

**PERBANDINGAN PERFORMA MODEL PREDIKSI *CUSTOMER CHURN*
BERBASIS *MACHINE LEARNING* PADA *FASHION E-COMMERCE***

(Skripsi)

Oleh

DWI LILYAWATI

NPM 1915061005



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
2023**

**PERBANDINGAN PERFORMA MODEL PREDIKSI *CUSTOMER CHURN*
BERBASIS *MACHINE LEARNING* PADA *FASHION E-COMMERCE***

Oleh

DWI LILYAWATI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA TEKNIK**

Pada

**Program Studi Teknik Informatika
Jurusan Teknik Elektro
Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

ABSTRAK

PERBANDINGAN PERFORMA MODEL PREDIKSI *CUSTOMER CHURN* BERBASIS *MACHINE LEARNING* PADA *FASHION E-COMMERCE*

Oleh

DWI LILYAWATI

Dalam era perkembangan bisnis *e-commerce*, terutama di industri *fashion*, persaingan yang semakin ketat antara beragam platform *e-commerce* menjadi salah satu tantangan utama yang tidak dapat diabaikan. Adanya berbagai macam produk dan layanan pada *e-commerce* memungkinkan pelanggan secara bebas dapat meninggalkan perusahaan untuk mencari produk atau layanan yang sesuai dengan kebutuhan. Fenomena ini dikenal dengan istilah *churn*. Berdasarkan permasalahan tersebut, penelitian ini berfokus untuk mengembangkan model yang dapat memprediksi pelanggan yang akan *churn* pada *fashion e-commerce* sehingga perusahaan dapat mengambil tindakan proaktif untuk menjaga pelanggan yang ada agar tidak *churn*. Metode yang digunakan dalam penelitian ini adalah *Cross Industry Standard Process for Data Mining* (CRISP-DM). Model prediksi ini dibangun dengan pendekatan klasifikasi menggunakan algoritma *machine learning* yaitu *Logistic Regression*, *Random Forest* dan *XGBoost*. Kategori *churn* dibagi menjadi dua yaitu *churn* dan *not churn*. Hasil penelitian menunjukkan bahwa *XGBoost* memiliki performa yang paling tinggi di antara ketiga model, dengan akurasi sebesar 97%, *precision* sebesar 97%, *recall* sebesar 98%, *f1-score* sebesar 98% dan nilai AUC mencapai 0.995. Hasil tersebut menjadikan *XGBoost* sebagai model terbaik dalam memprediksi pelanggan yang *churn*. Selain itu, penggunaan seleksi fitur dalam membangun ulang model melalui *feature importance* pada *XGBoost* berhasil mempercepat waktu komputasi dari 2.471 detik menjadi 0.584 detik dan menghasilkan performa kinerja model yang sama seperti menggunakan seluruh fitur. Hal tersebut membuktikan bahwa penggunaan seleksi fitur melalui *feature importance* dapat mengoptimalkan kinerja model sehingga lebih efisien dari segi waktu komputasi dan tetap efektif. Penelitian ini juga menghasilkan visualisasi dan rekomendasi untuk mengurangi *customer churn*.

Kata kunci : *customer churn*, prediksi, *logistic regression*, *random forest*, *xgboost*

ABSTRACT

PERFORMANCE COMPARISON OF CUSTOMER CHURN PREDICTION MODELS BASED ON MACHINE LEARNING IN FASHION E-COMMERCE

By

DWI LILYAWATI

In the era of e-commerce business development, especially in the fashion industry, the intensifying competition among various e-commerce platforms has emerged as a significant challenge that cannot be overlooked. The wide array of products and services in e-commerce enables customers to freely depart from a company in pursuit of products or services that align with their needs. This phenomenon is commonly referred to as churn. In light of these issues, this research is focused on developing a model for predicting customer churn in fashion e-commerce, enabling companies to take proactive measures to retain existing customers and prevent churn. The research employs the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology. The predictive model is constructed using a classification approach, employing machine learning algorithms such as Logistic Regression, Random Forest, and XGBoost. Churn is categorized into two groups: churn and not churn. The research findings indicate that XGBoost outperforms the other models, achieving an accuracy rate of 97%, precision of 97%, recall of 98%, an F1-score of 98%, and an AUC value of 0.995. These results establish XGBoost as the best model for predicting customer churn. Furthermore, the utilization of feature selection in model reconstruction through feature importance in XGBoost significantly reduces computation time from 2.471 seconds to 0.584 seconds, while maintaining identical model performance. This demonstrates that the use of feature selection through feature importance optimizes model efficiency in terms of computation time, all while remaining effective. The research also yields visualizations and recommendations aimed at reducing customer churn.

Keywords : Customer Churn, Prediction, Logistic Regression, Random Forest, XGBoost

Judul : **PERBANDINGAN PERFORMA MODEL
PREDIKSI *CUSTOMER CHURN* BERBASIS
MACHINE LEARNING PADA *FASHION
ECOMMERCE***

Nama Mahasiswa : **Dwi Liliyawati**


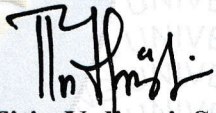
Nomor Pokok Mahasiswa : **1915061005**

Program Studi : **Teknik Informatika**

Fakultas : **Teknik**




1. **Komisi Pembimbing**

 **Dr. Eng. Ir. Mardiana, S.T., M.T., IPM**  **Ir. Titin Yulianti, S.T., M.Eng.**
NIP. 19720316 199903 2 002 NIP. 198807092019032015

2. **Mengetahui**

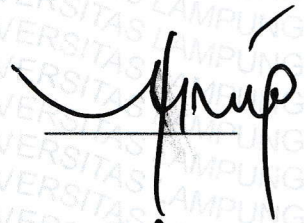
Ketua Jurusan Teknik Elektro


Herlinawati, S.T., M.T.
NIP. 19710314 199903 2 001

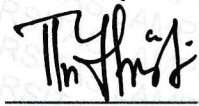
MENGESAHKAN

1. Tim Penguji

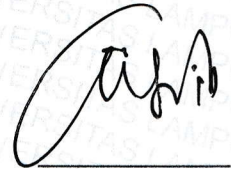
Ketua : **Dr. Eng. Ir. Mardiana, S.T., M.T., IPM.**



Sekretaris : **Ir. Titin Yulianti, S.T., M.Eng.**



Penguji : **Ir. Gigih Forda Nama, S.T., M.T.I., IPM.**



2. Dekan Fakultas Teknik



Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. J
NIP. 19750928200112 1 002

Tanggal Lulus Ujian Skripsi : 18 September 2023

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya dengan judul “Perbandingan Performa Model Prediksi *Customer Churn* Berbasis *Machine Learning* pada *Fashion E-Commerce*” dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 18 September 2023

Pembuat pernyataan,



Dwi Liliyawati

NPM 1915061005

RIWAYAT HIDUP



Penulis dilahirkan di Negara Ratu, pada tanggal 27 Juli 2001. Penulis merupakan anak kedua dari tiga bersaudara dari pasangan Bapak Paiman dan Ibu Widiawati.

Penulis menyelesaikan pendidikannya di MIN 6 Lampung Utara pada tahun 2013, MTsN 3 Lampung Utara pada tahun 2016 dan SMA Negeri 2 Kotabumi pada tahun 2019. Pada tahun 2019, penulis terdaftar sebagai mahasiswa Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik Universitas Lampung melalui jalur SNMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan, antara lain:

1. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Pendidikan dan Pengembangan Diri, Divisi Pendidikan periode 2019/2020 dan periode 2020/2021.
2. Menjadi asisten Laboratorium Teknik Komputer Universitas Lampung pada tahun 2021 sampai tahun 2022.
3. Mengikuti Program Kredensial Mikro Mahasiswa Indonesia (KMMI) dengan mengambil Kursus Konsep, Strategi, dan Implementasi Internet of Things (IoT) pada tahun 2021.
4. Mengikuti program Magang Bersertifikat Kampus Merdeka Batch 2 di PT Mojadi Aplikasi Indonesia sebagai Data Scientist pada 14 Februari 2022 sampai 22 Juli 2022.
5. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya dengan mengambil kelas Data Science di Startup Campus pada tahun 2022.
6. Melaksanakan Kuliah Kerja Nyata selama 40 hari di Desa Gunung Betuah, Kecamatan Abung Barat, Kabupaten Lampung Utara, Lampung, Indonesia

MOTTO

“Ya Tuhanku, lapangkanlah dadaku, dan mudahkanlah untukku urusanku, dan lepaskanlah kekakuan dari lidahku, agar mereka mengerti perkataanku.”

(QS. Taha, 20: 25-28)

“Cukup Allah (menjadi penolong) bagi kami dan Dia sebaik-baik pelindung”

(QS. Ali Imran, 3:173)

“Siapa yang menempuh jalan untuk mencari ilmu, maka Allah akan mudahkan baginya jalan menuju surga”

(HR. Muslim, no. 2699)

“Ini bukan tentang menang atau kalah, setiap dari kita adalah pemenang.

So take your time”

(트레저)

“최고가 되는 것보다 최선을 다하는 것이 낫다”

(방 예담)

PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji syukur kepada Allah SWT atas segala rahmat dan karunia-Nya sehingga saya dapat menyelesaikan skripsi ini. Shalawat serta salam teriring kepada Nabi Muhammad SAW sebagai suri tauladan *akhlakul karimah*.

KUPERSEMBAHKAN KARYA INI KEPADA:

“Kedua Orang Tuaku Tercinta

Yang senantiasa selalu memberikan segalanya yang terbaik, serta selalu melantunkan doa yang tiada henti-hentinya untukku. Kuucapkan terima kasih sebesar-besarnya karena telah memberikan kesempatan kepadaku untuk menimba ilmu hingga ke perguruan tinggi. Terima kasih telah membesarkan, mendidik, dan memberikan contoh kepadaku sehingga aku tumbuh menjadi pribadi yang jujur, sabar, penuh kasih sayang, berkecukupan, dan bahagia yang akan selalu aku syukuri seumur hidupku. Semoga dengan ilmu yang ku dapatkan dari hasil jerih payah kalian menyekolahkanku akan menjadi amal Jariyah bagi kalian”

“Diriku yang telah melakukan yang terbaik. Kamu sudah bekerja keras, terima kasih atas segala usaha yang telah diberikan. Terima kasih telah mau mencoba banyak hal dengan rasa penasaran yang cukup tinggi itu. Mari kita bekerja lebih keras lagi di masa depan. Semoga impian dan cita-citamu dapat segera tercapai.”

“Seluruh mentor-mentorku, terima kasih telah membimbing dan memberikan ilmu serta pengalaman baru untukku. Semoga dikemudian hari dapat bertemu kembali, sukses selalu ”

“Almamater tercinta, Universitas Lampung dan Jurusan Teknik Elektro”

SANWACANA

Puji syukur kehadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga skripsi dengan judul “Perbandingan Performa Model Prediksi *Customer Churn* Berbasis *Machine Learning* pada *Fashion E-Commerce*” dapat terselesaikan sebagaimana mestinya. Dalam pelaksanaan dan pembuatan skripsi ini terdapat banyak bantuan, bimbingan, serta dukungan baik secara moril maupun materil yang sangat berharga dari berbagai pihak. Oleh karena itu, diucapkan terima kasih kepada semua pihak yang telah membantu, khususnya kepada:

1. Kedua orang tua, kakak dan adik tercinta yang selalu tidak hentinya memberikan semangat, doa dan dukungan, serta materi hingga penulis dapat menyelesaikan penelitian ini dengan sangat baik.
2. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
3. Ibu Herlinawati, S.T., M.T. selaku ketua Jurusan Teknik Elektro Universitas Lampung;
4. Bapak Mona Arif Muda, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Universitas Lampung;
5. Ibu Dr.Eng., Ir. Mardiana, S.T., M.T., IPM. selaku Pembimbing Utama dan Dosen Pembimbing Akademik yang telah membantu proses pengerjaan penelitian dengan cara memberikan bimbingan, semangat dan mencurahkan waktunya yang demikian banyak selama pengerjaan skripsi ini;
6. Ibu Ir. Titin Yulianti, S.T., M.Eng. selaku Pembimbing Pendamping yang telah memberikan waktu, dukungan serta bimbingan secara mendetail dalam menyelesaikan skripsi ini;

7. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., IPM. yang telah bersedia menjadi penguji dalam sidang skripsi serta memberikan banyak saran dan masukan terhadap penelitian ini;
8. Ibu Prof. Dr. Lindriana Sari, S.E., M.Si., Ak., CA yang telah membantu dalam menyusun rekomendasi strategi pemasaran dalam penelitian ini;
9. Mbak Rika selaku *Admin* Program Studi Teknik Informatika yang telah banyak membantu penulis dalam urusan administrasi selama perkuliahan dan penelitian;
10. Tim Finlandia yang saling membantu dan bekerja sama dalam menyelesaikan *final project Data Science* di Startup Campus;
11. Teman-teman seperjuangan yang menemani, membantu, dan mendukung penulis selama kuliah dan kegiatan penelitian, Yovanta Anjelina, Alfiyah Widiyaningsih, Meilika Dwi Putri, Husniatun Aini, Selvia Eldina, Silvia Naim, Reistha Ramadhanty, dan Niwayan Dinayani;
12. Fela Rosa, Bening Damayanti Nurjanah, Reka Tiana, dan Meisi Yulanda sebagai teman-teman yang selalu menjadi tempat bercerita keluh kesah selama ini dengan memberikan semangat dan motivasi yang membangun;
13. Treasure yang selalu menjaga *mental health* penulis dengan memberikan inspirasi dan kebahagiaan, serta pesan penyemangat setiap hari senin.

