi hingga akhir masih terbilang jauh di atas yang melakukan pembatalan, namun hal ini terbilang cukup banyak secara jumlah yakni sebanyak 273 perusahaan.



Gambar 4. 2 Chart Jumlah *Leads* Berdasarkan Source

Variabel *source* menjelaskan sumber data klien PT "X". Pada Gambar 4.2, *bar* dengan warna biru menandakan *leads*

dengan transaksi berhasil dan *bar* oranye menandakan *leads* dengan pembatalan transaksi. Terlihat bahwa tiga sumber terbanyak adalah dari *Source9*, *Source2*, dan *Source8*. Sementara itu sumber dengan pembatalan transaksi terbanyak adalah *Source9*, *Source2*, dan *Source7*.

Tabel 4. 1 Nilai Median dan Rata-rata Pada Variabel Employee

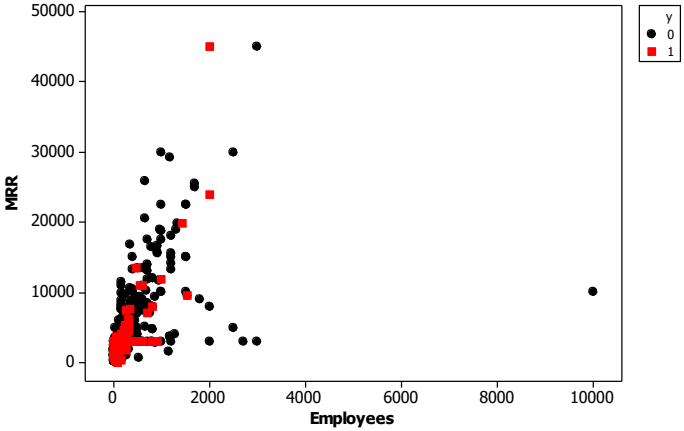
Employee	Median	Rata-rata
Deal (0)	100	166
Lost (1)	100	156

Berdasarkan Tabel 4.1, nilai tengah diantara kedua kelompok data sama-sama berada pada angka 100 karyawan. Sementara itu, terlihat bahwa tidak terdapat perbedaan yang jauh berbeda antara rata-rata jumlah karyawan pada *leads deal* dan *lost* yaitu hanya selisih sebanyak sepuluh karyawan.

Tabel 4. 2 Nilai Median dan Rata-rata Pada Variabel MRR

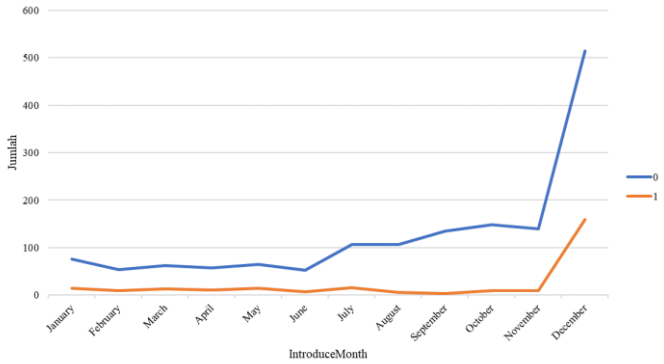
MRR	Median	Rata-rata
Deal (0)	2250	2997,9994
Lost (1)	2250	3103,8782

Apabila ditinjau dari nilai tengah dan rata-rata MRR, didapatkan nilai tengah yang sama dan perbedaan rata-rata yang tidak terlalu jauh, yaitu hanya selisih 105,8788 atau sebesar Rp105.878,8.



Gambar 4. 3 Scatterplot MRR dan Employee

Gambar 4.3 merupakan scatterplot jumlah karyawan dan MRR pada kelompok data dengan kategori *deal* (0) dan *lost* (1). Pada Gambar 4.3 terlihat bahwa titik hitam yang merupakan data dengan kategori 0 lebih menyebar dibandingkan titik merah.



Gambar 4. 4 *Line Chart* Variabel *IntroduceMonth*

IntroduceMonth atau bulan dimana produk mulai diperkenalkan kepada *leads* pada Gambar 4.4 terlihat

meningkat pada Q4 dan memiliki jumlah terbanyak pada Desember untuk *leads deal* (kategori 0) maupun *lost* (kategori 1).

4.2 Klasifikasi *Leads* PT "X" dengan Metode *Random Forest*

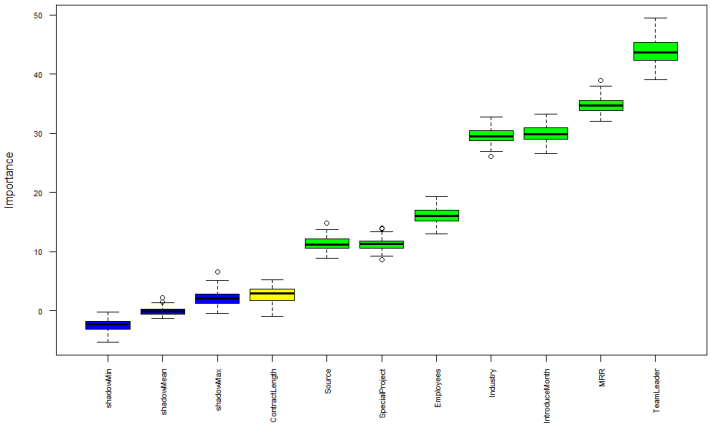
Setelah dilakukan analisis karakteristik data, selanjutnya akan dilakukan analisis klasifikasi klien PT "X" melakukan *deal* hingga akhir atau *lost* pada saat proses berlangsung dengan metode *Random Forest* dan regresi logistik biner.

Tahap pertama yang dilakukan sebelum melakukan klasifikasi yaitu dengan melakukan seleksi variabel yang berguna untuk mengetahui variabel yang dianggap penting dalam proses klasifikasi sehingga dapat menghasilkan ketepatan klasifikasi yang lebih baik.

Tabel 4. 3 Hasil *Impurity* dan Z-Score Boruta

Feature	*meanImp*	*medianImp*	*minImp*	*maxImp*	*normHits*	*Decision*
Employees	16,0051	15,8970	13,1388	19,0659	1,0000	*Confirmed*
MRR	34,8349	34,7248	30,8132	38,4583	1,0000	*Confirmed*
SpecialProject	11,1040	11,1868	8,9973	13,5294	1,0000	*Confirmed*
Industry	29,5824	29,4693	26,4626	33,7457	1,0000	*Confirmed*
IntroduceMonth	29,7222	29,6465	26,7903	34,3595	1,0000	*Confirmed*
TeamLeader	44,0784	44,0173	40,5585	48,1808	1,0000	*Confirmed*
ContractLength	2,5327	2,5788	-0,0220	5,0168	0,4646	*Tentative*
Source	11,0962	11,1820	8,3689	14,1902	1,0000	*Confirmed*

Pada Tabel 4.3 didapatkan nilai *impurity* dan Z-score masing-masing variabel. Kolom *normHits* merupakan acuan variabel tersebut dikatakan *important*, *tentative*, atau *not important*. Visualisasi Tabel 4.3 terdapat pada Gambar 4.5.



Gambar 4. 5 Seleksi Variabel Metode *Random Forest*

Hasil seleksi variabel pada Gambar 4.5 memberikan informasi *variable level of importance*. Boxplot dengan warna hijau berarti tidak perlu dilakukan penghapusan variabel, kuning tentative, dan merah lebih baik dihapus. Gambar 4.5 menunjukkan variabel ContractLength sebagai variabel tentative. Pada penelitian kali ini penulis memilih untuk mempertahankan seluruh variabel sehingga tidak perlu dilakukan penghapusan variabel.

Setelah melakukan seleksi variabel, pada metode *Random Forest* terpadat proses pembentukan pohon klasifikasi yang diperlukan variabel sebagai pemilah. Suatu variabel berskala nominal dengan kategori sebanyak *G* maka memiliki kemungkinan pemilah sebanyak 2^{G-1}-1. Jumlah kemungkinan pemilah pada masing-masing variabel untuk membentuk pohon klasifikasi klien yang melakukan *deal* atau *lost* ditampilkan pada Tabel 4.4.

Tabel 4. 4 Jumlah Kemungkinan Pemilah pada Variabel Input

Nama Variabel	Jumlah Kategori	Kemungkinan Pemilah
Special Project	2	1
Industry	14	8191
Introduce Month	12	2047
Team Leader	10	511
Contract Length	2	1
Source	12	2047

Setelah melakukan perhitungan jumlah kemungkinan pemilah, selanjutnya adalah menghitung indeks gini. Berikut merupakan contoh perhitungan indeks gini pada variabel *special project* data *train*.

Tabel 4. 5 Ilustrasi Pemilihan pada *Node Special Project*

Special Project	Prediksi		Total
	Deal	*Lost*	
FALSE	795	192	987
TRUE	571	53	624
Total	1366	245	1611

Tabel 4.5 menunjukkan hasil *cross-tabulation* pada variabel *SpecialProject* data *training*. Dengan membagi data menjadi sepuluh bagian menggunakan teknik *k-fold cross validation* akan terbentuk sembilan bagian data *training* dan menyisakan satu bagian pada data *testing*. Sehingga, pada 1790 data terbagi menjadi 1611 data *training* dan 179 data *testing*. Perhitungan indeks gini sesuai persamaan (2.15) pada masing-masing node kanan dan kiri adalah sebagai berikut.

$$imp(t_L) = 2 \times \frac{795}{986} \times \frac{192}{986} = 0{,}3133$$

$$imp(t_R) = 2 \times \frac{571}{624} \times \frac{53}{624} = 0{,}1554$$

Kemudian menghitung *goodness of split* untuk evaluasi pemilahan yang dilakukan oleh pemilah *s* pada simpul *t* berdasarkan persamaan (2.16). Pada variabel *special project* hanya terdapat satu kemungkinan pemilah, maka hanya akan ada satu nilai *goodness of split*.

$$\begin{aligned}\big(\phi(s,t)\big) &= imp(st) \\ &= 0{,}1289 - \left(\frac{986}{1611} \times 0{,}3133\right) \\ &\quad - \left(\frac{624}{1611} \times 0{,}1554\right) = -0{,}1232\end{aligned}$$

Analisis klasifikasi menggunakan metode *Random Forest* pada data *training imbalance* menghasilkan *confusion matrix* sebagai berikut.

Tabel 4. 6 *Confusion Matrix Fold* ke-10 *Data Training Imbalance*

Aktual	Prediksi	
	Deal	*Lost*
Deal	1365	1
Lost	0	245

Pada Tabel 4.6 merupakahan salah satu *confusion matrix* dari data *imbalance* yang menghasilkan nilai akurasi sebesar 99,9379%. Berikut merupakan nilai AUC, G-Mean, akurasi, sensitivitas, dan spesifisitas pada data *imbalance* dengan nilai *cp* optimum.

Tabel 4. 7 Ketepatan Klasifikasi RF Data Training Imbalance

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	99,9980	99,7600	99,8759	99,5935	99,9267
2	99,9880	98,8697	99,5034	97,9675	99,7802
3	99,9880	98,7006	99,5034	97,5610	99,8535

Tabel 4. 7 Ketepatan Klasifikasi RF Data Training Imbalance (Lanjutan)

4	99,9880	98,8697	99,5034	97,9675	99,7802
5	99,9880	98,8697	99,5034	97,9675	99,7802
6	99,9890	98,9423	99,6276	97,9675	99,9267
7	99,9880	98,8697	99,5034	97,9675	99,7802
8	99,9870	98,6956	99,5034	97,5510	99,8536
9	99,9870	98,6956	99,5034	97,5510	99,8536
10	99,9850	99,9634	99,9379	100,0000	99,9268
Median	99,9880	98,9060	99,5034	97,9675	99,8535
Rata-rata	99,9886	99,0244	99,5965	98,2094	99,8462

Pada Tabel 4.7 menunjukkan hasil ketepatan klasifikasi klien PT "X" melakukan pembatalan transaksi dengan menggunakan metode *Random Forest* menghasilkan rata-rata AUC sebesar 99,9886%, G-Mean 99,0244%, dan akurasi sebesar 99,5965%. Artinya, model *Random Forest* yang terbentuk sudah dapat menglasifikasikan data tersebut dengan sangat baik.