Akhir kata, semoga laporan ini dapat menjadi referensi bagi pengembangan keilmuan di bidang Teknik Informatika dan bermanfaat bagi yang membacanya.

DAFTAR ISI

	halaman
DAFTAR GAMBAR	vii
DAFTAR TABEL.....	x
I. PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	3
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	4
1.6 Sistematika Penulisan.....	4
II. TINJAUAN PUSTAKA.....	6
2.1 <i>E-commerce</i>	6
2.2 <i>Customer Churn</i>	8
2.3 <i>Customer Relationship Management (CRM)</i>	8
2.4 CRISP-DM	9
2.5 Python.....	11
2.6 Google Colab.....	12
2.7 <i>Data Integration</i>	13
2.8 <i>Cleaning Data</i>	13
2.8.1 <i>Handling Missing Values</i>	13
2.8.2 <i>Formatting Data</i>	14
2.8.3 <i>Removing Duplicates</i>	15
2.9 <i>Feature Engineering</i>	15
2.10 <i>Data Transformation</i>	17
2.10.1 Normalisasi Data.....	17
2.10.2 <i>One Hot Encoding</i>	18

2.11	<i>Machine learning</i>	18
2.12	Klasifikasi.....	20
2.13	<i>Ensemble Learning</i>	20
2.13.1	<i>Random Forest</i>	22
2.13.2	<i>XGBoost</i>	25
2.14	<i>Logistic Regression</i>	29
2.15	Tuning Parameter dengan <i>Grid Search CV</i>	31
2.16	Bias dan Varians.....	32
2.17	Pengukuran Kinerja Algoritma Klasifikasi	33
2.17.1	<i>Classification Report and Confusion Matrix</i>	33
2.17.2	<i>Receiver Operating Characteristic dan Area Under the Curve</i>	35
2.18	Perbandingan Algoritma <i>Logistic Regression, Random Forest, dan XGBoost</i>	36
2.19	Google Looker Studio	37
2.20	Penelitian Terkait	38
III.	METODOLOGI PENELITIAN	41
3.1	Waktu dan Tempat	41
3.2	Alat dan Bahan Penelitian	42
3.2.1	Alat.....	42
3.2.2	Bahan.....	42
3.3	Tahapan Penelitian	43
3.3.1	Tahap Pemahaman Bisnis (<i>Business Understanding Phase</i>).....	45
3.3.2	Tahap Pemahaman Data (<i>Data Understanding Phase</i>)	45
3.3.3	Tahap Persiapan Data (<i>Data Preparation Phase</i>).....	46
3.3.4	Tahap Pemodelan (<i>Modelling Phase</i>)	48
3.3.5	Tahap Evaluasi (<i>Evaluation</i>).....	49
3.3.6	Tahap Penyebaran (<i>Deployment</i>)	50
IV.	HASIL DAN PEMBAHASAN	51
4.1	Pemahaman Bisnis	51
4.2	Pemahaman Data	53
4.2.1	Mengumpulkan data.....	54
4.2.2	Memeriksa Kualitas Data.....	58

4.2.3	Melakukan Eksplorasi Data	60
4.3	Persiapan Data	68
4.3.1	<i>Merge Data</i>	69
4.3.2	<i>Feature Engineering</i>	71
4.3.3	<i>Labelling</i>	74
4.4	Pemodelan	77
4.4.1	Eksplorasi Data	78
4.4.2	<i>Data Preprocessing</i>	83
4.4.3	<i>Splitting Data</i>	85
4.4.4	<i>Classification</i>	85
4.5	Evaluasi	99
4.5.1	Hasil <i>testing</i> klasifikasi menggunakan model <i>Logistic Regression</i>	99
4.5.2	Hasil <i>testing</i> klasifikasi menggunakan model <i>Random Forest</i>	101
4.5.3	Hasil <i>testing</i> klasifikasi menggunakan model <i>XGBoost</i>	103
4.5.4	Perbandingan Kinerja Algoritma <i>Logistic Regression</i> , <i>Random Forest</i> , dan <i>XGBoost</i>	105
4.6	Penyebaran	108
4.6.1	Penerapan Model untuk Data Baru	108
4.6.2	Visualisasi Data.....	109
4.6.3	Rekomendasi Strategi Pemasaran	114
V.	KESIMPULAN DAN SARAN	121
5.1	Kesimpulan.....	121
5.2	Saran	122
	DAFTAR PUSTAKA	123
	LAMPIRAN.....	127

DAFTAR GAMBAR

Gambar	Halaman
2.1 Fase model referensi CRISP-DM [12].....	9
2.2 Tampilan Google Colab	12
2.3 Penerapan <i>Feature Engineering</i> [18].....	16
2.4 Penerapan <i>one hot encoding</i> pada kategori dari ketiga kota [16]	18
2.5 Bagan <i>Machine Learning</i> [21]	19
2.6 Metode bagging menggunakan suara mayoritas [24]	21
2.7 Metode boosting menggunakan strategi rata-rata tertimbang[24]	22
2.8 Pemisahan <i>node</i> dalam <i>Random Forest</i> berdasarkan <i>subset</i> acak dari fitur untuk setiap pohon [25].....	23
2.9 Diagram skema algoritma <i>XGBoost</i>	26
2.10 <i>Maximum Likelihood</i>	30
2.11 <i>Model complexity based on prediction error</i> [22]	32
2.12 <i>Confusion Matrix</i> [14].....	34
2.13 Contoh penggunaan Google Looker Studio [34]	38
3.1 Tahapan Penelitian	43
3.2 Tahapan penelitian metode CRISP-DM.....	44
3.3 Contoh derivasi variabel <i>partial churning</i> [40]	46
4.1 Contoh <i>missing value</i> pada dataset produk	59
4.2 Distribusi <i>customer</i> berdasarkan umur.....	60
4.3 <i>Customer</i> berdasarkan <i>gender</i>	60
4.4 <i>Customer</i> berdasarkan kota	61
4.5 <i>Customer</i> berdasarkan tipe <i>device</i>	61
4.6 Jumlah <i>customer</i> baru setiap tahun	62
4.7 Produk yang ditawarkan berdasarkan <i>gender</i>	63

4.8 Metode pembayaran yang sering digunakan.....	63
4.9 Penggunaan kode promo oleh pelanggan.....	64
4.10 Total pelanggan yang aktif bertransaksi setiap bulannya	64
4.11 Transaksi yang dilakukan pelanggan berdasarkan hari.....	65
4.12 Transaksi yang dilakukan pelanggan berdasarkan jam	66
4.13 <i>Top event</i> yang dilakukan pelanggan	67
4.14 <i>Keyword</i> yang paling banyak dicari oleh pelanggan.....	68
4.15 Library yang digunakan pada persiapan data.....	68
4.16 <i>Merge</i> dataset	69
4.17 Menghapus data duplikasi dan seleksi transaksi status sukses	69
4.18 Hasil <i>merge</i> dataset <i>customer</i> , transaksi, dan produk	70
4.19 Proses pembagian data pelatihan dan data pengujian	71
4.20 Pembagian 4 periode setiap 3 bulan.....	74
4.21 Menghitung <i>monetary</i> setiap periode.....	74
4.22 Hasil penanganan <i>missing value</i>	75
4.23 Fungsi pendefinisian <i>churn</i>	75
4.24 Hasil pelabelan <i>churn</i> pada dataset	76
4.25 <i>Library</i> yang diperlukan untuk pemodelan	77
4.26 Distribusi kelas <i>churn</i>	78
4.27 Eksplorasi data berdasarkan variabel kategorik.....	79
4.28 Eksplorasi data berdasarkan variabel numerik.....	80
4.29 Visualisasi <i>heatmap</i> dalam menampilkan korelasi antar variabel	82
4.30 Hasil dataset yang telah dinormalisasi	83
4.31 Hasil dataset yang telah dilakukan <i>one hot encoding</i>	84
4.32 Proses klasifikasi menggunakan <i>Logistic Regression</i>	85
4.33 <i>Feature Importance</i> pada <i>Model Logistic Regression</i>	87
4.34 Proses klasifikasi menggunakan <i>Random Forest</i> sebelum <i>hyperparameter</i> . 89	
4.35 Hasil <i>classification report</i> pada <i>training model Random Forest</i>	89
4.36 Hasil <i>classification report</i> pada <i>testing model Random Forest</i>	89
4.37 <i>Feature Importance Random Forest</i>	91
4.38 Proses klasifikasi menggunakan <i>XGBoost</i> sebelum <i>hyperparameter</i>	94

4.39 Hasil <i>classification report</i> pada <i>training model XGBoost</i> sebelum <i>hyperparameter tuning</i>	94
4.40 Hasil <i>classification report</i> pada <i>testing model XGBoost</i> sebelum <i>hyperparameter tuning</i>	95
4.41 <i>Feature Importance XGBoost</i>	97
4.42 Kurva ROC <i>Logistic Regression</i>	100
4.43 Kurva ROC <i>Random Forest</i>	102
4.44 Kurva ROC <i>XGBoost</i>	104
4.45 <i>Preprocessing Data Baru</i>	108
4.46 Penerapan model <i>XGBoost</i> pada data baru	109
4.47 <i>Prediction Customer Dashboard</i>	110
4.48 Distribusi pelanggan berdasarkan gender	114
4.49 Distribusi pelanggan <i>churn</i> berdasarkan rentang usia dan gender	115
4.50 Distribusi pelanggan berdasarkan metode pembayaran	116
4.51 Distribusi pelanggan berdasarkan metode pembayaran yang diminati	117

DAFTAR TABEL

Tabel	Halaman
2.1 Parameter pada algoritma <i>Random Forest</i> [26]	25
2.2 Parameter pada algoritma <i>XGBoost</i> [28].....	28
2.3 Parameter pada algoritma <i>Logistic Regression</i> [29]	31
2.4 Kategori nilai AUC	35
2.5 Perbandingan Algoritma <i>Machine Learning</i>	36
2.6 Penelitian Terkait	40
3.1 Jadwal Penelitian.....	41
3.2 Alat dan Bahan Penelitian	42
4.1 Dataset yang digunakan	53
4.2 Atribut dataset <i>customer</i>	54
4.3 Atribut Dataset Produk.....	55
4.4 Atribut dataset transaksi	56
4.5 Atribut dataset <i>click stream</i>	57
4.6 Jumlah <i>missing value</i> pada setiap dataset	58
4.7 Kategori produk yang ditawarkan	62
4.8 Hasil <i>feature engineering</i>	73
4.9 Proporsi Data <i>Training</i> dan Data <i>Testing</i>	85
4.10 <i>Confussion matrix</i> dari data <i>training Logistic Regression</i>	86
4.11 <i>Classification report training model Logistic Regression</i>	86
4.12 Hasil Tuning Parameter <i>Random Forest</i>	90
4.13 <i>Confussion matrix</i> dari data <i>training Random Forest</i>	90
4.14 <i>Classification report training model Random Forest</i> setelah <i>hyperparameter tuning</i>	91
4.15 Hasil Tuning Parameter <i>XGBoost</i>	95

4.16 <i>Confussion matrix</i> dari data <i>training XGBoost</i>	96
4.17 <i>Classification report training model XGBoost</i>	96
4.18 <i>Confussion matrix</i> dari data <i>testing Logistic Regression</i>	99
4.19 <i>Classification report testing model Logistic Regression</i>	99
4.20 <i>Confussion matrix</i> dari data <i>testing Random Forest</i>	101
4.21 <i>Classification report testing model Random Forest</i>	101
4.22 <i>Confussion matrix</i> dari data <i>testing XGBoost</i>	103
4.23 <i>Classification report testing model XGBoost</i>	103
4.24 Hasil pengukuran kinerja	105
4.25 Hasil pengukuran kinerja pada model yang telah dibangun ulang menggunakan fitur penting pada setiap algoritma.....	107

I. PENDAHULUAN

1.1 Latar Belakang

Kemajuan dunia digital berkembang dengan cepat, bahkan setiap tahunnya tidak dapat terlepas dari perkembangan dunia *online*. Dampaknya sangat signifikan bagi berbagai sektor, termasuk bisnis dan ekonomi. Salah satu contohnya adalah sektor bisnis yang berfokus pada platform *e-commerce*. *E-commerce* merujuk pada pembelian, penjualan, serta pemasaran produk atau layanan yang difasilitasi melalui internet. Keberadaan *e-commerce* telah mempermudah dan memberikan kenyamanan dalam proses pemenuhan kebutuhan. Kini, kegiatan tersebut bisa dilakukan dari rumah tanpa perlu berinteraksi secara langsung sehingga dapat menghemat waktu.

Bisnis *e-commerce* di Indonesia mengalami pertumbuhan yang signifikan seiring dengan adanya pandemi sejak awal tahun 2020. *E-commerce* mengalami peningkatan *customer* sebanyak 5-10 kali selama pandemi dengan penambahan *customer* baru mencapai 51% [1]. Berdasarkan laporan yang dipublikasikan pada tahun 2020 oleh Google, Temasek, dan Bain Company menyatakan bahwa *e-commerce* tetap menjadi pendorong pertumbuhan utama bagi ekonomi Indonesia sebesar 54% mengimbangi penurunan di sektor pariwisata yang dilihat dari nilai *Gross Merchandise Value* (GMV) [2]. Kemudian pada tahun 2021, *e-commerce* menjadi penyumbang terbesar bagi ekonomi digital Indonesia dengan nilai transaksinya mencapai US\$53 miliar. Proyeksinya menunjukkan peningkatan hingga mencapai US\$104 miliar pada tahun 2025 dengan tingkat pertumbuhan sebesar 18% [3].

Dalam era digital yang semakin maju, tren belanja mengalami transformasi signifikan. *E-commerce* telah menjadi pilihan yang semakin populer bagi banyak konsumen dalam mencari produk, salah satunya *fashion*. Menurut survei JakPat, mayoritas masyarakat Indonesia cenderung lebih memilih berbelanja produk busana melalui platform *e-commerce* daripada berbelanja di toko fisik. Hasil survei tersebut mengungkapkan bahwa pada semester pertama tahun 2022, sekitar 58% dari responden memilih *e-commerce* sebagai pilihan utama dalam berbelanja *fashion*, sementara hanya 29% yang memilih untuk berbelanja produk *fashion* secara langsung di toko fisik [4].

Seiring dengan pertumbuhan bisnis *e-commerce* khususnya di bidang *fashion*, salah satu tantangan utamanya adalah semakin meningkatnya tingkat persaingan di pasar. Adanya berbagai macam produk dan layanan pada *e-commerce* saat ini memungkinkan pelanggan secara bebas dapat meninggalkan perusahaan untuk mencari produk atau layanan yang sesuai dengan kebutuhan. Fenomena ini dikenal dengan istilah *churn*. *Customer* dikatakan *churn* ketika *customer* berhenti melakukan transaksi atau berhenti berlangganan layanan suatu perusahaan [5]. Kehilangan pelanggan yang sudah ada dapat merugikan perusahaan sebab pada *e-commerce* biaya yang dikeluarkan untuk mendapatkan pelanggan baru jauh lebih mahal daripada mempertahankan pelanggan yang sudah ada [6].

Untuk mempertahankan pelanggan, perusahaan harus meningkatkan pengelolaan interaksi yang baik dengan pelanggan, mengetahui pola perilaku, karakteristik, dan preferensi pelanggan terhadap produk, serta harus mengidentifikasi lebih awal pelanggan yang memiliki peluang tertinggi dalam meninggalkan perusahaan. Penting bagi perusahaan untuk memprediksi *customer* yang akan *churn* sehingga perusahaan dapat mengambil tindakan proaktif untuk menjaga pelanggan yang sudah ada agar tidak *churn*. Berdasarkan permasalahan tersebut, diperlukan model *machine learning* yang dapat memprediksi dan mengklasifikasikan pelanggan mana yang akan *churn* di masa depan, serta mengidentifikasi masalah yang menyebabkan *churn*. Hasil analisis data prediksi dan eksplorasi data perusahaan akan digunakan untuk menyusun strategi pemasaran yang dapat diterapkan dalam mencegah pelanggan yang akan *churn*.

1.2 Perumusan Masalah

Berdasarkan latar belakang, kajian masalah yang mendasari penelitian ini adalah:

1. Bagaimana membuat model *machine learning* untuk memprediksi pelanggan yang akan melakukan *churn*?
2. Manakah dari ketiga algoritma yaitu *XGBoost*, *Random Forest*, dan *Logistic Regression* yang akan memberikan performa terbaik dalam memprediksi *customer churn*?
3. Apa saja fitur-fitur penting yang dapat mempengaruhi pelanggan dalam melakukan *churn*?
4. Bagaimana penerapan Google Looker Studio dapat mempermudah visualisasi data?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Mengembangkan model untuk memprediksi pelanggan yang akan *churn* pada *fashion e-commerce*
2. Membandingkan tiga algoritma yaitu *XGBoost*, *Random Forest*, dan *Logistic Regression* untuk menentukan algoritma terbaik yang dapat digunakan dalam memprediksi pelanggan *churn*.
3. Melakukan pemilihan fitur berdasarkan analisis *feature importance* yang mempengaruhi pelanggan dalam melakukan *churn*.
4. Membuat visualisasi hasil prediksi dengan penerapan Google Looker Studio.

1.4 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat bagi perusahaan *e-commerce* dalam mengetahui model terbaik yang dapat digunakan untuk prediksi *customer churn* dan rekomendasi strategi pemasaran untuk menangani *customer* yang akan *churn*.

1.5 Batasan Masalah

Dalam penelitian ini, pembatasan masalah meliputi hal-hal sebagai berikut:

1. Model prediksi *churn* yang diujikan menggunakan tiga algoritma yaitu *XGBoost*, *Random Forest*, dan *Logistic Regression*.
2. Hanya membahas data transaksi penjualan produk dan data *click stream customer* untuk periode 1 Januari 2020 sampai 31 Juli 2022.
3. Tidak mengembangkan sistem untuk memprediksi pelanggan *churn*, melainkan hanya membuat visualisasi data berupa dashboard pada hasil prediksi.

1.6 Sistematika Penulisan

Adapun sistematika penulisan laporan penelitian ini adalah sebagai berikut:

1) PENDAHULUAN

Pada bagian ini membahas terkait latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan.

2) TINJAUAN PUSTAKA

Pada bagian ini membahas teori – teori penunjang penelitian seperti *E-commerce*, *Customer Churn*, *Customer Relationship Management*, Python, Google Colab, *Data Integration*, *Cleaning Data*, *Feature Engineering*, *Data Transformation*, *Machine learning*, Klasifikasi, *Logistic Regression*, *XGBoost*, *Random Forest*, Tuning Parameter dengan *Grid Search CV*, Bias dan Varians, Pengukuran Kinerja Algoritma Klasifikasi, Google Looker Studio, dan penelitian terkait.

3) METODOLOGI PENELITIAN

Pada bagian ini membahas waktu dan tempat penelitian, alat dan bahan, metode yang digunakan dan diagram alir metode yang dihasilkan.

4) PEMBAHASAN

Pada bagian ini membahas tahapan penelitian meliputi pemahaman bisnis, pemahaman data, persiapan data, pemodelan, evaluasi, dan penyebaran data dalam memberikan analisis terhadap prediksi *customer churn* serta

rekomendasi strategi bisnis yang dapat dilakukan dalam mengurangi *customer churn*.

5) KESIMPULAN DAN SARAN

Pada bagian ini berisi kesimpulan berdasarkan temuan penelitian dan saran untuk pengembangan lebih lanjut.

II. TINJAUAN PUSTAKA

2.1 *E-commerce*

E-commerce didefinisikan secara beragam sesuai dengan perspektif yang memanfaatkannya. *E-commerce* merupakan sebuah bentuk mekanisme bisnis elektronik yang berfokus pada transaksi bisnis individu melalui media internet sebagai cara untuk menukar barang atau jasa antara dua institusi (B-to-B) atau antara institusi dan konsumen (B-to-C) [7]. Teknologi yang diterapkan *e-commerce* memungkinkan melakukan transaksi komersial melintasi batas-batas budaya, regional, nasional, serta batas-batas negara dengan jauh lebih mudah dan hemat biaya dibandingkan dengan toko tradisional. Sehingga *e-commerce* dapat memperoleh pelanggan dengan jangkauan yang lebih luas. Selain itu, adanya *e-commerce* dapat memudahkan pedagang dalam menargetkan pemasaran produk kepada individu tertentu dengan menyesuaikan iklan dengan profil, minat atau pembelian sebelumnya [8].

Saat ini *e-commerce* telah berkembang pesat di seluruh dunia, termasuk Indonesia. Perkembangan *e-commerce* didukung oleh beberapa faktor, antara lain peningkatan ketersediaan internet, kemunculan perangkat seluler, dan semakin digemarinya platform belanja *online*. Pasar *e-commerce* Indonesia telah mampu menarik banyak pelanggan dengan menawarkan berbagai macam produk dan layanan, serta pilihan pembayaran dan pengiriman yang cepat dan nyaman. Contoh perusahaan *e-commerce* di Indonesia diantaranya Tokopedia, Shopee, Lazada, Blibi, Bukalapak, dan yang lainnya.

Adapun keunggulan dari penggunaan *e-commerce* diantaranya:

- 1) Dengan adanya *e-commerce* memungkinkan pelanggan untuk berbelanja dengan nyaman dimanapun dan kapanpun. Sehingga pelanggan dapat

menghemat waktu dan kerumitan pelanggan, terutama bagi pelanggan yang tidak dapat pergi ke toko selama jam kerja.

- 2) Pelanggan mendapatkan penawaran pilihan produk yang lebih luas di *e-commerce* daripada toko tradisional. Pelanggan dapat mencari berbagai produk yang berbeda dari seluruh dunia.
- 3) *E-commerce* sering menawarkan harga yang lebih rendah termasuk penawaran penggunaan promo atau diskon khusus. Hal ini dikarenakan *e-commerce* memiliki biaya produksi yang lebih rendah, seperti biaya sewa tempat dan karyawan.
- 4) *E-commerce* mencantumkan lebih banyak informasi tentang produk yang ditawarkan daripada toko fisik. Ini dikarenakan pada *e-commerce* dapat dengan mudah menyertakan deskripsi rinci, gambar maupun video.
- 5) *E-commerce* dapat memanfaatkan data untuk mempersonalisasikan pengalaman berbelanja bagi setiap pelanggan sehingga dapat meningkatkan penjualan dan kepuasan pelanggan.

Namun *E-commerce* juga memiliki kelemahan diantaranya:

- 1) Masalah keamanan merupakan tantangan dalam penggunaan *e-commerce*. Transaksi yang dilakukan pada *e-commerce* lebih rentan terhadap penipuan daripada transaksi di toko tradisional.
- 2) Pelanggan yang bertransaksi melalui *e-commerce* memiliki tanggungan biaya pengiriman. Biaya pengiriman dapat menjadi tinggi terutama alamat tujuan yang jauh maupun barang yang besar atau berat.
- 3) Pelanggan kemungkinan memiliki kesulitan dalam menilai kualitas produk sebelum membelinya secara *online*. Hal ini dikarenakan pelanggan tidak dapat melihat, menyentuh, atau menilai produk secara langsung.
- 4) Beberapa pelanggan mungkin memiliki keraguan untuk berbelanja secara *online* karena tidak mempercayai keamanan data situs *e-commerce*. Terlebih bila pelanggan mendapatkan pengalaman yang kurang baik selama bertransaksi melalui *e-commerce*.