Tabel 4. 8 Ketepatan Klasifikasi RF Data Testing Imbalance

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	51,6326	32,3314	81,5642	11,1111	94,0789
2	84,6735	71,2941	91,0615	51,8519	98,0263
3	88,2310	83,4794	91,0615	74,0741	94,0789
4	87,9386	78,8254	93,2961	62,9630	98,6842
5	80,9454	68,1130	83,7989	51,8519	89,4737
6	68,9815	42,3194	84,9162	18,5185	96,7105
7	75,0365	0,0000	84,9162	0,0000	100,0000
8	51,8094	26,0085	81,0056	7,1429	94,7020
9	82,7342	26,5485	84,3575	7,1429	98,6755
10	23,4153	13,6693	44,6927	3,5714	52,3179
Median	77,9910	37,3254	84,6369	14,8148	95,7063

Tabel 4. 8 Ketepatan Klasifikasi RF Data Testing Imbalance (Lanjutan)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
Rata-rata	69,5398	44,2589	82,0670	28,8228	91,6748

Tabel 4.8 merupakan ketepatan klasifikasi data testing pada kondisi imbalance. Jika dibandingkan dengan hasil pada Tabel 4.7, terdapat penurunan pada nilai sensitivitas yang berarti terdapat banyak kesalahan pada klasifikasi data kelas minor. Oleh karena itu dilakukan penanganan data *imbalance* dengan harapan mendapatkan hasil yang lebih baik pada klasifikasi. Penanganan data *imbalance* yang dilakukan adalah dengan melakukan replikasi data training pada kelompok data minoritas agar jumlahnya seimbang dengan kelompok data mayoritas. Metode ini dinamakan *Synthetic Minority Oversampling Technique-Nominal Continuous* (SMOTE-NC). Pada analisis ini akan dilakukan dua perlakuan terhadap metode SMOTE-NC, yaitu dengan melakukan *oversampling inside fold* dan *outside fold*.

Tahap pertama dalam melakukan SMOTE-NC adalah dengan mengambil data kelas minoritas secara acak.

Tabel 4. 9 Data Ilustrasi SMOTE-NC

Variabel	Data	
	x_{2538}	x_{471}
Employees	200	25
MRR	2000	625
SpecialProject	0	0
Industry	1	1
IntroduceMonth	12	12
TeamLeader	1	1
ContractLength	12	12
Source	2	9

Tabel 4.9 merupakan nilai dan kategori pada masing-masing variabel predictor pada observasi ke 2538 dan 471. Perbedaan antara kedua observasi pada variabel kategorik yaitu pada variabel *Source*. Berdasarkan persamaan (2.26) maka jarak VDM bernilai 0 untuk nilai yang sama. Jarak VDM pada variabel *Source* dihitung dengan nilai pada *cross tabulation* berikut.

Tabel 4. 10 *Cross Tabulation* Data Training SMOTE-NC

Source	*Deal*		Total
	Deal	*Lost*	
1	41	4	45
2	**264**	**63**	**327**
3	3	1	4
4	1	3	4
5	8	4	12
6	1	0	1
7	86	20	106
8	139	1	140
9	**708**	**140**	**848**
10	17	0	17
11	90	7	97
12	8	2	10

Hasil *cross tabulation* pada Tabel 4.10 digunakan untuk menghitung jarak VDM variabel *Source* dengan nilai 2 dan 9 sebagai berikut.

$$\delta(2,9) = \left|\frac{264}{327} - \frac{708}{848}\right| + \left|\frac{63}{327} - \frac{140}{848}\right| = 0{,}0551$$

Berikut merupakan hasil jarak antar kategori pada masing-masing variabel kategorik.

Tabel 4. 11 Jarak VDM Observasi 2538 dan 471

Variabel	$\delta(x_{2538}, x_{471})$
SpecialProject	0
Industry	0
IntroduceMonth	0
TeamLeader	0
ContractLength	0
Source	0,0551324

Sementara itu, untuk menghitung jarak VDM antara x_{2538} dan x_{471}, digunakan persamaan (2.26) sebagai berikut.

$$\Delta(x_{2538}, x_{471}) = \sum_{b=1}^{6} \delta(x_{2538,b}, x_{471,b})^2 = 0{,}00304$$

Setelah dilakukan perhitungan seperti di atas, ketepatan klasifikasi pada data *training* setelah dilakukan *oversampling inside fold* ditampilkan pada Tabel 4.12.

Tabel 4. 12 Ketepatan Klasifikasi RF Data Training *Oversampling Inside Fold* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	99,8633	99,5918	99,5918	99,6735	99,5102
2	99,9090	99,6326	99,6327	99,5102	99,7551
3	99,8638	99,5106	99,5106	99,5922	99,4290
4	99,8471	99,5514	99,5514	99,5922	99,5106
5	99,9458	99,6737	99,6737	99,5922	99,7553
6	99,9152	99,7552	99,7553	99,9184	99,5922
7	99,8446	99,5106	99,5106	99,5922	99,4290
8	99,8456	99,5106	99,5106	99,5106	99,5106
9	99,8415	99,5106	99,5106	99,5106	99,5106
10	99,9069	99,7552	99,7553	99,9184	99,5922
Median	99,8636	99,5716	99,5716	99,5922	99,5106
Rata-rata	99,8783	99,6002	99,6003	99,6410	99,5595

Tabel 4.12 merupakan hasil ketepatan klasifikasi pada data training *oversampling inside fold* dengan *Random Forest*. Didapatkan hasil lebih dari 99% pada seluruh pengukuran ketepatan klasifikasi. Sehingga, dapat dikatakan bahwa model RF-OIF sudah dapat memprediksi data training dengan sangat baik.

Tabel 4. 13 Ketepatan Klasifikasi RF Data Testing *Oversampling Inside Fold* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	51,4498	76,6318	78,2123	73,8095	79,5620
2	89,6199	52,4346	70,9497	33,3333	82,4818
3	90,1438	70,7228	79,3296	58,1395	86,0294
4	90,0585	67,6961	86,0335	46,5116	98,5294
5	87,7924	29,5312	59,7765	11,6279	75,0000
6	78,9961	32,6916	72,6257	11,6279	91,9118
7	86,1598	88,4198	91,0615	83,7209	93,3824
8	55,7829	69,9325	84,9162	51,1628	95,5882
9	81,8236	83,0341	88,2682	74,4186	92,6471
10	25,9697	4,8928	8,3799	2,3256	10,2941
Median	83,9917	68,8143	78,7710	48,8372	88,9706
Rata-rata	73,7796	57,5987	71,9553	44,6678	80,5426

Apabila hasil pada Tabel 4.10 dibandingkan dengan hasil ketepatan klasifikasi data testing *imbalance* pada Tabel 4.8, terdapat kenaikan rata-rata sensitivitas sebesar 15,845%, kenaikan rata-rata AUC sebesar 4,2399%, dan kenaikan rata-rata g-mean sebesar 13,3398%. Artinya, terdapat peningkatan performa ketepatan klasifikasi pada data kelas minor setelah dilakukan *balancing*. Sebagai perbandingan, hasil ketepatan klasifikasi dengan *oversampling outside fold* ditampilkan pada tabel berikut.

Tabel 4. 14 Ketepatan Klasifikasi RF Data Training *Oversampling Outside Fold* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	100,0000	99,5227	99,5238	100,0000	99,0476
2	100,0000	99,4859	99,4872	100,0000	98,9744
3	100,0000	99,4490	99,4505	100,0000	98,9011
4	100,0000	99,4494	99,4505	99,9267	98,9744
5	100,0000	99,4490	99,4508	100,0000	98,9011
6	100,0000	99,4490	99,4508	100,0000	98,9011
7	100,0000	99,4490	99,4508	100,0000	98,9011
8	100,0000	99,4494	99,4508	100,0000	98,9019
9	100,0000	99,4494	99,4508	100,0000	98,9019
10	100,0000	99,9268	99,9268	99,9267	99,9268
Median	100,0000	99,4494	99,4508	100,0000	98,9019
Rata-rata	100,0000	99,5080	99,5093	99,9853	99,0331

Seperti halnya dengan hasil RF-OIF pada Tabel 4.12, Tabel 4.14 menunjukkan rata-rata hasil ketepatan klasifikasi pada seluruh pengukuran di atas 99%. Sehingga, model yang dibentuk sudah dapat memprediksi data training dengan baik. Sementara itu, hasil ketepatan klasifikasi pada data testing sebagai berikut.

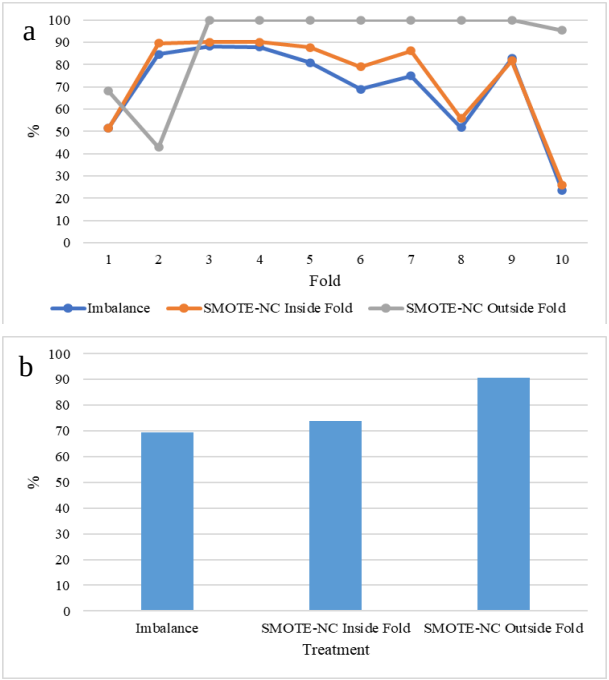
Tabel 4. 15 Ketepatan Klasifikasi RF Data Testing *Oversampling Outside Fold* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	68,1094	0,0000	50,0000	0,0000	100,0000
2	42,9320	47,1211	56,9079	25,0000	88,8158
3	100,0000	98,0064	98,0263	100,0000	96,0526
4	100,0000	100,0000	100,0000	100,0000	100,0000
5	100,0000	96,3136	96,3696	100,0000	92,7632
6	100,0000	100,0000	100,0000	100,0000	100,0000
7	100,0000	96,3136	96,3696	100,0000	92,7632

Tabel 4. 15 Ketepatan Klasifikasi RF Data Testing Oversampling Outside Fold (%) (Lanjutan)

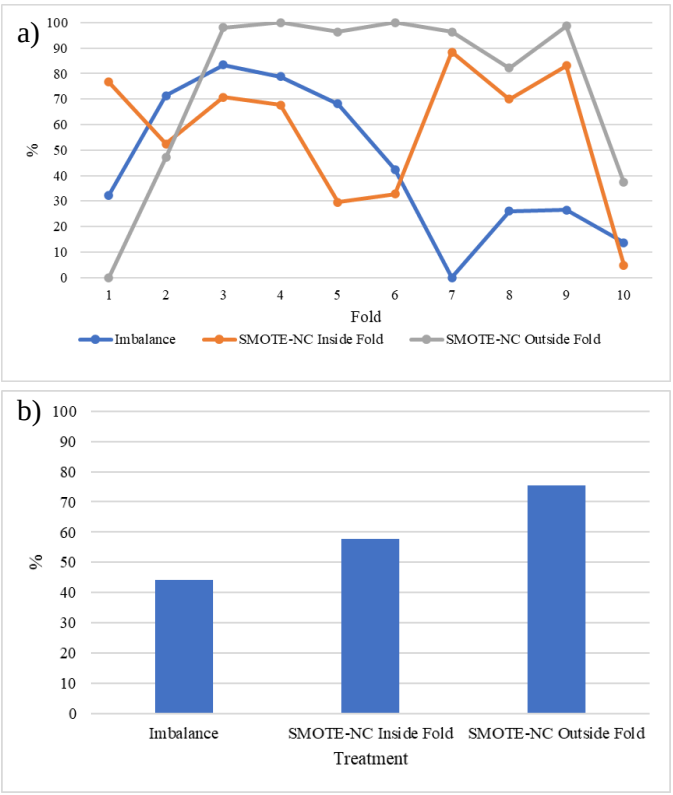
Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
8	100,0000	82,1886	83,8284	100,0000	67,5497
9	100,0000	98,6666	98,6799	100,0000	97,3510
10	95,3403	37,2925	57,0957	100,0000	13,9073
Median	100,0000	96,3136	96,3696	100,0000	94,4079
Rata-rata	90,6382	75,5902	83,7277	82,5000	84,9203

Pada kasus ini, dengan melakukan *balancing* data di luar *fold* menghasilkan kenaikan rata-rata sensitivitas sebesar 53,6772%, kenaikan rata-rata AUC sebesar 21,0984%, dan kenaikan rata-rata g-mean sebesar 31,3313%. Apabila dilihat pada nilai masing-masing *fold*, terdapat beberapa nilai dengan performa ketepatan klasifikasi sebesar 100%. Berdasarkan hasil data testing pada Tabel 4.15 memberikan kesimpulan bahwa model yang terbentuk sudah dapat memberikan ketepatan klasifikasi yang baik pada data validasi.