2.2 *Customer Churn*

Customer dikatakan *churn* ketika *customer* tersebut telah berhenti melakukan transaksi produk atau berhenti menggunakan layanan yang disediakan oleh suatu perusahaan dalam periode waktu tertentu [5]. Pada *e-commerce* tidak ada kontrak antara pelanggan dan perusahaan sehingga pelanggan memiliki kesempatan untuk secara terus menerus mengubah perilaku pembelian tanpa memberi tahu perusahaan. Hal tersebut yang dapat memicu pelanggan untuk melakukan *churn* sebagian atau disebut juga dengan *partial churmer* [9]. Apabila pelanggan yang termasuk dalam *partial churmer* dalam jangka panjang tidak segera ditangani maka terdapat peluang yang besar bahwa seiring berjalannya waktu pelanggan akan beralih sepenuhnya ke perusahaan pesaing yang menyediakan produk ataupun layanan yang sama. *Churn* harus segera ditangani agar perusahaan tidak merugi karena kehilangan pelanggan lama dan harus mengeluarkan biaya yang besar demi menarik pelanggan yang baru [10].

2.3 *Customer Relationship Management (CRM)*

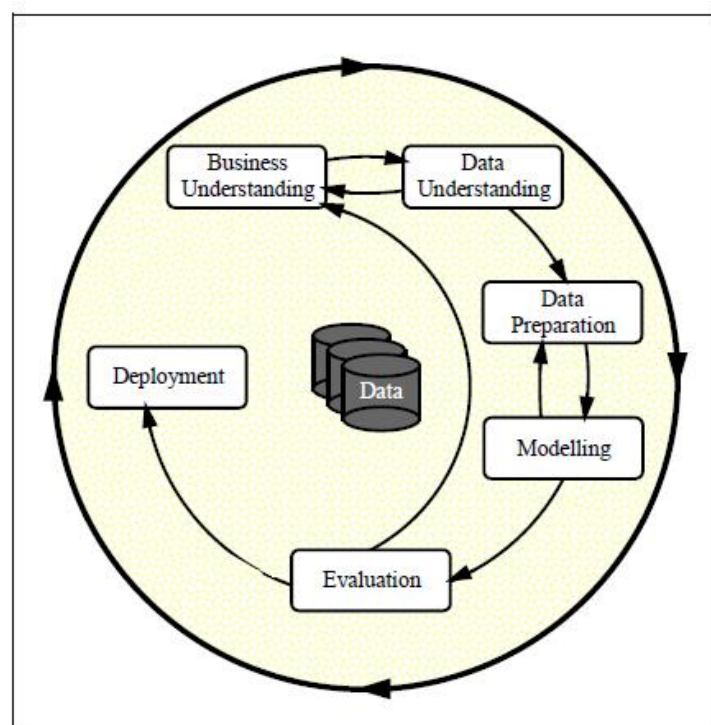
Customer Relationship Management (CRM) merupakan strategi bisnis dalam merencanakan, menjadwalkan dan mengendalikan penjualan dengan tujuan meningkatkan pendapatan, keuntungan, dan kepuasan pelanggan dengan mengatur segmentasi pelanggan, mendorong perilaku yang dapat memuaskan pelanggan, dan menerapkan budaya bisnis yang berfokus pada pelanggan [11]. Tujuannya adalah untuk meningkatkan loyalitas pelanggan dan meningkatkan pertumbuhan jangka panjang dan profitabilitas melalui pemahaman yang lebih mendalam mengenai perilaku dan karakteristik pelanggan sehingga dapat memberikan umpan balik yang lebih efektif dan meningkatkan integrasi untuk mengukur laba atas investasi atau yang dikenal dengan *Return on Investment (ROI)*.

Penting bagi perusahaan untuk memiliki hubungan yang baik dengan pelanggan. Perusahaan dapat mewujudkan hasil yang lebih baik ketika mengelola basis pelanggan untuk mengidentifikasi, memperoleh, memuaskan dan mempertahankan pelanggan yang menguntungkan. Mengelola retensi dan *tenure* pelanggan secara

bijak memiliki dua manfaat bagi perusahaan yaitu mengurangi biaya pemasaran dan memiliki wawasan pelanggan yang lebih baik. Saat hubungan semakin baik maka kepercayaan dan komitmen di antara perusahaan dan pelanggan cenderung tumbuh.

2.4 CRISP-DM

Cross Industry Standard Process for Data Mining (CRISP-DM) merupakan salah satu standar yang umum digunakan dalam melakukan proses data mining yang dipublikasikan pada tahun 1999. Metodologi CRISP-DM bertujuan untuk membuat proyek-proyek data mining yang besar, dapat diandalkan, dapat diulang, mudah dikelola dan lebih cepat [12]. Terdapat 6 fase pengembangan dalam CRISP-DM yaitu *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation*, dan *Deployment*.



Gambar 2.1 Fase model referensi CRISP-DM [12]

Tahapan dari CRISP-DM ini dimulai dari tahapan *business understanding* dengan mengandalkan data yang dimiliki untuk memahami tujuan dan persyaratan proyek dari perspektif bisnis. Selanjutnya *data understanding*, tahapan ini dilakukan dengan mengumpulkan dan memahami data, mengidentifikasi kualitas data,

menemukan wawasan awal atau bagian menarik dalam data yang dapat digunakan dalam membentuk hipotesis. Pada tahapan ini dapat kembali ke tahapan sebelumnya yaitu *business understanding* untuk memastikan data – data yang dikumpulkan dapat digunakan dalam mencapai tujuan bisnis. Selanjutnya tahapan *data preparation*, di dalamnya dilakukan proses untuk mempersiapkan data – data yang terkumpul dengan cara memilih data, membersihkan data, membuat atribut baru dari data yang sudah ada, mengintegrasikan data dan memformat data. Selanjutnya tahapan *modelling*, pada tahapan ini berbagai teknik pemodelan dipilih dan diterapkan serta parameter yang digunakan dikalibrasi ke nilai optimal. Pada tahapan ini apabila terdapat masalah atau atribut data yang kurang memenuhi dalam pemodelan dapat kembali ke tahapan *data preparation* untuk memperbaiki data yang digunakan. Selanjutnya tahapan *evaluation*, setelah membangun model dilakukan pengevaluasian secara menyeluruh dan meninjau langkah – langkah yang dilakukan untuk memastikan model mencapai tujuan bisnis dengan benar dan memiliki performa yang baik. Pada tahapan *evaluation* terdapat siklus balik ke tahapan *business understanding* sebab tidak semua kasus harus dilakukan tahapan *deployment* dimana terdapat kasus setelah melakukan tahapan evaluasi akan kembali ke tahapan *business understanding* bila terdapat beberapa masalah bisnis yang belum dipertimbangkan secara memadai.

Berikut adalah penjelasan setiap tahapannya.

1) *Business understanding* (pemahaman bisnis)

Tahap pemahaman bisnis merupakan tahap awal yang berfokus pada pemahaman tujuan dan persyaratan proyek dari perspektif bisnis, kemudian mengubah pengetahuan menjadi masalah *data mining* dan rencana proyek awal yang dirancang untuk mencapai tujuan.

2) *Data understanding* (pemahaman data)

Tahap pemahaman data dimulai dari pengumpulan data awal dan dilanjutkan dengan kegiatan membiasakan diri dengan data, mengidentifikasi masalah kualitas data, menemukan wawasan ke dalam data atau mendeteksi *subset* yang menarik untuk membentuk hipotesis informasi yang tersembunyi.

3) *Data preparation* (persiapan data)

Tahap persiapan data mencakup kegiatan untuk membangun dataset akhir yang akan dimasukkan ke dalam pemodelan dari data mentah. Kegiatan yang dilakukan kemungkinan besar akan dilakukan beberapa kali dan tidak berurut. Kegiatan yang dilakukan yaitu pemilihan atribut, pembersihan data, konstruksi atribut baru, dan transformasi data untuk pemodelan.

4) *Modelling* (pemodelan)

Tahap pemodelan terdiri dari kegiatan pemilihan dan penerapan teknik pemodelan. Untuk membangun model, parameter-parameter spesifik harus ditetapkan.

5) *Evaluation* (evaluasi)

Tahap evaluasi dilakukan dengan mengevaluasi model secara menyeluruh dan meninjau langkah – langkah yang dilakukan untuk membangun model, serta memastikan bahwa model tersebut dapat mencapai tujuan bisnis dengan baik.

6) *Deployment* (penyebaran)

Tahap penyebaran dilakukan dengan menyajikan data sehingga mudah dipahami dan dapat digunakan oleh pengguna [13]

2.5 Python

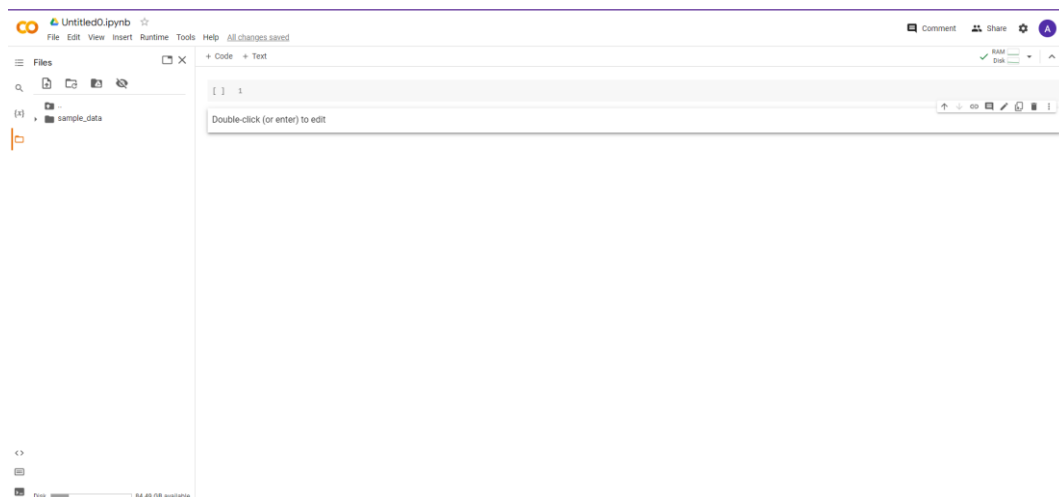
Python merupakan bahasa pemrograman tingkat tinggi yang umum digunakan dalam pemrograman, salah satunya dalam bidang analisis data. Python dikembangkan oleh Guido van Rossum yang diperkenalkan pertama kali pada tahun 1991. Sintaks python sederhana dan mudah dipelajari, serta memiliki banyak pustaka standar yang didistribusikan secara bebas. Adapun *library* yang digunakan dalam penelitian ini adalah sebagai berikut.

- 1) Pandas merupakan pustaka yang digunakan untuk mempermudah dalam menganalisis data, membersihkan data, memanipulasi data, serta mendukung operasi seperti penggabungan data, menghilangkan duplikasi, melakukan pengindeksan ulang, dan yang lainnya.
- 2) Numpy (*Numerical Python*) merupakan pustaka yang mendukung penggunaan array dan fungsi perhitungan seperti statistik, aljabar, matriks, dan yang lainnya.

- 3) Matplotlib merupakan pustaka yang digunakan dalam memvisualisasikan data, seperti menampilkan histogram, scatterplot, diagram lingkaran, grafik, dan yang lainnya.
- 4) Scikit-learn merupakan pustaka python yang mendukung penggunaan *machine learning* dengan berbagai algoritma baik *supervised learning* maupun *unsupervised learning* seperti *linear regression*, *classification*, *clustering* dan lainnya.

2.6 Google Colab

Google colab merupakan salah satu tool yang disediakan oleh Google berupa notebook yang dapat digunakan untuk menulis, mengeksekusi, serta menyimpan kode program python berbasis cloud secara gratis. Google colab memiliki fitur untuk menunjang projek *data science* maupun *machine learning*, di antaranya tidak diperlukan konfigurasi apapun, *library data science* yang telah terinstal, terdapat penyimpanan otomatis dan kontrol versi, dapat mengakses GPU maupun TPU, serta mudah dalam berbagi dan berkolaborasi.



Gambar 2.2 Tampilan Google Colab

2.7 *Data Integration*

Pada proses penambahan data sering kali melibatkan integrasi data dari berbagai penyimpanan data maupun dataset yang berbeda menjadi satu kesatuan yang lebih lengkap dan komprehensif. Tujuan utama dari integrasi data adalah menciptakan tampilan yang terpadu dan holistik dari data yang sebelumnya terpisah, sehingga memungkinkan analisis dan pemahaman yang lebih baik mengenai informasi yang terkandung dalam data tersebut. Integrasi yang dilakukan dengan cermat memiliki manfaat besar, seperti mengurangi redundansi dan inkonsistensi dalam kumpulan data yang dihasilkan. Hal tersebut dapat membantu meningkatkan akurasi dan kecepatan proses penambahan data. Namun, heterogenitas semantik dan struktur data menimbulkan tantangan besar dalam integrasi data. Terdapat beberapa hal yang perlu diperhatikan seperti esensi dari masalah identifikasi entitas, atribut yang berkorelasi, maupun data yang terduplikasi [14].

2.8 *Cleaning Data*

Data yang didapatkan umumnya tidak lengkap, sering kali mengandung nilai yang hilang dan tidak konsisten sehingga perlu dilakukan pembersihan data. Pembersihan data merupakan proses mengidentifikasi dan mengoreksi kesalahan, ketidakkonsistenan, dan ketidakakuratan dalam kumpulan data untuk memastikan bahwa data tersebut akurat, dapat diandalkan, dan sesuai untuk analisis dan penggunaan lebih lanjut. Hal yang dilakukan dalam pembersihan data di antaranya mengisi nilai yang hilang dan memperbaiki ketidakkonsistenan dalam data.

2.8.1 *Handling Missing Values*

Salah satu metode dasar dalam pembersihan data adalah menangani nilai – nilai yang hilang atau yang disebut dengan *missing values*. Terdapat beberapa cara yang dapat dilakukan dalam menangani nilai – nilai yang hilang yaitu [14]:

- 1) Penanganan dengan cara mengabaikan tuple tersebut. Metode ini tidak terlalu efektif bila persentase nilai yang hilang per atribut sangat bervariasi, kecuali jika tuple berisi beberapa atribut dengan nilai yang hilang.

- 2) Penanganan dengan cara mengisi nilai yang hilang secara manual. Cara ini memakan waktu yang lama dan mungkin tidak dapat dilakukan pada dataset yang besar dengan banyak nilai yang hilang.
- 3) Penanganan dengan cara menggunakan konstanta global untuk mengisi nilai yang hilang. Contohnya nilai yang hilang diganti dengan ‘*unknown*’, maka program data mining mungkin salah mengira bahwa nilai tersebut membentuk sebuah konsep yang menarik karena memiliki nilai yang sama yaitu ‘*unknown*’. Oleh karena itu, walaupun cara ini sederhana, cara ini juga tidak mudah.
- 4) Penanganan dengan cara menggunakan nilai mean atau median untuk mengisi nilai yang hilang. Untuk distribusi data yang normal menggunakan nilai mean, sedangkan untuk distribusi data miring menggunakan nilai median.
- 5) Penanganan dengan cara menggunakan mean atau median untuk semua sampel yang termasuk dalam kelas yang sama. Contohnya dalam mengelompokkan pelanggan menurut risiko kredit, maka dapat ditangani dengan mengganti nilai yang hilang dengan nilai pendapatan rata – rata untuk pelanggan yang sama dengan kategori risiko kredit yang diberikan.
- 6) Penanganan dengan cara menggunakan nilai yang paling mungkin untuk mengisi nilai yang hilang.

2.8.2 *Formatting Data*

Salah satu bentuk pembersihan data yang paling umum adalah mengubah data yang tidak terbaca atau sulit dibaca menjadi format yang konsisten dan dapat dibaca dengan mudah [15]. Memastikan bahwa tanggal dan waktu dalam format yang konsisten dan tipe data yang benar untuk memfasilitasi analisis dan perbandingan kronologis. Contohnya format tanggal yang berbeda pada beberapa negara, ada yang menggunakan bulan, hari, tahun (MM/DD/YY), ada pula yang menggunakan hari, bulan, tahun (DD/MM/YY). Hal tersebut perlu diperhatikan dan dipastikan bahwa variasi dalam pemformatan tidak diperlakukan sebagai entitas yang berbeda selama pemrosesan data.

2.8.3 *Removing Duplicates*

Menghilangkan data duplikat sangat penting untuk dilakukan untuk menjaga keakuratan data, terutama bila menggunakan beberapa dataset yang berisi data survei atau menggunakan data mentah yang mungkin berisi entri duplikat. Untuk dataset menyertakan pengidentifikasi unik, maka dapat menggunakannya untuk memverifikasi bahwa data yang digunakan secara tidak sengaja terdapat data duplikat. Sedangkan untuk dataset yang tidak memiliki sistem pengindeksan dapat menggunakan metode yang efektif dalam mengidentifikasi setiap entri unik [15].

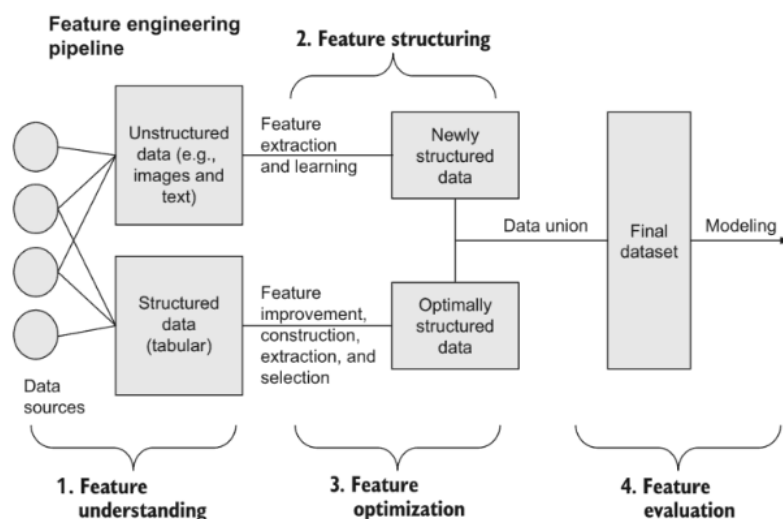
2.9 *Feature Engineering*

Fitur merupakan representasi numerik dari data mentah. Terdapat banyak cara untuk mengubah data mentah menjadi pengukuran numerik. Fitur harus berasal dari jenis data yang tersedia. Fitur yang tepat yaitu fitur yang relevan dengan tugas yang sedang dikerjakan dan harus mudah dicerna oleh model. *Feature engineering* (rekayasa fitur) merupakan proses merumuskan fitur yang paling sesuai menggunakan pengetahuan domain dari data [16]. Agar algoritma pembelajaran prediktif/klasifikasi dapat bekerja untuk masalah yang mendasarinya, *input* harus ditransformasikan ke dalam format tertentu sehingga algoritma dapat memahaminya dan mengklasifikasikan kelas yang akurat dimana suatu entitas berada dan prediksi di masa depan. Proses ini biasanya memakan waktu paling lama dalam membangun model klasifikasi [17]. Bila tidak ada cukup banyak fitur yang informatif atau terlalu banyak fitur serta tidak relevan, maka model tidak akan dapat melakukan tugas utama bahkan dapat menjadikan model sulit untuk dilatih yang dapat berdampak pada kinerja model.

Terdapat lima tipe dari *feature engineering*, diantaranya sebagai berikut [18].

- 1) *Feature Improvement* (peningkatan fitur) yaitu membuat fitur yang ada lebih bermanfaat melalui transformasi matematis. Contohnya mengisi nilai yang hilang pada kolom suhu pada dataset cuaca dengan menyimpulkannya dari kolom lain.

- 2) *Feature Construction* (konstruksi fitur) yaitu menambah dataset dengan membuat fitur baru yang dapat diinterpretasikan dari fitur yang dapat diinterpretasikan yang ada. Contohnya membagi total harga fitur rumah dengan luas fitur rumah untuk membuat fitur harga per kaki persegi dalam dataset penilaian rumah.
- 3) *Feature Selection* (pemilihan fitur) yaitu memilih subset fitur terbaik dari sekumpulan fitur yang ada. Contohnya adalah setelah membuat fitur harga per kaki persegi, mungkin menghapus dua fitur sebelumnya jika mereka tidak menambah nilai pada model *machine learning*.
- 4) *Feature Extraction* (ekstraksi fitur) yaitu mengandalkan algoritma untuk secara otomatis membuat fitur baru yang terkadang tidak dapat ditafsirkan. Biasanya didasarkan pada pembuatan asumsi parametrik tentang data. Contohnya mengandalkan model pembelajaran transfer yang sudah terlatih untuk memetakan teks yang tidak terstruktur ke ruang vektor yang terstruktur dan umumnya tidak dapat ditafsirkan.
- 5) *Feature Learning* (pembelajaran fitur) yaitu menghasilkan serangkaian fitur baru secara otomatis, umumnya dengan mengekstraksi struktur dan mempelajari representasi dari data mentah yang tidak terstruktur, seperti video, teks, dan gambar.



Gambar 2.3 Penerapan *Feature Engineering* [18]

2.10 Data Transformation

Transformasi data merupakan proses mengubah atau mengkonsolidasikan data – data ke dalam bentuk yang sesuai sehingga proses penambangan yang dihasilkan lebih efisien dan mudah dalam memahami pola yang ditemukan [14]. Strategi yang dilakukan untuk transformasi data di antaranya sebagai berikut.

2.10.1 Normalisasi Data

Normalisasi data merupakan salah satu teknik transformasi data yang mengubah skala data atribut sehingga berada dalam rentang yang lebih kecil, seperti -1 hingga 1 atau 0 hingga 1 [14]. Untuk menghindari ketergantungan pada pilihan unit pengukuran, maka data harus dinormalisasikan. Menormalkan data mencoba memberikan bobot yang sama pada semua atribut. Dengan menormalkan nilai input untuk setiap atribut yang diukur pada data pelatihan dapat membantu mempercepat fase pembelajaran model. Salah satu metode untuk normalisasi data adalah menggunakan *min-max normalization* (normalisasi min-max). Normalisasi min-max melakukan transformasi linear pada data asli. Misalkan \min_A dan \max_A merupakan nilai minimum dan maksimum dari sebuah atribut A. Normalisasi min-max memetakan sebuah nilai v_i dari atribut A ke v'_i dalam rentang $[\text{new_min}_A, \text{new_max}_A]$ dengan cara komputasi sebagai berikut.

$$v'_i = \frac{v_i - \min_A}{\max_A - \min_A} (\text{new_max}_A - \text{new_min}_A) + \text{new_min}_A \quad (2.1)$$

Keterangan variabel :

v'_i	: nilai atribut setelah dinormalisasi
v_i	: nilai atribut saat ini
\max_A	: maksimum nilai atribut A
\min_A	: minimum nilai atribut A
$\text{new_min}_A, \text{new_max}_A$: rentang minimum dan maksimum yang baru (0,1 atau -1,1)

2.10.2 One Hot Encoding

One hot encoding merupakan teknik transformasi data yang digunakan yang mengubah variabel kategorikal menjadi bentuk yang dapat dimengerti oleh algoritma pembelajaran mesin. Setiap bit mewakili kategori yang mungkin. Jika variabel tidak dapat menjadi bagian dari beberapa kategori sekaligus, maka hanya satu bit dalam kelompok tersebut yang menjadi aktif [16]. Berikut contoh penggunaan *one hot encoding* ditunjukkan pada gambar 2.4.

	e_1	e_2	e_3
San Francisco	1	0	0
New York	0	1	0
Seattle	0	0	1

Gambar 2.4 Penerapan *one hot encoding* pada kategori dari ketiga kota [16]