Gambar 4. 6 a) Nilai AUC Berdasarkan Fold; b) Rata-rata Nilai AUC Data Testing Metode Random Forest

Gambar 4.6 poin a) merupakan grafik dari nilai AUC pada data testing *imbalance*, *oversampling inside fold*, dan *oversampling outside fold*. Hasil *oversampling inside fold* memperlihatkan garis yang mengikuti pola hasil *imbalance* dengan peningkatan. Sementara hasil *oversampling outside fold* cenderung berada pada posisi 100%. Sementara itu, pada poin b) menggambarkan rata-rata nilai AUC pada data *imbalance*, *oversampling inside fold*, dan *oversampling outside fold.* Peningkatan performa klasifikasi setelah dilakukan *oversampling inside fold* sebesar 4,24% dan perningkatan setelah *oversampling outside fold* sebesar 21,1%.



Gambar 4. 7 a) Nilai G-Mean Berdasarkan *Fold*; b) Rata-rata Nilai G-Mean Data Testing Metode *Random Forest*

Nilai g-mean pada data imbalance, *oversampling inside fold*, dan *oversampling outside fold* dirangkum pada Gambar 4.7 poin a). Hasil *oversampling* di dalam *fold* mengikuti pola data *imbalance* dengan nilai lebih tinggi maupun lebih rendah, sementara hasil *oversampling* di luar *fold* cenderung mendekati 100%. Kenaikan nilai rata-rata g-mean dengan dilakukan *oversampling inside fold* sebesar 13,33% dan kenaikan nilai rata-rata setelah dilakukan *oversampling outside fold* sebesar 31,33%.

4.3 Klasifikasi *Leads* PT "X" dengan Metode Regresi Logistik Biner

Pada klasifikasi menggunakan metode regresi logistik biner ini dibentuk variabel *dummy* untuk variabel kategorik agar dapat memberikan hasil yang lebih spesifik terkait kelompok *leads* tertentu dengan kemungkinan yang tinggi terhadap pembatalan transaksi. Hasil klasifikasi pada data training *imbalance* menggunakan metode regresi logistik biner adalah sebagai berikut.

Tabel 4. 16 Ketepatan Klasifikasi Regresi Logistik Biner Data Training Imbalance (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	87,8598	56,3753	87,7716	32,5203	97,7289
2	85,7914	46,3171	86,1577	21,9512	97,7289
3	85,3167	43,7172	85,9714	19,5122	97,9487
4	83,7550	43,7499	86,0956	19,5122	98,0952
5	85,2119	50,9302	87,2129	26,4228	98,1685
6	85,8257	49,2645	86,7163	24,7967	97,8755
7	85,8087	53,1295	87,2750	28,8618	97,8022
8	85,8933	52,8620	87,2750	28,5714	97,8038
9	85,4143	50,5458	86,9025	26,1224	97,8038
10	88,9052	62,2892	88,3302	40,0000	96,9985
Median	85,8001	50,6908	87,0577	26,2726	97,8038
Rata-rata	85,9782	51,2208	86,9708	26,8271	97,7954

Berbeda dengan hasil klasifikasi dengan metode *Random Forest* pada data training jumlah data *imbalance*, hasil menggunakan metode regresi logistik biner pada Tabel 4.16 tidak memperlihatkan hasil sebesar metode *Random Forest* walaupun pada data *training*. Nilai sensitivitas dan g-mean yang tidak terlalu tinggi menunjukkan terdapat kesalahan

klasifikasi pada data kelas minor. Hasil ketepatan klasifikasi menggunakan data *testing* sebagai berikut.

Tabel 4. 17 Ketepatan Klasifikasi Regresi Logistik Biner Data Testing *Imbalance* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	50,0609	25,9329	78,2123	7,4074	90,7895
2	88,3285	76,4241	88,2682	62,9630	92,7632
3	89,4250	82,4958	92,7374	70,3704	96,7105
4	98,2700	57,7350	89,9441	33,3333	100,0000
5	84,0887	37,9802	84,9162	14,8148	97,3684
6	75,0731	38,4900	87,1508	14,8148	100,0000
7	76,5107	0,0000	82,6816	0,0000	97,3684
8	78,6660	0,0000	83,2402	0,0000	98,6755
9	84,6500	32,7327	86,0335	10,7143	100,0000
10	33,6802	23,9244	68,7151	7,1429	80,1325
Median	81,3774	35,3564	85,4749	12,7646	97,3684
Rata-rata	75,8753	37,5715	84,1899	22,1561	95,3808

Hasil pada Tabel 4.17 menunjukkan terdapat nilai 0% pada sensitivitas *fold* 7 dan 8 yang mengartikan terdapat kesalahan sepenuhnya pada klasifikasi kelas minor. *Confusion matrix fold* 7 adalah sebagai berikut.

Tabel 4. 18 *Confusion Matrix Fold* 7 Metode Regresi Logistik Biner Pada Data *Testing Imbalance*

Actual	*Predicted*	
	Deal	*Lost/Cancel*
Deal	149	3
Lost/Cancel	27	0

Confusion matrix pada Tabel 4.18 memberikan informasi terdapat 27 observasi yang secara aktual merupakan *leads* dengan pembatalan transaksi namun diprediksi sebaliknya. Tidak terdapat observasi yang berhasil diprediksi untuk kasus *lost/cancel*. Seperti pada metode *Random Forest*, dilakukan *oversampling inside fold* dan *oversampling outside*

fold pada data *imbalance*. Sehingga, didapatkan hasil klasifikasi pada data *training* dan *testing* sebagai berikut.

Tabel 4. 19 Ketepatan Klasifikasi Regresi Logistik Biner Data Training *Oversampling Inside Fold* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	95,9919	89,8200	89,8535	92,3077	87,3993
2	94,5140	89,0557	89,1209	92,5275	85,7143
3	94,1729	88,6329	88,7179	92,6007	84,8352
4	93,8405	87,0438	87,1062	90,4029	83,8095
5	94,5548	87,7190	87,7656	90,6227	84,9084
6	94,6864	88,8602	88,9744	93,4799	84,4689
7	94,6427	88,3931	88,4615	91,9414	84,9817
8	94,6511	88,0435	88,0674	90,1171	86,0176
9	94,1784	88,6413	88,7262	92,6061	84,8463
10	95,4278	90,1852	90,2269	92,9722	87,4817
Median	94,5987	88,6371	88,7221	92,4176	84,9451
Rata-rata	94,6660	88,6395	88,7020	91,9578	85,4463

Pada Tabel 4.19 merupakan hasil ketepatan klasifikasi data training OIF dengan regresi logistik biner dan dapat dikatakan model yang terbentuk sudah dapat memberikan ketepatan klasifikasi yang baik pada data training. Apabila dibandingkan pada hasil data training RF-OIF pada Tabel 4.12, secara keseluruhan hasil dengan regresi logistik biner memiliki nilai yang lebih rendah dibandingkan dengan *Random Forest*.

Tabel 4. 20 Ketepatan Klasifikasi Regresi Logistik Biner Data Testing *Oversampling Inside Fold* (%)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	50,5726	47,2953	62,0112	33,3333	67,1053
2	88,9620	79,4719	73,7430	88,8889	71,0526
3	90,6676	84,0611	78,7710	92,5926	76,3158
4	99,1715	86,0663	96,0894	74,0741	100,0000

Tabel 4. 20 Ketepatan Klasifikasi RF Data Testing *Oversampling Inside Fold* (%) (Lanjutan)

Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
5	87,9386	73,3825	82,1229	62,9630	85,5263
6	80,2632	58,4064	83,7989	37,0370	92,1053
7	81,2865	56,3899	86,0335	33,3333	95,3947
8	80,0615	47,6257	80,4469	25,0000	90,7285
9	76,3245	59,3643	88,8268	35,7143	98,6755
10	35,6433	39,9562	46,9274	32,1429	49,6689
Median	80,7749	58,8854	81,2849	36,3757	88,1274
Rata-rata	77,0891	63,2020	77,8771	51,5079	82,6573

Setelah dilakukan *oversampling* di dalam *fold*, terdapat peningkatan rata-rata g-mean sebesar 25,6305%, peningkatan rata-rata sensitivitas sebesar 29,3519%. Hasil ketepatan klasifikasi dengan *oversampling outside fold* tercantum pada Tabel 4.21 dan 4.22.

Tabel 4. 21 Ketepatan Klasifikasi Regresi Logistik Biner Data Training *Oversampling Outside Fold* (%)

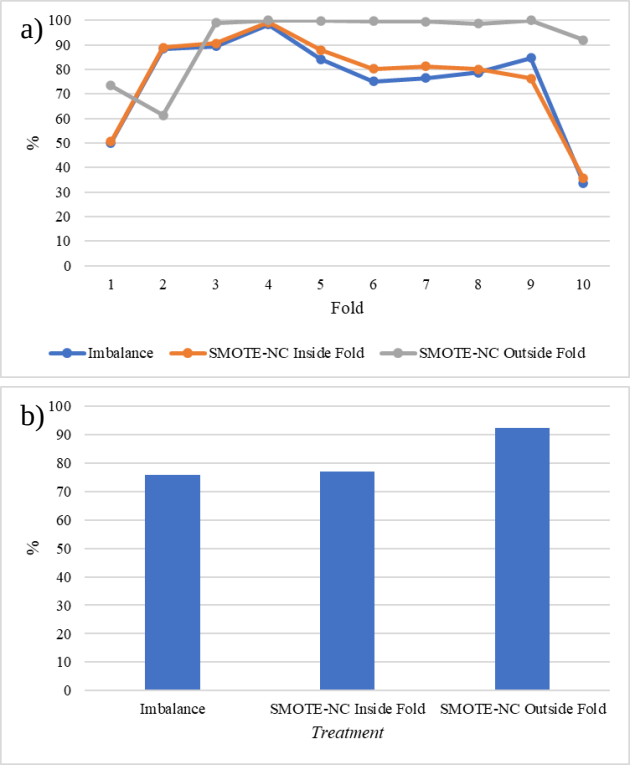
Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	97,2079	91,5582	91,5751	93,3333	89,8169
2	98,0496	92,9225	92,9304	94,1392	91,7216
3	95,6923	89,1920	89,1941	89,8169	88,5714
4	95,3409	88,3103	88,3150	89,2308	87,3993
5	95,5162	88,7898	88,7953	89,7511	87,8388
6	95,4947	88,5339	88,5390	89,4583	87,6190
7	95,3384	88,8954	88,9052	90,1903	87,6190
8	95,7070	88,6811	88,6855	89,5971	87,7745
9	95,2421	88,2061	88,2094	89,0110	87,4085
10	96,6160	90,0768	90,0769	89,9634	90,1903
Median	95,6043	88,8426	88,8502	89,7840	87,8067
Rata-rata	96,0205	89,5166	89,5226	90,4491	88,5959

Ketepatan klasifikasi data training RL-OOF pada Tabel 4.21 memberikan hasil yang sedikit lebih baik dibandingkan dengan RL-OIF pada Tabel 4.19 dengan kenaikan sebesar 1-3% pada nilai rata-rata pengukuran ketepatan klasifikasi. Apabila dibandingkan dengan RF-OOF pada Tabel 4.14 memberikan selisih sebesar 4 hingga 11% dengan nilai RL-OOF yang lebih kecil.