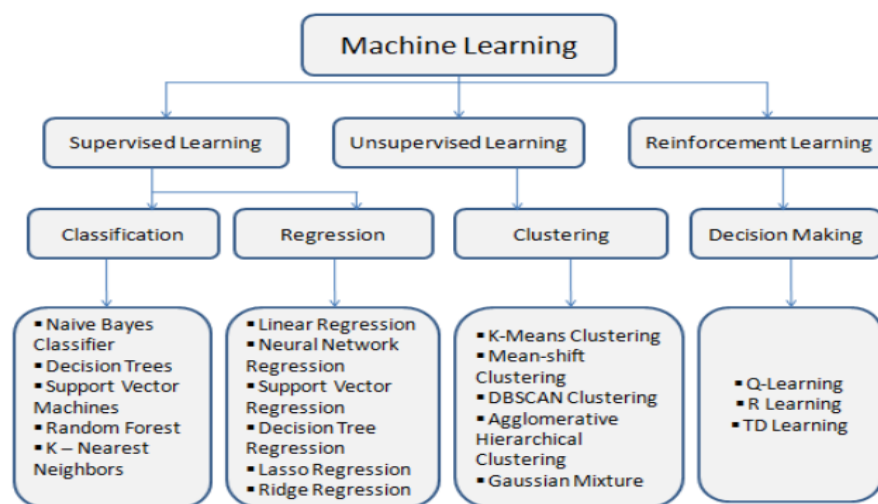
2.11 Machine learning

Menurut Batta Mahesh, *machine learning* merupakan pembelajaran mesin yang digunakan untuk mengajari mesin cara menangani data dengan lebih efisien [19]. *Machine learning* mengandalkan algoritma yang berbeda untuk memecahkan masalah data dimana jenis algoritma yang digunakan tergantung pada jenis masalah yang ingin diselesaikan, jumlah variabel, jenis model yang paling sesuai, dan sebagainya. *Machine learning* secara umum dikategorikan menjadi 3 yaitu [20] :

- 1) *Supervised Learning* (pembelajaran yang diawasi) merupakan pembelajaran mesin yang bekerja dengan kumpulan data berlabel dimana berfokus pada pola pembelajaran dengan menghubungkan antara variabel dan hasil yang diketahui. *Supervised learning* bekerja dengan memberikan data sampel mesin dengan berbagai fitur (direpresentasikan sebagai X) dan *output* nilai data yang benar (direpresentasikan sebagai y). Nilai keluaran dan fitur yang diketahui menjadikan dataset tersebut sebagai berlabel. Algoritma kemudian menguraikan pola – pola yang ada dalam data dan membuat model yang dapat mereproduksi aturan dasar yang sama dengan data baru. Pembelajaran diawasi diklasifikasikan ke dalam dua kategori algoritma yaitu untuk memecahkan masalah klasifikasi ketika variabel keluarannya berupa

kategori dan masalah regresi ketika variabel outputnya berupa nilai riil. Contoh algoritma *supervised learning* yaitu *Logistic Regression*, *decision tree*, *Random Forest*, *XGBoost*, *k-nearest neighbors*, *neural networks*, dan *support vector machine*.

- 2) *Unsupervised Learning* (pembelajaran tanpa pengawasan) merupakan pembelajaran mesin yang tidak semua variabel dan pola data diklasifikasikan sehingga mesin harus mengungkap pola tersembunyi dan membuat label melalui penggunaan algoritma pembelajaran tanpa pengawasan. Pembelajaran tanpa pengawasan diklasifikasikan ke dalam dua kategori algoritma yaitu untuk memecahkan masalah pengelompokan untuk mengelompokkan data ke dalam kategori – kategori dan masalah pembelajaran aturan asosiasi yang melibatkan ringkasan distribusi data. Contoh algoritma *unsupervised learning* adalah *K-Means Clustering*.
- 3) *Reinforcement Learning* merupakan pembelajaran mesin yang terus meningkatkan modelnya dengan memanfaatkan umpan balik dari iterasi sebelumnya. Pada *reinforcement learning*, algoritma ditetapkan untuk melatih model melalui pembelajaran berkelanjutan. Model yang dihasilkan memiliki kriteria kinerja yang terukur dimana *outputnya* tidak diberikan label, melainkan dinilai.



Gambar 2.5 Bagan *Machine Learning* [21]

2.12 Klasifikasi

Terdapat dua metode analisis data yang dapat digunakan untuk mendapatkan model yang menggambarkan kategori – kategori tertentu atau untuk memprediksi tren data di masa yang akan datang yaitu klasifikasi dan prediksi numerik. Klasifikasi merupakan teknik menganalisis data yang mengekstrak pola yang menjelaskan kelompok data yang penting. Pola – pola ini disebut *classifiers* yang memprediksi label kelas yang bersifat kategoris (diskrit, tidak terorganisir) [14]. Sedangkan prediksi numerik merupakan jenis pembelajaran klasifikasi dimana hasilnya berupa nilai numerik, bukan kategori [22]. Dalam implementasinya, klasifikasi digunakan dalam membangun model untuk memprediksi kelas label, seperti *churn* dan tidak *churn*. Sedangkan prediksi numerik digunakan untuk memprediksi fungsi yang bernilai kontinu, seperti berapa banyak pelanggan yang akan sering melakukan transaksi.

Cara kerja klasifikasi melibatkan dua tahap, yaitu tahap pelatihan dimana model klasifikasi dibuat dan tahapan klasifikasi dimana model tersebut digunakan untuk memprediksi label kelas untuk data yang diberikan. Ketepatan model klasifikasi pada set data uji tertentu adalah proporsi tupel dalam set uji yang diidentifikasi dengan benar oleh pengklasifikasi. Kelas yang diprediksi oleh pengklasifikasi untuk setiap tupel uji dibandingkan dengan label kelas aktual dalam set data yang telah dipelajari.

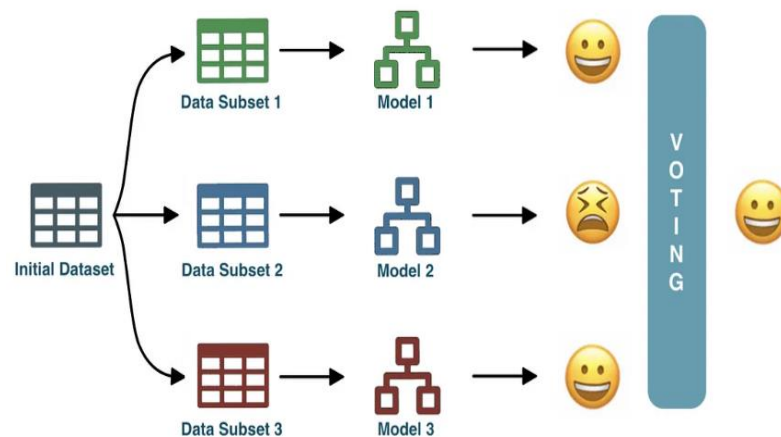
2.13 Ensemble Learning

Ensemble learning merupakan teknik yang dilakukan untuk melatih beberapa pelajar untuk mengatasi masalah yang sama. Teknik ansambel ini bertujuan untuk membangun sekelompok pelajar yang dilatih dan menggabungkannya. Sehingga ensemble learning ini disebut juga sebagai pembelajaran berbasis kelompok atau sistem pengklasifikasian ganda [23]. Sebuah ansambel berisi sejumlah peserta yang dilatih yang disebut *base learners*. Mayoritas teknik ansambel menggunakan algoritma pembelajaran dasar tunggal untuk menghasilkan pelajar dasar yang seragam, yang memiliki jenis yang sama sehingga menghasilkan ansambel yang

homogen. Namun, beberapa teknik menggunakan berbagai algoritma pembelajaran untuk menciptakan beragam pelajar, yang memiliki jenis yang berbeda sehingga menghasilkan ansambel yang heterogen.

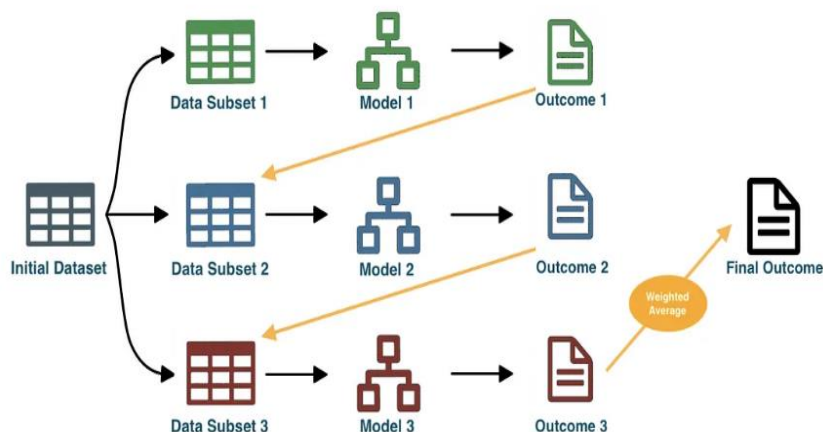
Terdapat beberapa teknik dalam metode pembelajaran ansambel, diantaranya adalah

- 1) Teknik Bagging (*Bootstrap Aggregating*) merupakan teknik penggabungan algoritma dasar yang homogen menggunakan kumpulan data yang diambil secara acak dan menggabungkan prediksi untuk merancang model terpadu berdasarkan proses pemungutan suara diantara data pelatihan. Contoh algoritma populer dari bagging adalah *Random Forest* [20].



Gambar 2.6 Metode bagging menggunakan suara mayoritas [24]

- 2) Teknik Boosting merupakan teknik pelatihan pembelajar dasar yang homogen secara berulang-ulang untuk meminimalisir kesalahan dari sebelumnya. Teknik ini membuat model baru untuk mengurangi kesalahan model sebelumnya. Masing-masing model ini disebut *weak learner* dengan hasil akhir dari pembelajaran dibentuk *strong learner* dengan mendapatkan rata – rata tertimbang dari semua *weak learner*. Contoh algoritma populer dari boosting adalah *AdaBoost*, *Gradient Boosting*, dan *XGBoost*.



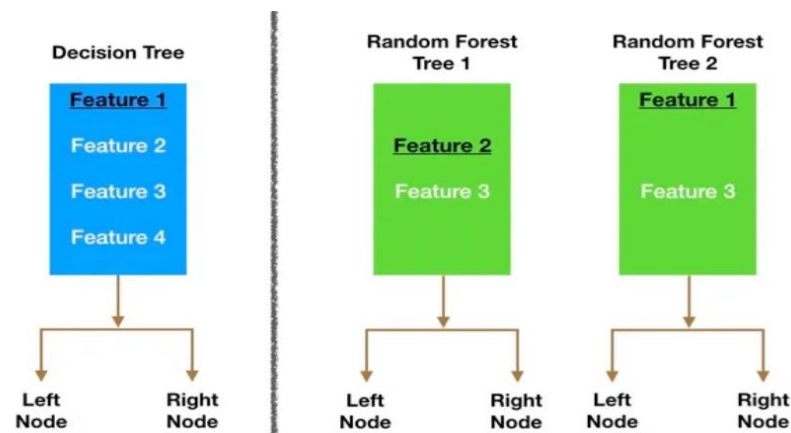
Gambar 2.7 Metode boosting menggunakan strategi rata-rata tertimbang[24]

2.13.1 *Random Forest*

Random Forest adalah algoritma klasifikasi yang terdiri dari banyak pohon keputusan. Algoritma ini menggunakan bagging dan *random feature* saat membangun setiap pohon individu yang mencoba membuat hutan pepohonan yang tidak berkorelasi [25]. Menurut Breiman dalam [23], *Random Forest* merupakan representatif dari metode ansambel dimana selama pembangunan pohon keputusan komponen pada setiap langkah pemilihan terpisah, *Random Forest* memilih secara acak subset fitur terlebih dahulu, kemudian melakukan prosedur pemilihan split konvensional dalam subset fitur yang dipilih.

Pohon keputusan yang digunakan sebagai dasar pembelajaran dalam *Random Forest* sangat sensitif terhadap data latih, dimana perubahan kecil pada data latih dapat menghasilkan struktur pohon yang berbeda. *Random Forest* menggunakan hal ini dengan mengizinkan setiap pohon secara acak mengambil sampel dari kumpulan data secara bergantian dan menghasilkan pohon yang berbeda. Dalam proses pembagian *node*, pohon keputusan membagi sebuah *node* dengan mempertimbangkan setiap fitur yang mungkin dan memilih salah satu yang menghasilkan paling banyak antara pengamatan *node* kiri dan *node* kanan. Sedangkan pada *Random Forest* menerapkan *random feature*, dimana setiap pohon hanya dapat memilih dari *subset* fitur secara acak sehingga menyebabkan lebih

banyak variasi diantara pohon – pohon dalam model. Pada akhirnya menghasilkan korelasi yang lebih rendah di antara pohon – pohon dan lebih banyak diversifikasi.