Tabel 4. 22 Ketepatan Klasifikasi Regresi Logistik Biner Data Testing *Oversampling Outside Fold* (%)

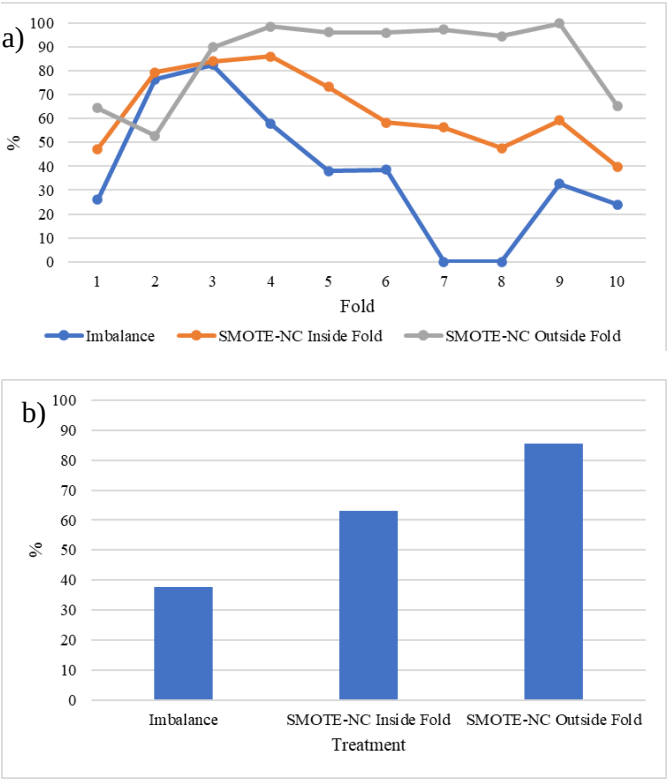
Fold	AUC	G-Mean	Akurasi	Sensitivitas	Spesifisitas
1	73,4418	64,3796	65,1316	55,2632	75,0000
2	61,3184	52,8368	58,8816	32,8947	84,8684
3	98,8963	90,0235	90,4605	99,3421	81,5789
4	99,9654	98,6842	98,6842	98,6842	98,6842
5	99,8649	96,3136	96,3696	100,0000	92,7632
6	99,6340	96,0259	96,0396	98,0132	94,0789
7	99,4597	97,3597	97,3597	98,0132	96,7105
8	98,6537	94,5537	94,7195	100,0000	89,4040
9	100,0000	100,0000	100,0000	100,0000	100,0000
10	91,9876	65,4568	70,9571	98,0263	43,7086
Median	99,1780	95,2898	95,3795	98,3553	91,0836
Rata-rata	92,3222	85,5634	86,8603	88,0237	85,6797

Agar perbandingan dapat dilihat lebih jelas, grafik nilai AUC dan g-mean pada masing-masing *fold* terdapat pada Gambar 4.8 dan Gambar 4.9.



Gambar 4. 8 a) Nilai AUC Berdasarkan *Fold*; b) Rata-rata Nilai AUC Data Testing Metode Regresi Logistik Biner

Nilai AUC dari hasil *oversampling* di dalam *fold* dan data *imbalance* terlihat tidak jauh berbeda dan terdapat *overlap* pada beberapa *fold*. *Oversampling* di luar *fold* memberikan nilai yang optimis dengan adanya nilai mendekati 100% pada beberapa *fold.* Kenaikan nilai rata-rata AUC dengan dilakukan *oversampling inside fold* sebesar 1,21% dan kenaikan nilai rata-rata setelah dilakukan *oversampling outside fold* sebesar 14,47%.



Gambar 4. 9 a) Nilai G-Mean Berdasarkan Fold; b) Rata-rata Nilai G-Mean Data Testing Metode Regresi Logistik Biner

Pada Gambar 4.9 menunjukkan nilai *oversampling* di dalam *fold* berada di atas hasil data *imbalance* pada *fold* 1 hingga 10. Hal ini menunjukkan terjadi peningkatan ketepatan klasifikasi pada data kelas minor. Sementara itu, pada poin b) menggambarkan rata-rata nilai g-mean pada data *imbalance*, *oversampling inside fold*, dan *oversampling outside fold.* Peningkatan performa klasifikasi setelah dilakukan

oversampling inside fold sebesar 25,63% dan perningkatan setelah *oversampling outside fold* sebesar 47,99%

Pemodelan *leads* PT "X" melakukan transaksi menggunakan metode regresi logistik biner dengan kategori respon 0 menunjukkan *leads* melakukan transaksi hingga selesai dan kategori respon 1 menunjukkan *leads* melakukan pembatalan transaksi. Pemodelan akan dilakukan pada data *oversampling inside fold* dan *oversampling outside fold* sebagai bentuk perbandingan.

4.3.1 Estimasi Parameter Model *Leads* PT "X"

Interpretasi mengenai estimasi parameter dan signifikansi pada model *oversampling inside* dan *outside fold* akan dibahas lebih lanjut pada sub bab ini. Sebagian hasil estimasi parameter pada *oversampling inside fold* dengan metode *backward* disajikan dalam Tabel 4.23, selengkapnya pada Lampiran 6 dan 7, serta hasil *oversampling outside fold* pada Lampiran 8 dan 9.

Tabel 4. 23 Nilai Estimasi Parameter Oversampling Inside Fold

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Pr(>|z|)
(Intercept)	-2,1250	0,6848	0,0019
MRR	0,0001	0,0000	0,0017
Industry2	0,6810	0,6086	0,2631
Industry3	-0,2448	0,6859	0,7212
…	…	…	…
IntroduceMonth2	0,7635	0,3824	0,0459
IntroduceMonth3	0,8206	0,4070	0,0438
…	…	…	…
TeamLeader2	-4,5000	1,2470	0,0003
TeamLeader3	-2,2330	0,2782	0,0000
…	…	…	…
Source2	0,3252	0,4102	0,4280

Tabel 4. 20 Nilai Estimasi Parameter *Oversampling Inside Fold* (Lanjutan)

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Pr(>|z|)
Source3	0,4766	1,3630	0,7266
…	…	…	…

Berdasarkan Tabel 4.23 kolom $\hat{\beta}$, apabila nilai $\hat{\beta}$ positif maka variabel tersebut memiliki hubungan berbanding lurus dengan terjadinya pembatalan atau hilangnya klien di tengah proses transaksi. Sebaliknya, apabila nilai $\hat{\beta}$ negatif maka hubungannya berbanding terbalik dengan terjadinya pembatalan atau hilangnya klien di tengah proses transaksi. Variabel signifikan pada tingkat kesahalan 5% apabila nilai kuadrat Wald lebih besar dari 3,841. Sehingga, jika dilihat dari nilai $\hat{\beta}$ dan berdasarkan signifikansi nilai Wald, maka variabel dan kelompok kategori yang bernilai $\hat{\beta}$ positif dan signifikan pada tingkat 5% yaitu *MRR, Industry9, Industry10, IntroduceMonth2, dan IntroduceMonth3*. Sementara itu apabila dilihat dari variabel dan kelompok kategori dengan nilai $\hat{\beta}$ negative dan signifikan pada tingkat 5% yaitu *IntroduceMonth9, IntroduceMonth11, IntroduceMonth12, TeamLeader2, TeamLeader3, TeamLeader4, TeamLeader5, TeamLeader6, TeamLeader10, Source8,* dan *Source 11*.

Berdasarkan hasil estimasi parameter pada Tabel 4.20 maka model probabilitas *leads* melakukan pembatalan di tengah proses transaksi adalah,

$$\pi(x) = \frac{\exp(0{,}0001MRR + 1{,}5Industry_9 + 3{,}26Industry_{10} + \cdots - 2{,}9Source_{11})}{1 + \exp(0{,}0001MRR + 1{,}5Industry_9 + 3{,}26Industry_{10} + \cdots - 2{,}9Source_{11})}$$

Probabilitas *leads* melakukan pembatalan transaksi dapat dihitung dengan melakukan substitusi nilai variable prediktor ke dalam persamaan model probabilitas.

4.3.2 Pengujian Signifikansi Parameter

Pada sub-bab ini membahas mengenai signifikansi parameter secara serentak dan parsial. Setelah melakukan estimasi parameter, selanjutnya dilakukan uji serentak untuk mengetahui terdapat setidaknya satu variabel independent yang berpengaruh signifikan terhadap variabel respon. Berdasarkan hasil pengujian signifikansi parameter secara serentak pada *oversampling inside* dan *outside fold* sebagai berikut.

Tabel 4. 24 Pengujian Signifikansi Parameter Secara Serentak

Treatment	$\chi^2_{(45,0,05)}$	G^2
Oversampling inside fold	61,6560	2171,2000
Oversampling outside fold	61,6560	2491,3000

Berdasarkan Tabel 4.24, nilai $G^2 > \chi^2_{(45,0,05)}$ sehingga keputusan yang diambil adalah tolak H_0 dengan kesimpulan terdapat minimal satu variabel independent yang berpengaruh signifikan terhadap variabel respon. Oleh karena itu, dilakukan pengujian signifikansi parameter secara parsial untuk mengetahui variabel yang berpengaruh signifikan terhadap variabel respon. Hasil pengujian signifikansi parameter secara parsial dengan model *oversampling inside fold* terdapat pada Tabel 4.25.

Tabel 4. 25 Pengujian Signifikansi Parameter Secara Parsial Model *Oversampling Inside Fold*

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Wald	Pr(>|z|)
MRR	0,0001	0,0000	9,8982	0,0017
Industry2	0,6810	0,6086	1,2520	0,2631
Industry3	-0,2448	0,6859	0,1273	0,7212
…	…	…	…	…
IntroduceMonth2	0,7635	0,3824	3,9864	0,0459
IntroduceMonth3	0,8206	0,4070	4,0651	0,0438
…	…	…	…	…

Tabel 4. 25 Pengujian Signifikansi Parameter Secara Parsial Model *Oversampling Inside Fold* (Lanjutan)

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Wald	Pr(>|z|)
TeamLeader2	-4,5000	1,2470	13,0224	0,0003
TeamLeader3	-2,2330	0,2782	64,4263	0,0000
…	…	…	…	…
Source2	0,3252	0,4102	0,6285	0,4280
Source3	0,4766	1,3630	0,1222	0,7266
…	…	…	…	…

Uji signifikansi parameter secara parsial pada masing-masing variabel independen dilakukan dengan membandingkan nilai Uji Wald dan nilai tabel distribusi *chi-square* dengan nilai $\chi^2_{(1,0,05)}$ = 3,841. Nilai Uji Wald > 3,841 disimpulkan berpengaruh signifikan terhadap respon. Berdasarkan Tabel 4.22, variabel yang signifikan yaitu *MRR*, *Industry*, *IntroduceMonth*, *TeamLeader*, dan *Source*. Sementara itu, pengujian signifikansi parameter secara parsial pada model *oversampling outside fold* adalah sebagai berikut. Tabel pengujian signifikansi parameter secara partial pada model *oversampling inside fold* dapat dilihat pada Lampiran 10.

Tabel 4. 26 Pengujian Signifikansi Parameter Secara Parsial Model *Oversampling Outside Fold*

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Wald	Pr(>|z|)
SpecialProject1	-0,9305	0,1728	28,9965	0,0000
Industry2	-0,0545	0,6943	0,0062	0,9375
Industry3	-0,7331	0,7143	1,0533	0,3047
…	…	…	…	…
IntroduceMonth2	0,0476	0,5520	0,0074	0,9313
IntroduceMonth3	1,0820	0,4860	4,9566	0,0259

Tabel 4. 26 Pengujian Signifikansi Parameter Secara Parsial Model *Oversampling Outside Fold* (Lanjutan)

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Wald	Pr(>|z|)
…	…	…	…	…
TeamLeader2	-3,0930	1,2750	5,8849	0,0153
TeamLeader3	-1,0370	0,3111	11,1111	0,0009
…	…	…	…	…
Source2	0,8823	0,6698	1,7352	0,1877
Source3	2,1360	1,3900	2,3614	0,1246
…	…	…	…	…

Menggunakan cara yang sama seperti pengujian signifikansi parameter secara parsial pada model *oversampling inside fold*, didapatkan variabel yang berpengaruh signifikan pada model dengan *treatment oversampling outside fold* yaitu *SpecialProject, Industry, IntroduceMonth, TeamLeader*, dan *Source*. Hasil lengkap dapat dilihat pada Lampiran 11.

4.3.3 Interpretasi Model *Leads* PT "X"

Besarnya pengaruh masing-masing variabel independent terhadap resiko terjadinya pembatalan transaksi dapat diketahui dengan menghitung nilai *odds ratio* sebagai berikut.