Gambar 2.8 Pemisahan *node* dalam *Random Forest* berdasarkan *subset* acak dari fitur untuk setiap pohon [25]

Berikut terdapat langkah-langkah dalam menyusun dan memprediksi dengan menggunakan metode *Random Forest* :

1) Tahapan *Bootstrapping*

Langkah awal dalam memulai pembangunan *Random Forest* adalah mengambil sampel secara acak dengan ukuran n dari himpunan data asli dengan pengembalian.

2) Tahapan *Random Feature Selection*

Pada langkah ini, pohon akan dibangun tanpa pemangkasan hingga mencapai ukuran maksimum. Selama proses pemilahan variabel prediktor, variabel prediktor m dipilih secara acak, dengan m yang jauh lebih kecil dari p . Kemudian, pemilah terbaik dipilih berdasarkan m prediktor.

Pada tahap awal penentuan pohon keputusan, langkah yang diambil adalah melakukan perhitungan nilai *entropy* untuk mengukur tingkat keberagaman atribut dan nilai *information gain*. Perhitungan nilai *entropy* dapat dilakukan dengan menggunakan rumus persamaan (2.1) untuk satu atribut, persamaan (2.2) untuk dua atribut menggunakan tabel frekuensi, dan menentukan nilai *information gain* menggunakan persamaan (2.3).

$$\text{Entropy}(S) = \sum_{i=1}^c -p_i \log_2 p_i \quad (2.2)$$

Keterangan:

- S = himpunan dataset
- C = jumlah kelas
- p_i = probabilitas frekuensi kelas ke-I dalam dataset

$$\text{Entropy}(T, X) = \sum_{c \in X} p(c) E(c) \quad (2.3)$$

Keterangan:

- (T,X) = atribut T dan atribut X
- P(c) = probabilitas kelas atribut
- E(c) = nilai entropy kelas atribut

$$\text{Gain}(A) = \text{Entropy}(S) - \sum_{i=1}^k \frac{|S_i|}{|S|} \times \text{Entropy}(S_i) \quad (2.4)$$

Keterangan:

- S = himpunan data set
- A = atribut
- $|S_i|$ = jumlah sampel untuk nilai i
- $|S|$ = jumlah seluruh sampel data
- $\text{Entropy}(S_i)$ = entropy untuk sampel yang memiliki nilai i

Adapun parameter yang digunakan pada algoritma *Random Forest* untuk tuning hyperparameter dapat dilihat pada tabel 2.1 berikut.

Tabel 2.1 Parameter pada algoritma *Random Forest* [26]

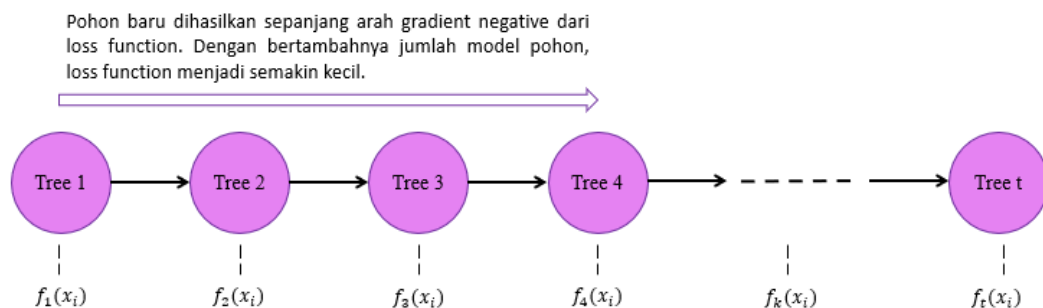
Parameter	Keterangan
<i>n_estimator</i>	jumlah <i>tree</i> di dalam <i>forest</i>
<i>max_features</i>	jumlah maksimum fitur yang dipertimbangkan untuk memisahkan sebuah <i>node</i>
<i>max_depth</i>	jumlah maksimum level dalam setiap pohon keputusan
<i>min_samples_split</i>	jumlah minimum titik data yang ditempatkan dalam sebuah <i>node</i> sebelum <i>node</i> tersebut dipecah
<i>min_samples_leaf</i>	jumlah minimum titik data yang diperbolehkan dalam simpul daun
<i>bootstrap</i>	rasio <i>subsample</i> kolom pada setiap tingkatan
<i>criterion</i>	Metrik yang digunakan dalam proses pengambilan keputusan saat membangun setiap pohon dalam hutan acak (<i>gini</i> , <i>entropy</i>)

2.13.2 XGBoost

XGBoost (*eXtreme Gradient Boosting*) adalah jenis pembelajaran mesin yang berada di bawah kategori pembelajaran ansambel dan pohon peningkatan gradien. *XGBoost* diperkenalkan oleh Tianqi Chen dan Carlos Guestrin pada tahun 2014 dan saat ini menjadi salah satu algoritma yang paling banyak digunakan dan efisien untuk menyelesaikan masalah klasifikasi, regresi, dan ranking [27]. Dalam kebanyakan implementasinya, *XGBoost* menggunakan *Decision Tree* sebagai base learner yang disebut dengan *Classification and Decision Tree* (CART). Dimana *classification tree* sebagai variabel target bersifat kategorikal dan pohon digunakan untuk mengidentifikasi kelas variabel targetnya, serta *regression tree* sebagai variabel target bersifat kontinu dan pohon digunakan untuk memprediksi nilainya [24].

Pada dasarnya, *XGBoost* berfungsi dengan menggabungkan beberapa pohon keputusan yang lemah (*weak learner*) untuk membentuk model yang kuat (*strong learner*). Kemudian digabungkan untuk membuat model yang lebih baik. Setiap pohon keputusan di *XGBoost* memiliki kedalaman yang rendah dan menggunakan sebagian fitur yang ada di dataset. Dalam *XGBoost*, proses pembangunan model dilakukan secara iteratif dengan mengoptimalkan setiap pohon keputusan yang dibangun. Pohon keputusan berikutnya yang dibangun pada iterasi berikutnya akan menggunakan informasi yang dihasilkan oleh pohon keputusan dari iterasi sebelumnya. Hal ini dikenal dengan istilah *gradient boosting*, yang didesain untuk mengurangi nilai *loss function*. *Loss function* mengukur seberapa jauh prediksi yang didapatkan dari hasil aktual titik data tertentu. Semakin baik prediksinya maka semakin rendah juga *output* dari *loss function*.

Berikut merupakan proses komputasi dari algoritma *XGBoost* ditunjukkan pada Gambar 2.5 berikut.



Gambar 2.9 Diagram skema algoritma *XGBoost*

Nilai prediksi pada langkah t dinyatakan dengan $\hat{y}^{(t)}$ dimana:

$$\hat{y}^{(t)} = \sum_{k=1}^t f_k(x_i) \quad (2.5)$$

$f_k(x_i)$ menyatakan model pohon. Untuk y_i diperoleh dari perhitungan berikut.

$$\hat{y}_i^{(0)} = 0$$

$$\hat{y}_i^{(1)} = f_1(x_1) = \hat{y}_i^{(0)} + f_1(x_1)$$

$$\hat{y}_i^{(2)} = f_1(x_1) + f_2(x_2) = \hat{y}_i^{(1)} + f_2(x_2)$$

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i)$$

$$\hat{y}^{(t)} = \sum_{k=1}^t f_k(x_i)$$

Dimana :

- $\hat{y}_i^{(t)}$ = final *tree* model
- $\hat{y}_i^{(t-1)}$ = model pohon yang dihasilkan sebelumnya
- $f_t(x_i)$ = model baru yang dibangun
- t = jumlah total model dari *base tree*

Penting bagi algoritma *XGBoost* dalam menentukan jumlah pohon dan *depth*. Kesulitan dalam mencari algoritma yang optimal dapat diatasi dengan mencari pendekatan klasifikasi baru yang dapat mengurangi *loss function*, berdasarkan persamaan berikut.

$$\text{Obj}^{(t)} = \sum_{i=1}^t l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^t \Omega(f_i) \quad (2.6)$$

Dimana:

- $\hat{y}_i^{(t)}$ = nilai prediksi
- y_i = nilai aktual
- $l(y_i, \hat{y}_i^{(t)})$ = *loss function*
- $\Omega(f_i)$ = istilah regulasi

XGBoost adalah sebuah algoritma yang memiliki banyak karakteristik yang membuatnya menjadi alat yang sangat efisien, yang meliputi:

- Regularisasi dimana *XGBoost* menawarkan berbagai teknik regularisasi untuk meminimalkan kemungkinan *overfitting*.
- *Loss function* yang dapat beradaptasi dimana *XGBoost* menyajikan beberapa pilihan fungsi *loss* yang dapat disesuaikan dengan masalah spesifik yang akan diselesaikan.

- Skalabilitas dimana *XGBoost* mampu menangani kumpulan data yang sangat besar, dan proses pelatihan model yang cepat.
- Manajemen *missing value* dimana *XGBoost* dapat menangani data yang memiliki nilai yang hilang dengan mengisolasinya di cabang pohon keputusan yang berbeda.

Adapun parameter yang digunakan pada algoritma *XGBoost* untuk tuning hyperparameter dapat dilihat pada tabel 2.2 berikut.

Tabel 2.2 Parameter pada algoritma *XGBoost* [28]

Parameter	Keterangan
<i>max_depth</i>	kedalaman maksimum per <i>tree</i>
<i>min_child_weight</i>	jumlah minimum bobot <i>instance</i> yang dibutuhkan <i>child</i>
<i>max_delta_step</i>	langkah delta maksimum yang diizinkan untuk setiap <i>output leaf</i>
<i>eta (learning_rate)</i>	penyusutan ukuran yang digunakan untuk mencegah <i>overfitting</i>
<i>n_estimators</i>	jumlah pohon pada <i>tree</i>
<i>colsample_bytree</i>	rasio <i>subsample</i> kolom pada setiap tingkatan
<i>subsample</i>	jumlah sampel yang digunakan saat proses pelatihan sebelum membangun <i>tree</i>
<i>sampling_method</i>	metode yang digunakan untuk mengambil sampel pelatihan
<i>alpha (reg_alpha)</i>	L1 regulasi pada bobot
<i>lambda (reg_lambda)</i>	L2 regulasi pada bobot
<i>gamma (min_split_loss)</i>	nilai minimum <i>loss reduction</i>

2.14 *Logistic Regression*

Logistic Regression merupakan teknik yang biasanya digunakan untuk klasifikasi biner yang menerapkan fungsi sigmoid dalam menganalisis data dan memprediksi dua kelas diskrit yang ada di dalam kumpulan data [20]. Fungsi sigmoid menghasilkan kurva berbentuk S yang dapat mengubah angka apapun dan memetakannya menjadi nilai numerik antara 0 dan 1 walaupun tanpa pernah mencapai batas – batas yang tepat. Oleh karena itu, *Logistic Regression* dapat diterapkan untuk mengklasifikasikan *customer churn* dan *customer tidak churn*.

Terdapat tiga tipe *Logistic Regression*, yaitu:

- 1) *Binary Logistic Regression* merupakan tipe *Logistic Regression* yang hanya mengklasifikasikan data ke dalam dua kelas saja. Contohnya mengklasifikasikan pelanggan *churn* atau tidak *churn*.
- 2) *Multinomial Logistic Regression* merupakan tipe *Logistic Regression* yang mengklasifikasikan data ke dalam dua kelas atau lebih. Contohnya pada kasus analisis sentimen yang mengklasifikasikan kalimat positif, negatif, atau netral.
- 3) *Ordinal Logistic Regression* merupakan tipe *Logistic Regression* yang mengklasifikasikan data ke dalam dua kelas atau lebih dengan memperhatikan urutannya. Contohnya pada kasus pembagian kelas siswa berdasarkan rentang nilainya.