Tabel 4. 27 Odds Ratio Model *Oversampling Inside Fold*

Variabel	$\hat{\beta}$	Odds
MRR	0,0001	1,0001
Industry9	1,5010	4,4862
Industry10	3,2620	26,1017
IntroduceMonth2	0,7635	2,1458
IntroduceMonth3	0,8206	2,2719
IntroduceMonth9	-2,0870	0,1241
IntroduceMonth11	-1,7700	0,1703

Tabel 4. 27 Odds Ratio Model *Oversampling Inside Fold* (Lanjutan)

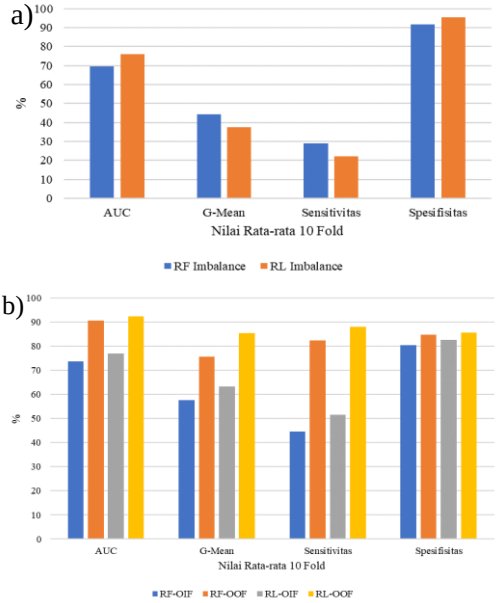
Variabel	$\hat{\beta}$	Odds
IntroduceMonth12	2,1930	8,9621
TeamLeader2	-4,5000	0,0111
TeamLeader3	-2,2330	0,1072
TeamLeader4	-3,5970	0,0274
TeamLeader5	-2,0050	0,1347
TeamLeader6	-1,0670	0,3440
TeamLeader10	-4,0850	0,0168
Source8	-5,0800	0,0062
Source11	-2,9060	0,0547

Pada Tabel 4.27 hanya ditampilkan nilai *odds ratio* pada variabel dan kategori yang memiliki nilai Wald lebih besar dari 3,841, hasil lengkap ditampilkan pada Lampiran 12. Hasil *odds ratio* dapat digunakan untuk menjelaskan besar pengaruh dari masing-masing variabel respon yang berpengaruh signifikan terhadap respon. Apabila MRR naik sebesar Rp1.000.000,00 maka memberikan resiko sebesar 1,0001 bagi *leads* untuk membatalkan transaksi. Nilai odds ratio 1,0001 dapat dibulatkan menjadi 1 sehingga didapatkan resiko yang sama antara melakukan pembatalan atau melanjutkan transaksi. Nilai odds ratio mendekati 1 didukung oleh nilai rata-rata MRR yang tidak berbeda jauh antara *leads deal* dan *lost* seperti tercantum pada Tabel 4.2. Pada variabel jenis industri, *Industry9* memiliki kecenderungan untuk membatalkan transaksi 4,5 kali lebih besar dibandingkan Industry1 dan Industry10 memiliki kecenderungan melakukan pembatalan transaksi 26 kali lebih besar dibandingkan *Industry1*. *Leads* yang diperkenalkan produk pada Februari dan Maret memiliki kecenderungan tidak melanjutkan transaksi 2 kali lebih besar dan memperkenalkan produk pada Desember memiliki kecenderungan *leads* tidak

melakukan transaksi sebesar 8 kali dibandingkan pada Januari. Sementara memperkenalkan produk pada September dan November dapat memperbesar kecenderungan *leads* untuk melakukan transaksi (*deal*) sebesar 6 hingga 8 kali lipat dibandingkan pada Januari. *Leads* yang didapatkan dari *Source8* memiliki kecenderungan untuk melakukan transaksi sebesar 160 kali lebih besar dibandingkan *Source1* dan dari *Source11* memiliki kecenderungan 18 kali lebih besar dibandingkan *Source1*. *Odds ratio* model *oversampling outside fold* dapat dilihat pada Lampiran 22.

4.4 Perbandingan Performa Klasifikasi Pada Data Imbalance, Oversampling Inside, dan Outside Fold

Pada sub-bab ini membahas mengenai perbandingan performa klasifikasi pada data *imbalance*, *oversampling inside*



Gambar 4. 10 Grafik Nilai Rata-rata AUC, G-Mean, Sensitivitas, dan Spesifisitas

fold, dan *oversampling outside fold* menggunakan metode *Random Forest* dan regresi logistik biner.

Pada Gambar 4.10 poin a) merupakan grafik nilai rata-rata pada 10-fold dengan menggunakan metode Random Forest dan regresi logistik biner pada data tidak seimbang. Apabila dilihat secara keseluruhan tidak terdapat selisih yang jauh pada hasil keduanya. Ditinjau dari nilai sensitivitas, metode *Random Forest* menghasilkan ketepatan klasifikasi kelas minor yang lebih baik dibandingkan regresi logistik biner dengan selisih sebesar 6,6667%. Sebaliknya, metode regresi logistik biner memberikan hasil ketepatan klasifikasi pada kelas mayor lebih tinggi dibandingkan *Random Forest* dengan selisih 3,7060%. Secara umum, klasifikasi data imbalance dengan RF dan LR (tanpa kombinasi OIF dan OIF) menghasilkan akurasi kelas minor (sensitivitas) jauh lebih rendah daripada akurasi kelas mayor (spesivisitas). Secara keseluruhan, apabila ditinjau dari nilai AUC dan g-mean didapatkan akurasi *Random Forest* dan regresi logistik biner relatif rendah pada klasifikasi data *imbalance*.

Perbandingan hasil setelah dilakukan *oversampling inside* dan *outside fold* terdapat pada Gambar 4.10 poin b). Berdasarkan grafik tersebut, secara umum kombinasi OIF dan OOF telah meningkatkan akurasi klasifikasi pada data *imbalance*. *Oversampling outside fold* menghasilkan performa ketepatan klasifikasi yang lebih baik dibandingkan *oversampling inside fold* pada kedua metode *Random Forest* dan regresi logistik biner. Kombinasi RL-OOF memberikan hasil yang lebih baik dibandingkan RF-OOF dengan peningkatan rata-rata AUC sebesar 16,4469%, g-mean sebesar 47,9919%, dan peningkatan rata-rata sensitivitas sebesar 65,8676%.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan analisis dan pembahasan yang telah dilakukan, diperoleh kesimpulan bahwa *Source* paling banyak dalam melakukan pembatalan transaksi yaitu *Source9, Source2,* dan *Source7*. MRR pada kategori *deal* lebih menyebar dibandingkan kategori *lost* dan terdapat peningkatan jumlah *leads deal* maupun *lost* pada Q4.

Hasil ketepatan klasifikasi dengan metode RF-OIF menghasilkan rata-rata AUC sebesar 73,77%, rata-rata g-mean sebesar 57,59%, dan rata-rata sensitivity sebesar 44,66% dengan kenaikan nilai rata-rata dari data *imbalance* sebesar 4,23%, 13,33%, dan 15,84%. Sementara itu, RF-OOF menghasilkan nilai rata-rata AUC sebesar 90,63%, rata-rata g-mean sebesar 75,59%, dan rata-rata sensitivitas sebesar 82,50%. Sehingga, apabila dibandingkan dengan hasil ketepatan klasifikasi pada data *imbalance* didapatkan kenaikan nilai AUC sebesar 21,09%, kenaikan g-mean sebesar 31,33%, dan kenaikan sensitivity sebesar 53,67%.

Didapatkan hasil rata-rata AUC dengan RL-OIF sebesar 77,08%, rata-rata g-mean 63,20%, dan rata-rata sensitivity 51,50% dengan kenaikan masing-masing sebesar 1,21%, 25,63%, dan 29,35% terhadap data *imbalance*. Sementara itu, dengan RL-OOF menghasilkan rata-rata AUC sebesar 92,32%, rata-rata g-mean 85,56%, dan rata-rata sensitivity 88,02%. Kenaikan yang dihasilkan terhadap data *imbalance* masing-masing sebesar 16,47%, 47,99%, dan 65,86%. Variabel signifikan dari hasil RL-OIF adalah MRR, *Industry*, *IntroduceMonth*, *TeamLeader*, dan *Source*. Sementara itu, variabel signifikan hasil RL-OOF adalah *SpecialProject*, *Industry*, *IntroduceMonth*, *TeamLeader*, dan *Source* dengan

odds ratio terbesar pada kategori *Industry10* yang memiliki resiko 11 kali lebih besar untuk membatalkan transaksi dibandingkan *Industry1*.

Perbandingan antara *oversampling inside* dan *oversampling outside fold* menghasilkan kesimpulan melakukan *oversampling outside fold* memberikan hasil yang lebih baik dengan kenaikan akurasi kelas minor hingga 65%.

5.2 Saran

Berdasarkan kesimpulan yang telah diperoleh, jenis industri memengaruhi penggunaan produk PT "X" yang berbasis SaaS, dengan ini dapat dilakukan analisis lebih lanjut mengenai perusahaan *leads* yang merupakan rintisan (*start-up*) atau perusahaan *leads* yang sudah melalui masa rintisan (korporasi). Apabila sudah didapatkan analisis lebih dalam mengenai perusahaan *leads* maka dapat dilakukan pendekatan yang lebih sesuai dari sisi produk maupun *marketing*. Selain itu, faktor pelaksanaan pengenalan produk menjadi salah satu pengaruh keputusan *leads* untuk *deal* atau *lost* maka untuk mempertahankan *revenue* PT "X" perlu dilakukan pendekatan khusus pada Q1 dan Q2. Saran terkait analisis data menggunakan *oversampling* adalah dengan mencoba metode tambahan selain SMOTE sebagai perbandingan hasil analisis dan upaya mendapatkan hasil akurasi yang lebih baik.

DAFTAR PUSTAKA

Agresti, A. (2012). *Categorical Data Analysis* (3rd ed.). Wiley.

Akosa, J. S. (2017). Predictive Accuracy: A Misleading Performance Measure for Highly Imbalanced Data. *SaS Global Forum 2017*, 942.

Berthold, M. R., & Hand, D. (2003). *Intelligent Data Analysis* (2nd ed.). Berlin: Springer-Verlag Berlin Heidelberg. doi:10.1007/978-3-540-48625-1

Blagus, R., & Lusa, L. (2015). Joint Use of Over- and Undersampling Techniques and Cross-Validation for The Development and Assessment of Prediction Models. *BMC Bioinformatics, 16*(362). doi:10.1186/s12859-015-0784-9

Blanchard, D. (2017, July 17). *Digitalization Boosts Demand for Supply Chain Software*. Retrieved January 2020, from Material Handling & Logistic: https://www.mhlnews.com/technology-automation/article/22054451/digitalization-boosts-demand-for-supply-chain-software

Breiman, L. (2001, October). Random Forest. (R. E. Schapire, Ed.) *Machine Learning*(45), 5-32. doi:10.1023/A:1010933404324

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1993). *Classification and Regression Trees.* New York: Chapman Hall.

Chawla, N., Bowyer, K., Hall, L., & Kegelmeyer, W. P. (2002, June). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research, 16*, 321-357. Retrieved 2020

Christopher, E. (2019, May 8). *There Are Only Two Types of SaaS Applications In Your Business Today*. (Forbes

Technology Council) Retrieved January 2020, from Forbes: https://www.forbes.com/sites/forbestechcouncil/2019/05/08/there-are-only-two-types-of-saas-applications-in-your-business-today/#162bc4141c86

Dormann, C., Elith, J., Bacher, S., Buchmann, C., Carl, G., Carré, G., . . . Lautenbach, S. (2012, May 18). Collinearity: A Review of Methods To Deal With It and A Simulation Study Evaluating Their Performance. *Ecography, 36*(1).

Fawcett, T. (2006, June). An Introduction to ROC Analysis. *Pattern Recognition Letters*, 861-874. doi:10.1016/j.patrec.2005.10.010

Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression.* New York: John Wiley & Sons, Inc.

Kursa, M. B. (2020, May 21). *Boruta for Those In A Hurry*. Retrieved from The Comprehensive R Archive Network: https://cran.r-project.org/web/packages/Boruta/vignettes/inahurry.pdf

Kursa, M. B., Jankowski, A., & Rudnicki, W. R. (2010). Boruta – A System for Feature Selection. *Fundamenta Informaticae 101*, 271-285. doi:10.3233/FI-2010-288

Kutcher, E., Nottebohm, O., & Sprague, K. (2014, April). Software and online-services companies can quickly become billion-dollar giants, but the recipe for sustained growth remains elusive. *Grow Fast or Die Slow*. Retrieved January 2020, from McKinsey & Company: https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/grow-fast-or-die-slow#

Lamb, C. W., Hair, J. F., McDaniel, C., Kapoor, H., Appleby, R., & Shearer, J. (2002). *MKTG: Principles of Marketing* (Second Canadian ed.). Toronto, Ontario, Canada: Nelson College Indigenous.

Mao, W., & Wang, F.-Y. (2013). *New Advances in Intelligence and Security Informatics*. Elsevier Inc. doi:10.1016/C2011-0-07828-6

Narkhede, S. (2018, June 26). *Understanding AUC - ROC Curve*. Retrieved January 2020, from Towards Data Science: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

Red Hat. (n.d.). *Red Hat: Cloud Computing*. Retrieved from What is SaaS?: https://www.redhat.com/en/topics/cloud-computing/what-is-saas

Sabbeh, S. F. (2018). Machine-Learning Techniques for Customer Retention: A Comparative Study. *International Journal of Advanced Computer Science and Applications, 9*(2). Retrieved January 2020

Smyrnova, T. (2019, September 13). *The Future of SaaS Applications*. Retrieved January 2020, from Syndicode: https://syndicode.com/2019/09/13/the-future-of-saas-applications/

Twin, A. (2019, June 25). *Marketing*. Retrieved January 2020, from Investopedia: https://www.investopedia.com/terms/m/marketing.asp

Williams, G. (2011). *Data Mining with Rattle and R: The Art of Excavating Data for Knowledge Discovery (Use R!)*. New York: Springer.

Wray, N. R., Yang, J., Goddard, M. E., & Visscher, P. M. (2010, February 26). The Genetic Interpretation of Area under the ROC Curve in Genomic Profiling. (N.

J. Schork, Ed.) *PLOS Genetics*, *6*(2). doi:10.1371/journal.pgen.1000864

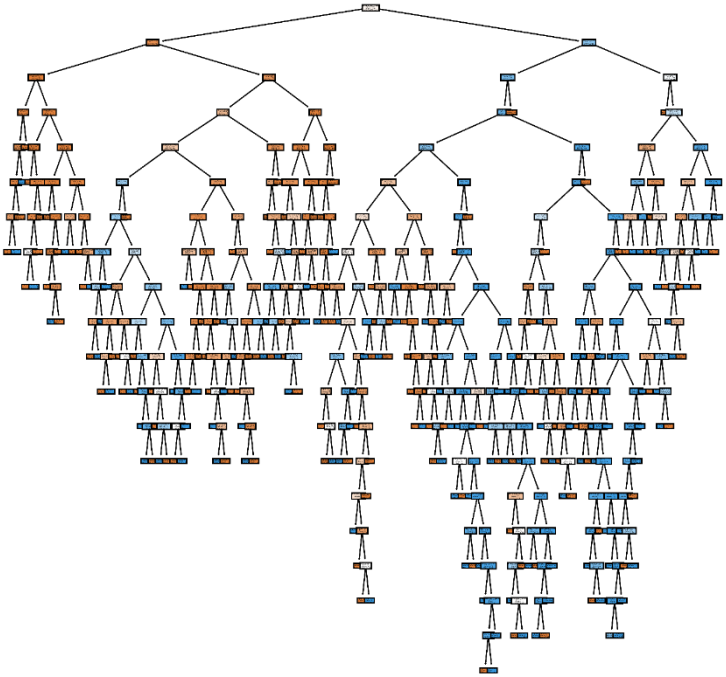
Wulandari, S. P., Salamah, M., & Susilaningrum, D. (2009). *Analisis Data Kualitatif*. Surabaya.

LAMPIRAN

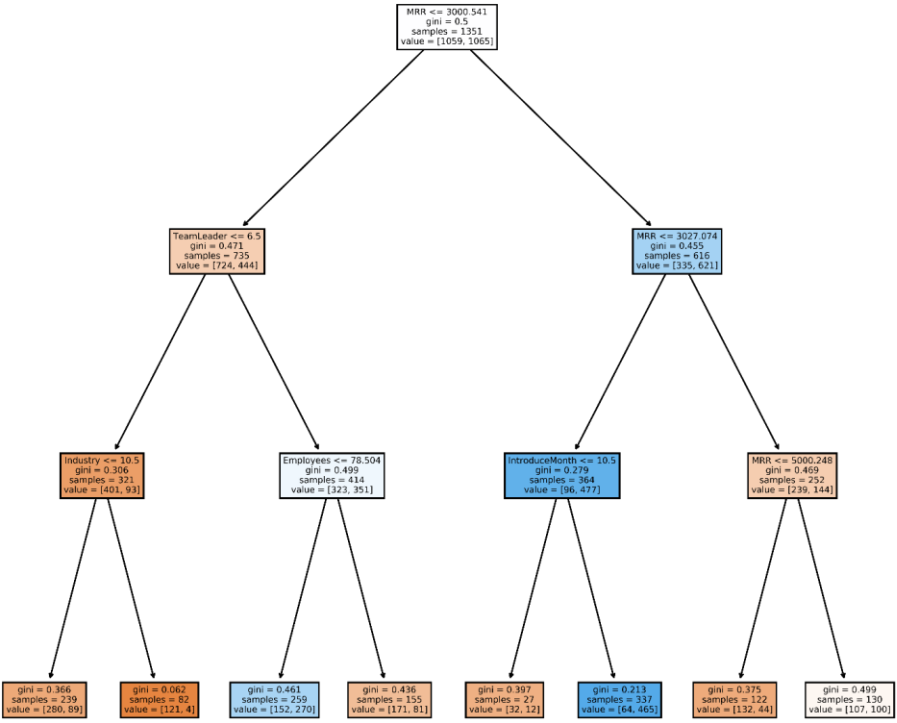
Lampiran 1. Data Leads Deal dan Lost/ Cancelled Deal

No	y	Employees	MRR (000)	SpecialProject	Industry	IntroduceMonth	TeamLeader	ContractLength	Source
1	0	250	7500	0	10	11	9	12	7
2	0	30	1000	1	10	12	6	12	2
3	0	90	2000	0	9	11	6	12	9
4	0	70	2000	0	3	10	6	6	9
5	0	165	750	0	10	10	3	12	9
…									
462	0	300	3000	0	4	9	6	12	11
463	0	100	1700	1	13	7	9	12	7
464	0	40	360	1	11	8	9	12	9
465	0	17	300	1	13	7	5	12	7
…									
1788	1	169	3816	0	10	7	9	12	9
1789	1	250	7500	0	10	11	9	12	7
1790	1	100	2000	0	10	12	3	12	2

Lampiran 2. Random Forest Plot Average 18 Nodes



Lampiran 3. Pengambilan Pohon 3 Nodes



Lampiran 4. Syntax Random Forest dan Regresi Logistik Biner [Python]

```
from sklearn.ensemble import RandomForestClassifier
from sklearn import metrics
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold, StratifiedKFold
from sklearn.model_selection import train_test_split
from sklearn.model_selection import ShuffleSplit
from sklearn.metrics import (accuracy_score,confusion_matrix,
classification_report, roc_auc_score,
f1_score, recall_score, precision_score, SCORERS, auc,
roc_curve)
import numpy as np
import seaborn as sns
import pandas as pd
from scipy import interp
from sklearn.datasets import make_classification
from imblearn.over_sampling import SMOTENC
from sklearn.datasets import make_classification
import matplotlib.pyplot as plt
import matplotlib
%matplotlib inline
matplotlib.rcParams.update({'font.size': 20})
from collections import Counter
from sklearn.tree import export_graphviz
from sklearn import tree
import pydot
import statsmodels.api as sm
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
np.random.seed(10)
import warnings
warnings.filterwarnings("ignore")
import time

df['y'] = df['y'].astype('category')
df['Source'] = df['Source'].astype('category')
df['Industry'] = df['Industry'].astype('category')
```

Lampiran 4. Syntax Random Forest dan Regresi Logistik Biner [Python] (Lanjutan)

```
df['IntroduceMonth'] = df['IntroduceMonth'].astype('category')
df['TeamLeader'] = df['TeamLeader'].astype('category')
df['ContractLength'] = df['ContractLength'].astype('category')
df['SpecialProject'] = df['SpecialProject'].astype('category')
df['Employees'] = df['Employees'].astype('float')
df.info()

y = df.y
X = df.drop('y', axis=1)


RF=RandomForestClassifier(random_state=123)
n=10
kf=StratifiedKFold(n_splits=n, random_state=123)

#IMBALANCE 10 FOLDS
for train_index, test_index in kf.split(Xv, yv):
    X_train, X_test = Xv[train_index], Xv[test_index]
    y_train, y_test = yv[train_index], yv[test_index]
    RF.fit(X_train, y_train)
    y_pred = RF.predict(X_test)
    cm.append((confusion_matrix(y_test, y_pred)).astype(float))

#OVERSAMPLING INSIDE 10 FOLDS
for train_index, test_index in kf.split(Xv, yv):
    X_train, X_test = Xv[train_index], Xv[test_index]
    y_train, y_test = yv[train_index], yv[test_index]
    smote = SMOTENC(categorical_features=[2,3,4,5,6,7],
random_state=123)
    X_train, y_train = smote.fit_resample(X_train, y_train)
    RF.fit(X_train, y_train)
    y_pred = RF.predict(X_train)
    cm.append((confusion_matrix(y_train, y_pred)).astype(float))
```

Lampiran 4. Syntax Random Forest dan Regresi Logistik Biner [Python] (Lanjutan)

```
#OVERSAMPLING OUTSIDE 10 FOLDS
sm = SMOTENC(categorical_features=[2,3,4,5,6,7],
random_state=123)
X_train, y_train = sm.fit_resample(X_train, y_train)

for train_index, test_index in kf.split(Xv_smote, yv_smote):
    X_train, X_test = Xv_smote[train_index],
Xv_smote[test_index]
    y_train, y_test = yv_smote[train_index], yv_smote[test_index]
    RF.fit(Xv_smote[train_index], yv_smote[train_index])
    y_pred = RF.predict(Xv_smote[train_index])
    cm.append((confusion_matrix(yv_smote[train_index],
y_pred)).astype(float))

#ROC-AUC
for train, test in kf.split(Xv, yv):
    X_train, X_test = Xv[train], Xv[test]
    y_train, y_test = yv[train], yv[test]

    smote = SMOTENC(categorical_features=[2,3,4,5,6,7],
random_state=123)
    X_train, y_train = smote.fit_resample(X_train, y_train)

    probas_ = RF.fit(X_train, y_train).predict_proba(X_train)
    # Compute accuracy
    y_pred = RF.predict(X_train)
    acc[i] = (y_pred == y_train).mean()
    # Confusion matrix
    confm = confm + confusion_matrix(y_train, y_pred)
    # Compute ROC curve and area the curve
    fpr, tpr, thresholds = roc_curve(y_train, probas_[:, 1])
    tprs.append(interp(mean_fpr, fpr, tpr))
    tprs[-1][0] = 0.0
    roc_auc = auc(fpr, tpr)
    aucs.append(roc_auc)
    plt.plot(fpr, tpr, lw=1, alpha=0.3,
             label='ROC fold %d (AUC = %0.4f)' % (i, roc_auc))
```

Lampiran 4. Syntax Random Forest dan Regresi Logistik Biner [Python] (Lanjutan)

```
#RF PLOT FULL
fig, axes = plt.subplots(nrows = 1,ncols = 1,figsize = (10,10), dpi=800)
tree.plot_tree(RFsmote.estimators_[0],
               feature_names = feature_list,
               filled = True);

#RF PLOT LIMITED TO THREE NODES
RFsmote_small = RandomForestClassifier(n_estimators=10, max_depth = 3)
RFsmote_small.fit(X_smote, y_smote)
tree_small = RFsmote_small.estimators_[5]

fig, axes = plt.subplots(nrows = 1,ncols = 1,figsize = (10,10), dpi=800)
tree.plot_tree(tree_small,
               feature_names = feature_list,
               filled = True);
#DUMMY
cat_vars = ['Industry','IntroduceMonth','TeamLeader','ContractLength','Source']
for var in cat_vars:
    cat_list = 'var'+'_'+var
    cat_list = pd.get_dummies(df[var], prefix=var)
    df1 = df.join(cat_list)
    df = df1
cat_vars = ['Industry','IntroduceMonth','TeamLeader','ContractLength','Source']
data_vars = df.columns.values.tolist()
to_keep = [i for i in data_vars if i not in cat_vars]
data_final = df[to_keep]
data_final.columns.values
```

Lampiran 4. Syntax Random Forest dan Regresi Logistik Biner [Python] (Lanjutan)

```
#Proses Pengklasifikasian dengan Regresi Logistik Biner
RF=LogisticRegression(solver='newton-cg', random_state=123)
n=10
kf=StratifiedKFold(n_splits=n, random_state=123)
#Klasifikasi menggunakan syntax pada #IMBALANCE 10
FOLDS, #OVERSAMPLING INSIDE 10 FOLDS,
#OVERSAMPLING OUTSIDE 10 FOLDS, dan #ROC-AUC
```

Lampiran 5. Syntax Model Regresi Logistik Biner [R]

```
library(nlme)
library(lmtest)
library(StepReg)

names <- c(1,4:63)
data[,names] <- lapply(data[,names], factor)
data$Employees <- as.integer(data$Employees)
sapply(data, class)

fit <- glm(y~.,data=data,family=binomial())
summary(fit)
fitback <- stepAIC(fit, direction = "backward", trace = FALSE)
summary(fitback)
```

Lampiran 6. Output Nilai Estimasi Parameter Regresi Logistik Biner *Oversampling Inside Fold* Awal

Variabel	Estimate	Std. Error	Pr(>|z|)
(Intercept)	-15.9500	3956.0000	0.9968
Employees	-0.0002	0.0003	0.5378
MRR	0.0001	0.0000	0.0190
SpecialProject1	-0.1955	0.1596	0.2205
Industry2	0.6740	0.6099	0.2691
Industry3	-0.2487	0.6849	0.7165
Industry4	-0.6690	0.5701	0.2406
Industry5	-15.2300	537.6000	0.9774
Industry6	0.1517	0.7652	0.8428
Industry7	0.4191	0.6074	0.4902
Industry8	-0.0358	0.5603	0.9491
Industry9	1.4800	0.4743	0.0018
Industry10	3.2530	0.4436	0.0000
Industry11	-1.3000	0.6951	0.0615
Industry12	-0.0257	1.1380	0.9820
Industry13	-0.1047	0.5353	0.8449
Industry14	0.3914	0.5098	0.4427
IntroduceMonth2	0.7883	0.3855	0.0408
IntroduceMonth3	0.8230	0.4108	0.0451
IntroduceMonth4	0.0699	0.4023	0.8621
IntroduceMonth5	0.2109	0.3841	0.5829
IntroduceMonth6	-0.2092	0.4559	0.6464
IntroduceMonth7	-0.4742	0.3879	0.2216
IntroduceMonth8	0.0666	0.3696	0.8571
IntroduceMonth9	-2.1090	0.4796	0.0000
IntroduceMonth10	-0.2685	0.3359	0.4242

Lampiran 6. Output Nilai Estimasi Parameter Regresi Logistik Biner *Oversampling Inside Fold* Awal (Lanjutan)

Variabel	Estimate	Std. Error	Pr(>|z|)
IntroduceMonth11	-1.7580	0.3949	0.0000
IntroduceMonth12	2.1490	0.2958	0.0000
TeamLeader2	-4.4350	1.2490	0.0004
TeamLeader3	-2.1470	0.2859	0.0000
TeamLeader4	-3.4480	0.7833	0.0000
TeamLeader5	-1.9530	0.3054	0.0000
TeamLeader6	-0.9996	0.2500	0.0001
TeamLeader7	0.1167	0.3642	0.7487
TeamLeader8	-16.5500	3956.0000	0.9967
TeamLeader9	-0.3339	0.2308	0.1480
TeamLeader10	-4.9020	1.7840	0.0060
ContractLength3	-2.8440	4337.0000	0.9995
ContractLength5	-1.2480	4824.0000	0.9998
ContractLength6	13.8000	3956.0000	0.9972
ContractLength7	1.1600	5595.0000	0.9998
ContractLength8	-1.7330	5595.0000	0.9998
ContractLength11	3.2940	5595.0000	0.9995
ContractLength12	13.9000	3956.0000	0.9972
ContractLength13	0.6355	4292.0000	0.9999
ContractLength14	-1.6970	4094.0000	0.9997
ContractLength18	1.0250	5595.0000	0.9999
ContractLength24	12.0600	3956.0000	0.9976
ContractLength26	1.1750	5595.0000	0.9998
ContractLength36	1.2150	5595.0000	0.9998
ContractLength37	-2.1780	4355.0000	0.9996
ContractLength40	-0.3831	5595.0000	0.9999

Lampiran 6. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Inside Fold Awal (Lanjutan)

Variabel	Estimate	Std. Error	Pr(>|z|)
Source2	0.3128	0.4128	0.4486
Source3	0.5911	1.4510	0.6837
Source4	2.8320	5.1450	0.5821
Source5	-1.1080	0.8618	0.1985
Source6	-15.2300	3956.0000	0.9969
Source7	-0.2103	0.4595	0.6472
Source8	-3.3930	1.9320	0.0791
Source9	0.0728	0.4106	0.8593
Source10	-18.3200	685.4000	0.9787
Source11	-2.8930	0.6445	0.0000
Source12	-0.4784	1.0520	0.6494

Lampiran 7. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Inside Fold Backward

Variabel	βˆ	SE(B)	Pr(>|z|)
(Intercept)	-2.1250	0.6848	0.0019
MRR	0.0001	0.0000	0.0017
Industry2	0.6810	0.6086	0.2631
Industry3	-0.2448	0.6859	0.7212
Industry4	-0.6747	0.5690	0.2357
Industry5	-15.2200	536.6000	0.9774
Industry6	0.1366	0.7646	0.8582
Industry7	0.4325	0.6079	0.4768
Industry8	-0.0371	0.5589	0.9472
Industry9	1.5010	0.4720	0.0015
Industry10	3.2620	0.4431	0.0000
Industry11	-1.3540	0.6924	0.0505
Industry12	-0.0227	1.1370	0.9841
Industry13	-0.0954	0.5345	0.8583
Industry14	0.4126	0.5085	0.4171
IntroduceMonth2	0.7635	0.3824	0.0459
IntroduceMonth3	0.8206	0.4070	0.0438
IntroduceMonth4	0.0352	0.3959	0.9293
IntroduceMonth5	0.2185	0.3811	0.5664
IntroduceMonth6	-0.3110	0.4451	0.4847
IntroduceMonth7	-0.4516	0.3838	0.2394
IntroduceMonth8	-0.0110	0.3601	0.9757
IntroduceMonth9	-2.0870	0.4736	0.0000
IntroduceMonth10	-0.2672	0.3327	0.4219
IntroduceMonth11	-1.7700	0.3912	0.0000
IntroduceMonth12	2.1930	0.2923	0.0000

Lampiran 8. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Inside Fold Backward (Lanjutan)

Variabel	β^	SE(B)	Pr(>|z|)
TeamLeader2	-4.5000	1.2470	0.0003
TeamLeader3	-2.2330	0.2782	0.0000
TeamLeader4	-3.5970	0.7746	0.0000
TeamLeader5	-2.0050	0.3030	0.0000
TeamLeader6	-1.0670	0.2449	0.0000
TeamLeader7	0.0298	0.3543	0.9329
TeamLeader8	-16.8100	3956.0000	0.9966
TeamLeader9	-0.4167	0.2233	0.0620
TeamLeader10	-4.0850	1.2120	0.0007
Source2	0.3252	0.4102	0.4280
Source3	0.4766	1.3630	0.7266
Source4	2.8280	5.1110	0.5800
Source5	-1.1570	0.8351	0.1658
Source6	-15.3100	3956.0000	0.9969
Source7	-0.2082	0.4587	0.6499
Source8	-5.0800	1.2880	0.0001
Source9	0.0699	0.4081	0.8641
Source10	-18.5300	701.4000	0.9789
Source11	-2.9060	0.6465	0.0000
Source12	-0.4627	1.0510	0.6598

Lampiran 8. Output Nilai Estimasi Parameter Regresi Logistik Biner *Oversampling Outside Fold* Awal

Variabel	$\hat{\beta}$	SE(B)	Pr(>|z|)
(Intercept)	-18.2500	3956.0000	0.9963
Employees	-0.0004	0.0004	0.3135
MRR	0.0000	0.0000	0.2444
SpecialProject1	-0.9077	0.1810	0.0000
Industry2	-0.0876	0.6985	0.9002
Industry3	-0.7813	0.7162	0.2753
Industry4	-0.8734	0.5635	0.1212
Industry5	-15.4500	520.7000	0.9763
Industry6	0.0418	0.7455	0.9553
Industry7	-0.3557	0.6841	0.6031
Industry8	-0.2204	0.5679	0.6980
Industry9	0.3076	0.4810	0.5224
Industry10	2.4010	0.4110	0.0000
Industry11	-1.0200	0.6127	0.0960
Industry12	-0.3907	1.1410	0.7320
Industry13	-0.8325	0.5597	0.1369
Industry14	-0.8753	0.5671	0.1227
IntroduceMonth2	0.0268	0.5553	0.9615
IntroduceMonth3	1.0700	0.4901	0.0291
IntroduceMonth4	-0.0801	0.5045	0.8739
IntroduceMonth5	0.2700	0.4687	0.5645
IntroduceMonth6	-0.4451	0.5943	0.4538
IntroduceMonth7	0.0167	0.4538	0.9706
IntroduceMonth8	-0.3102	0.4866	0.5239
IntroduceMonth9	-2.6900	0.7047	0.0001
IntroduceMonth10	-1.3570	0.4829	0.0050
IntroduceMonth11	-1.9620	0.5102	0.0001

Lampiran 8. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Outside Fold Awal (Lanjutan)

Variabel	$\hat{\beta}$	SE(B)	Pr(>|z|)
IntroduceMonth12	2.2690	0.3564	0.0000
TeamLeader2	-3.0930	1.2730	0.0151
TeamLeader3	-1.0430	0.3129	0.0009
TeamLeader4	-2.2400	0.7814	0.0042
TeamLeader5	-1.0720	0.3452	0.0019
TeamLeader6	-0.9802	0.2823	0.0005
TeamLeader7	0.3867	0.4289	0.3673
TeamLeader8	-14.2200	3956.0000	0.9971
TeamLeader9	0.5853	0.2329	0.0120
TeamLeader10	-4.8390	2.5840	0.0611
ContractLength3	-0.6828	4375.0000	0.9999
ContractLength5	0.1659	4827.0000	1.0000
ContractLength6	14.8600	3956.0000	0.9970
ContractLength7	2.3760	5595.0000	0.9997
ContractLength8	-0.8068	5595.0000	0.9999
ContractLength11	3.1050	5595.0000	0.9996
ContractLength12	15.2800	3956.0000	0.9969
ContractLength13	2.3150	4301.0000	0.9996
ContractLength14	-0.5589	4094.0000	0.9999
ContractLength18	2.2840	5595.0000	0.9997
ContractLength24	14.0900	3956.0000	0.9972
ContractLength26	2.3860	5595.0000	0.9997
ContractLength36	3.2340	5595.0000	0.9995
ContractLength37	-0.9672	4378.0000	0.9998
ContractLength40	0.5496	5595.0000	0.9999
Source2	0.8906	0.6744	0.1866
Source3	2.2370	1.4300	0.1178

Lampiran 8. Output Nilai Estimasi Parameter Regresi Logistik Biner *Oversampling Outside Fold* Awal (Lanjutan)

Variabel	$\hat{\beta}$	SE(B)	Pr(>|z|)
Source4	3.5200	5.8730	0.5490
Source5	-0.0933	1.2510	0.9406
Source6	-12.1000	3956.0000	0.9976
Source7	0.8603	0.7273	0.2369
Source8	-1.5580	2.6980	0.5637
Source9	1.1750	0.6648	0.0772
Source10	-17.2900	683.0000	0.9798
Source11	-1.0690	0.8081	0.1857
Source12	0.5521	1.2120	0.6488

Lampiran 9. Output Nilai Estimasi Parameter Regresi Logistik Biner Oversampling Outside Fold Backward

Variabel	$\hat{\beta}$	SE(B)	Pr(>|z|)
(Intercept)	-2.9290	0.8479	0.0006
SpecialProject1	-0.9305	0.1728	0.0000
Industry2	-0.0545	0.6943	0.9375
Industry3	-0.7331	0.7143	0.3047
Industry4	-0.8865	0.5599	0.1134
Industry5	-15.4200	520.2000	0.9764
Industry6	0.0366	0.7398	0.9605
Industry7	-0.3086	0.6828	0.6513
Industry8	-0.2201	0.5645	0.6967
Industry9	0.3017	0.4756	0.5259
Industry10	2.4250	0.4071	0.0000
Industry11	-1.0330	0.6060	0.0881
Industry12	-0.4232	1.1370	0.7097
Industry13	-0.8018	0.5555	0.1489
Industry14	-0.9102	0.5622	0.1054
IntroduceMonth2	0.0476	0.5520	0.9313
IntroduceMonth3	1.0820	0.4860	0.0259
IntroduceMonth4	-0.1319	0.4992	0.7916
IntroduceMonth5	0.2570	0.4655	0.5809
IntroduceMonth6	-0.5085	0.5838	0.3837
IntroduceMonth7	0.0055	0.4510	0.9902
IntroduceMonth8	-0.3477	0.4842	0.4727
IntroduceMonth9	-2.7330	0.7008	0.0001
IntroduceMonth10	-1.3880	0.4807	0.0039
IntroduceMonth11	-1.9820	0.5060	0.0001
IntroduceMonth12	2.2960	0.3538	0.0000
TeamLeader2	-3.0930	1.2750	0.0153

Lampiran 9. Output Nilai Estimasi Parameter Regresi Logistik Biner *Oversampling Outside Fold Backward* (Lanjutan)

Variabel		β̂	SE(B)	Pr(>|z|)
TeamLeader3		-1.0370	0.3111	0.0009
TeamLeader4		-2.2300	0.7865	0.0046
TeamLeader5		-1.1060	0.3434	0.0013
TeamLeader6	-0.9960		0.2810	0.0004
TeamLeader7	0.3948		0.4249	0.3528
TeamLeader8	-14.5600		3956.0000	0.9971
TeamLeader9	0.5521		0.2311	0.0169
TeamLeader10	-3.2050		1.3320	0.0161
Source2	0.8823		0.6698	0.1877
Source3	2.1360		1.3900	0.1246
Source4	3.5410		5.9610	0.5525
Source5	-0.3374		1.1850	0.7759
Source6	-11.9600		3956.0000	0.9976
Source7	0.8319		0.7223	0.2494
Source8	-3.9900		1.4820	0.0071
Source9	1.1600		0.6602	0.0790
Source10	-17.7200		697.9000	0.9797
Source11	-1.1060		0.8070	0.1703
Source12	0.6397		1.1940	0.5921

Lampiran 10. Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Inside Fold

Variabel	β^	SE(B)	Wald	Pr(>|z|)
MRR	0.0001	0.0000	9.8982	0.0017
Industry2	0.6810	0.6086	1.2521	0.2631
Industry3	-0.2448	0.6859	0.1274	0.7212
Industry4	-0.6747	0.5690	1.4060	0.2357
Industry5	-15.2200	536.6000	0.0008	0.9774
Industry6	0.1366	0.7646	0.0319	0.8582
Industry7	0.4325	0.6079	0.5062	0.4768
Industry8	-0.0371	0.5589	0.0044	0.9472
Industry9	1.5010	0.4720	10.1129	0.0015
Industry10	3.2620	0.4431	54.1956	0.0000
Industry11	-1.3540	0.6924	3.8240	0.0505
Industry12	-0.0227	1.1370	0.0004	0.9841
Industry13	-0.0954	0.5345	0.0319	0.8583
Industry14	0.4126	0.5085	0.6584	0.4171
IntroduceMonth2	0.7635	0.3824	3.9864	0.0459
IntroduceMonth3	0.8206	0.4070	4.0651	0.0438
IntroduceMonth4	0.0352	0.3959	0.0079	0.9293
IntroduceMonth5	0.2185	0.3811	0.3287	0.5664
IntroduceMonth6	-0.3110	0.4451	0.4882	0.4847
IntroduceMonth7	-0.4516	0.3838	1.3845	0.2394
IntroduceMonth8	-0.0110	0.3601	0.0009	0.9757
IntroduceMonth9	-2.0870	0.4736	19.4188	0.0000
IntroduceMonth10	-0.2672	0.3327	0.6450	0.4219
IntroduceMonth11	-1.7700	0.3912	20.4715	0.0000
IntroduceMonth12	2.1930	0.2923	56.2885	0.0000
TeamLeader2	-4.5000	1.2470	13.0224	0.0003
TeamLeader3	-2.2330	0.2782	64.4263	0.0000
TeamLeader4	-3.5970	0.7746	21.5638	0.0000
TeamLeader5	-2.0050	0.3030	43.7868	0.0000

Lampiran 10. Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Inside Fold (Lanjutan)

Variabel	βˆ	SE(B)	Wald	Pr(>|z|)
TeamLeader6	-1.0670	0.2449	18.9824	0.0000
TeamLeader7	0.0298	0.3543	0.0071	0.9329
TeamLeader8	-16.8100	3956.0000	0.0000	0.9966
TeamLeader9	-0.4167	0.2233	3.4823	0.0620
TeamLeader10	-4.0850	1.2120	11.3600	0.0007
Source2	0.3252	0.4102	0.6285	0.4280
Source3	0.4766	1.3630	0.1223	0.7266
Source4	2.8280	5.1110	0.3062	0.5800
Source5	-1.1570	0.8351	1.9195	0.1658
Source6	-15.3100	3956.0000	0.0000	0.9969
Source7	-0.2082	0.4587	0.2060	0.6499
Source8	-5.0800	1.2880	15.5559	0.0001
Source9	0.0699	0.4081	0.0293	0.8641
Source10	-18.5300	701.4000	0.0007	0.9789
Source11	-2.9060	0.6465	20.2048	0.0000
Source12	-0.4627	1.0510	0.1938	0.6598

Lampiran 11. Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Outside Fold

Variabel	$\hat{\beta}$	SE($\hat{\beta}$)	Wald	Pr(>|z|)
SpecialProject1	-0.9305	0.1728	28.9965	0.0000
Industry2	-0.0545	0.6943	0.0062	0.9375
Industry3	-0.7331	0.7143	1.0533	0.3047
Industry4	-0.8865	0.5599	2.5069	0.1134
Industry5	-15.4200	520.2000	0.0009	0.9764
Industry6	0.0366	0.7398	0.0024	0.9605
Industry7	-0.3086	0.6828	0.2043	0.6513
Industry8	-0.2201	0.5645	0.1520	0.6967
Industry9	0.3017	0.4756	0.4024	0.5259
Industry10	2.4250	0.4071	35.4831	0.0000
Industry11	-1.0330	0.6060	2.9057	0.0881
Industry12	-0.4232	1.1370	0.1385	0.7097
Industry13	-0.8018	0.5555	2.0834	0.1489
Industry14	-0.9102	0.5622	2.6212	0.1054
IntroduceMonth2	0.0476	0.5520	0.0074	0.9313
IntroduceMonth3	1.0820	0.4860	4.9566	0.0259
IntroduceMonth4	-0.1319	0.4992	0.0698	0.7916
IntroduceMonth5	0.2570	0.4655	0.3048	0.5809
IntroduceMonth6	-0.5085	0.5838	0.7587	0.3837
IntroduceMonth7	0.0055	0.4510	0.0001	0.9902
IntroduceMonth8	-0.3477	0.4842	0.5157	0.4727
IntroduceMonth9	-2.7330	0.7008	15.2087	0.0001
IntroduceMonth10	-1.3880	0.4807	8.3374	0.0039
IntroduceMonth11	-1.9820	0.5060	15.3429	0.0001
IntroduceMonth12	2.2960	0.3538	42.1142	0.0000
TeamLeader2	-3.0930	1.2750	5.8849	0.0153
TeamLeader3	-1.0370	0.3111	11.1111	0.0009
TeamLeader4	-2.2300	0.7865	8.0392	0.0046
TeamLeader5	-1.1060	0.3434	10.3731	0.0013

Lampiran 11. Pengujian Signifikansi Parameter Secara Parsial Model Oversampling Outside Fold (Lanjutan)

Variabel	β´	SE(β̂)	Wald	Pr(>|z|)
TeamLeader6	-0.9960	0.2810	12.5634	0.0004
TeamLeader7	0.3948	0.4249	0.8633	0.3528
TeamLeader8	-14.5600	3956.0000	0.0000	0.9971
TeamLeader9	0.5521	0.2311	5.7074	0.0169
TeamLeader10	-3.2050	1.3320	5.7896	0.0161
Source2	0.8823	0.6698	1.7352	0.1877
Source3	2.1360	1.3900	2.3614	0.1246
Source4	3.5410	5.9610	0.3529	0.5525
Source5	-0.3374	1.1850	0.0811	0.7759
Source6	-11.9600	3956.0000	0.0000	0.9976
Source7	0.8319	0.7223	1.3265	0.2494
Source8	-3.9900	1.4820	7.2485	0.0071
Source9	1.1600	0.6602	3.0872	0.0790
Source10	-17.7200	697.9000	0.0006	0.9797
Source11	-1.1060	0.8070	1.8783	0.1703
Source12	0.6397	1.1940	0.2870	0.5921

Lampiran 12. Odds Ratio Model Oversampling Inside Fold

Variabel	$\hat{\beta}$	Odds
MRR	0.0001	1.0001
Industry2	0.6810	1.9759
Industry3	-0.2448	0.7829
Industry4	-0.6747	0.5093
Industry5	-15.2200	0.0000
Industry6	0.1366	1.1464
Industry7	0.4325	1.5411
Industry8	-0.0371	0.9636
Industry9	1.5010	4.4862
Industry10	3.2620	26.1017
Industry11	-1.3540	0.2582
Industry12	-0.0227	0.9776
Industry13	-0.0954	0.9090
Industry14	0.4126	1.5107
IntroduceMonth2	0.7635	2.1458
IntroduceMonth3	0.8206	2.2719
IntroduceMonth4	0.0352	1.0358
IntroduceMonth5	0.2185	1.2442
IntroduceMonth6	-0.3110	0.7327
IntroduceMonth7	-0.4516	0.6366
IntroduceMonth8	-0.0110	0.9891
IntroduceMonth9	-2.0870	0.1241
IntroduceMonth10	-0.2672	0.7655
IntroduceMonth11	-1.7700	0.1703
IntroduceMonth12	2.1930	8.9621
TeamLeader2	-4.5000	0.0111
TeamLeader3	-2.2330	0.1072
TeamLeader4	-3.5970	0.0274

Lampiran 12. Odds Ratio Model Oversampling Inside Fold (Lanjutan)

Variabel	$\hat{\beta}$	Odds
TeamLeader5	-2.0050	0.1347
TeamLeader6	-1.0670	0.3440
TeamLeader7	0.0298	1.0303
TeamLeader8	-16.8100	0.0000
TeamLeader9	-0.4167	0.6592
TeamLeader10	-4.0850	0.0168
Source2	0.3252	1.3843
Source3	0.4766	1.6106
Source4	2.8280	16.9116
Source5	-1.1570	0.3144
Source6	-15.3100	0.0000
Source7	-0.2082	0.8120
Source8	-5.0800	0.0062
Source9	0.0699	1.0724
Source10	-18.5300	0.0000
Source11	-2.9060	0.0547
Source12	-0.4627	0.6296

Lampiran 13. Odds Ratio Model Oversampling Outside Fold

Variabel	$\hat{\beta}$	Odds
SpecialProject1	-0.9305	0.3944
Industry2	-0.0545	0.9470
Industry3	-0.7331	0.4804
Industry4	-0.8865	0.4121
Industry5	-15.4200	0.0000
Industry6	0.0366	1.0373
Industry7	-0.3086	0.7345
Industry8	-0.2201	0.8024
Industry9	0.3017	1.3522
Industry10	2.4250	11.3022
Industry11	-1.0330	0.3559
Industry12	-0.4232	0.6549
Industry13	-0.8018	0.4485
Industry14	-0.9102	0.4024
IntroduceMonth2	0.0476	1.0488
IntroduceMonth3	1.0820	2.9506
IntroduceMonth4	-0.1319	0.8764
IntroduceMonth5	0.2570	1.2930
IntroduceMonth6	-0.5085	0.6014
IntroduceMonth7	0.0055	1.0055
IntroduceMonth8	-0.3477	0.7063
IntroduceMonth9	-2.7330	0.0650
IntroduceMonth10	-1.3880	0.2496
IntroduceMonth11	-1.9820	0.1378
IntroduceMonth12	2.2960	9.9344
TeamLeader2	-3.0930	0.0454
TeamLeader3	-1.0370	0.3545
TeamLeader4	-2.2300	0.1075

Lampiran 13. Odds Ratio Model Oversampling Outside Fold (Lanjutan)

Variabel	$\hat{\beta}$	Odds
TeamLeader5	-1.1060	0.3309
TeamLeader6	-0.9960	0.3694
TeamLeader7	0.3948	1.4841
TeamLeader8	-14.5600	0.0000
TeamLeader9	0.5521	1.7369
TeamLeader10	-3.2050	0.0406
Source2	0.8823	2.4165
Source3	2.1360	8.4655
Source4	3.5410	34.5014
Source5	-0.3374	0.7136
Source6	-11.9600	0.0000
Source7	0.8319	2.2977
Source8	-3.9900	0.0185
Source9	1.1600	3.1899
Source10	-17.7200	0.0000
Source11	-1.1060	0.3309
Source12	0.6397	1.8959

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS

Sabilah Margirizki, lahir di Bogor pada Mei 1998, merupakan anak kedua dari tiga bersaudara. Penulis menempuh pendidikan formal di SMA Negeri 1 Kota Bogor dan melanjutkan Pendidikan sarjana di Departemen Statistika FSAD ITS. Semasa perkuliahan, penulis aktif mengikuti kegiatan internasionalisasi berupa konferensi dan pertukaran pelajar selama dua semester. Penulis juga terdaftar sebagai anggota Tim Barunastra ITS, salah satu tim robotik internasional ITS, bertindak sebagai Non-technical Manager dan berhasil meraih juara pertama International Roboboat Competition di South Daytona, Florida pada 2018 dan 2019. Penulis pernah mengikuti Musyawarah Kerja Nasional oleh Ikatan Himpunan Mahasiswa Statistika se-Indonesia pada 2017. Kritik dan saran membangun mengenai Tugas Akhir dapat disampaikan melalui margi.sabil@yahoo.com.