Logistic Function merupakan fungsi yang terbentuk dari menyamakan nilai Y pada fungsi linear dengan nilai Y pada fungsi sigmoid dengan tujuan untuk merepresentasikan data – data ke dalam bentuk fungsi sigmoid. Berikut langkah – langkah membentuk *logistic function* :

- a) Pertama, melakukan operasi *invers* pada fungsi sigmoid, sehingga sigmoid berubah bentuk menjadi $Y = \ln \left(\frac{p}{1-p} \right)$
- b) Kedua, menyetarakannya dengan fungsi linear $Y = b_0 + b_1 * X$ sehingga didapatkan persamaan $\ln \left(\frac{p}{1-p} \right) = b_0 + b_1 * X$

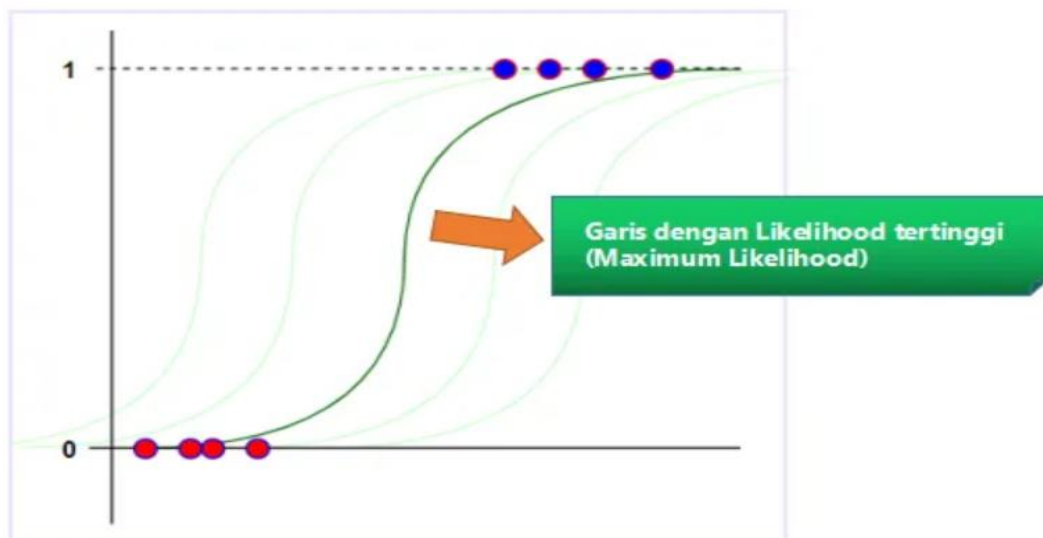
c) Ketiga, mengubah persamaan $\ln\left(\frac{p}{1-p}\right) = b_0 + b_1 * X$ sehingga didapatkan

$$\text{persamaan } P = \frac{1}{(1+e^{-(b_0+b_1*X)})}$$

Menentukan koefisien *logistic function* yaitu *maximum likelihood* dan *R-Squared*. *Maximum likelihood* merupakan cara dalam menentukan posisi sigmoid yang menjadi model terbaik yang dapat terbentuk dari data-data yang ada. Persamaan fungsi logistik dapat dilihat pada rumus 2.7 berikut.

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q x_q)}} \quad (2.7)$$

Dari persamaan diatas dapat diketahui bahwa yang memengaruhi posisi fungsi sigmoid adalah persamaan $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_q x_q$, yang berarti perubahan garis akan memengaruhi posisi dan nilai *likelihood* dari fungsi logistik. Garis dengan *likelihood* tertinggi menunjukkan bahwa garis yang menentukan posisi sigmoid mengklasifikasikan data dengan baik.



Gambar 2.10 *Maximum Likelihood*

R-Squared merupakan cara untuk mengetahui apakah fungsi logistik dengan nilai maksimum *likelihood* dapat menyatakan data dengan baik. Dimana terdapat parameter penting yang dibutuhkan dalam mencari nilai *R-Squared*, yaitu *Maximum Likelihood* dan *Badfit Likelihood*.

Formula *R-Squared* dapat dilihat pada rumus 2.8 berikut.

$$R - \text{Squared} = \frac{\text{Badfit Likelihood} - \text{Maximum Likelihood}}{\text{Badfit Likelihood}} \quad (2.8)$$

Adapun parameter yang digunakan pada algoritma *Logistic Regression* untuk tuning hyperparameter dapat dilihat pada tabel 2.3 berikut.

Tabel 2.3 Parameter pada algoritma *Logistic Regression* [29]

Parameter	Keterangan
<i>penalty</i>	jumlah <i>tree</i> di dalam <i>forest</i>
<i>c</i>	jumlah maksimum fitur yang dipertimbangkan untuk memisahkan sebuah <i>node</i>
<i>class_weight</i>	jumlah maksimum level dalam setiap pohon keputusan
<i>solver</i>	Algoritma yang akan digunakan dalam masalah optimasi
<i>max_iter</i>	jumlah iterasi maksimum yang diperlukan agar pemecah masalah dapat konvergen
<i>multi_class</i>	(auto, ovr, multinomial)

2.15 Tuning Parameter dengan *Grid Search CV*

Nilai – nilai hyperparameter algoritma pembelajaran mengatur prosedur pembelajaran dan memengaruhi parameter akhir model. Tujuan dari pengoptimalan hyperparameter adalah untuk menemukan nilai optimal untuk parameter ini sehingga memungkinkan untuk mencapai hasil yang baik dari data dengan cepat [30]. Modifikasi pada hyperparameter model pembelajaran mesin secara substansial dapat memengaruhi kinerja prediksi model.

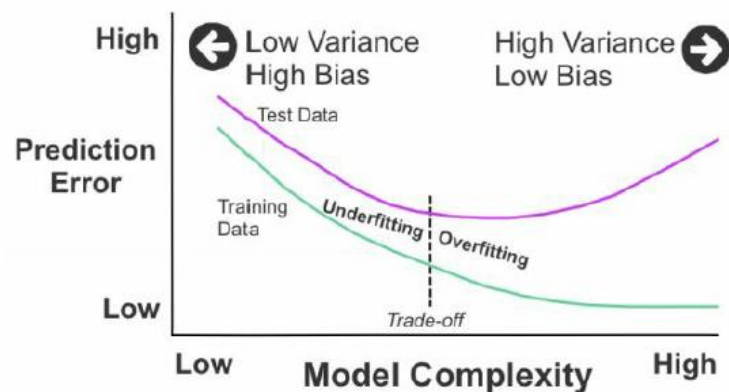
Grid search merupakan salah satu teknik untuk tuning hyperparameter yang dapat diimplementasikan. Proses *grid search* dimulai dengan menentukan kisi ruang pencarian yang terdiri dari nama hyperparameter yang dipilih dan nilai yang sesuai.

Melalui eksplorasi mendalam, *grid search* mengidentifikasi kombinasi optimal dari nilai yang diberikan [31]. Dalam implementasi codingnya menggunakan class *GridSearchCV* pada pustaka *scikit-learn*. Kelas *GridSearchCV* menyediakan alternatif untuk membagi set data menjadi set pelatihan dan pengujian dengan mengaktifkan opsi untuk melakukan validasi silang. Ini melibatkan *resampling* data menjadi beberapa *folds* dan menggunakan setiap *record* untuk tujuan pelatihan dan pengujian.

2.16 Bias dan Varians

Memilih algoritma yang sesuai merupakan aspek penting dalam membangun model peramalan yang tepat. Namun, bisa menjadi sulit untuk menyeimbangkan antara akurasi dan implementasi ketika memilih algoritma dengan tingkat keberhasilan yang tinggi. Tantangan yang selalu ada dalam *machine learning* adalah mengelola masalah *underfitting* dan *overfitting*, yang mengacu pada sejauh mana model yang dibuat mengikuti pola asli dataset.

Bias menunjukkan perbedaan antara nilai estimasi dan nilai faktual. Jika terdapat tingkat keberpihakan yang tinggi, maka estimasi mungkin akan terdistorsi ke arah tertentu yang menyimpang dari nilai faktual. Sedangkan varians menunjukkan seberapa tersebar nilai yang diperkirakan. Hasil yang buruk dapat diakibatkan oleh kesalahan dalam menangani *trade-off bias-variens*. Hal ini dapat menyebabkan model menjadi terlalu sederhana dan kaku (*underfitting*) atau terlalu rumit dan fleksibel (*overfitting*) [20].



Gambar 2.11 Model complexity based on prediction error [22]

- *Underfitting* memiliki varians yang rendah dan bias yang tinggi. *Underfitting* terjadi ketika model terlalu sederhana dan gagal menemukan pola yang mendasari dataset, sehingga mengakibatkan prediksi yang tidak tepat untuk data pelatihan dan pengujian. Data pelatihan yang tidak memadai untuk mencakup semua kombinasi potensial, serta contoh dimana data pelatihan dan pengujian tidak diacak dengan benar adalah penyebab umum dari *underfitting*.
- *Overfitting* memiliki varians yang tinggi dan bias yang rendah. Model yang *overfitting* akan memberikan perkiraan yang tepat dari data pelatihan, tetapi mungkin menghasilkan perkiraan yang kurang akurat dari data pengujian. *Overfitting* juga dapat terjadi jika data pelatihan dan pengujian tidak diacak sebelum dipisahkan, serta pola dalam data tidak terdistribusi secara merata di antara kedua segmen data.

2.17 Pengukuran Kinerja Algoritma Klasifikasi

Tahapan evaluasi model yang akan dibuat merupakan kunci untuk menentukan model mana yang paling baik untuk digunakan pada masalah pengklasifikasian dan mengevaluasi terkait bagaimana metode-metode yang berbeda bekerja dan membandingkan antara satu dengan yang lain.

2.17.1 *Classification Report and Confusion Matrix*

Confusion matrix merupakan alat yang digunakan untuk menganalisis seberapa baik pengklasifikasian dilakukan dalam mengenali tupel dari kelas yang berbeda. TP dan TN untuk pengklasifikasi menilai benar, sedangkan FP dan FN untuk pengklasifikasi melakukan kesalahan [14].

- *True Positive* (TP) yaitu tupel positif yang diberi label dengan benar oleh pengklasifikasi, sehingga TP menjadi jumlah positif yang benar.
- *True Negative* (TN) yaitu tupel negatif yang diberi label dengan benar oleh pengklasifikasi, sehingga TN menjadi jumlah negatif yang benar.
- *False Positive* (FP) yaitu tupel negatif yang diberi label yang salah sebagai positif, sehingga FP menjadi jumlah positif yang salah.

- *False Negative* (FN) yaitu tupel positif yang salah diberi label sebagai negatif, sehingga FN menjadi jumlah negatif yang salah.

		Predicted class		Total
		<i>yes</i>	<i>no</i>	
Actual class	<i>yes</i>	<i>TP</i>	<i>FN</i>	<i>P</i>
	<i>no</i>	<i>FP</i>	<i>TN</i>	<i>N</i>
Total		<i>P'</i>	<i>N'</i>	<i>P + N</i>

Gambar 2.12 *Confusion Matrix* [14]

Classification report merupakan sebuah laporan berupa ringkasan evaluasi kualitas pengklasifikasian dari model *machine learning* yang dibangun. Ukuran – ukuran yang dievaluasi yaitu *accuracy*, *precision*, *recall*, *F1-score*, dan *support*.

- *Accuracy* merujuk pada kemampuan prediksi pengklasifikasian. Akurasi dari sebuah pengklasifikasian pada data uji yang diberikan adalah persentase tupel data uji yang diklasifikasikan dengan benar oleh pengklasifikasi.

$$\text{accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{FN} + \text{FP} + \text{TN})} \quad (2.9)$$

- *Precision* merujuk sebagai ukuran ketepatan yang dihitung sehubungan dengan nilai prediksi yakni berapa persen tupel yang diberi label positif sebenarnya.

$$\text{precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \quad (2.10)$$

- *Recall* merujuk pada ukuran kelengkapan yang dihitung sehubungan dengan nilai aktual dalam dataset yakni berapa persen tupel positif yang benar - benar diberi label positif sebenarnya.

$$\text{recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad (2.11)$$

- *F1-Score* merujuk pada rata – rata dari *precision* dan *recall*
- *Support* merujuk pada total data pada setiap kelas pada dataset sebenarnya.

2.17.2 Receiver Operating Characteristic dan Area Under the Curve

Receiver Operating Characteristic (ROC) merupakan suatu alat visualisasi yang berguna untuk membandingkan dua model klasifikasi. Kurva ROC untuk suatu model yang diberikan menunjukkan pertukaran antara tingkat positif yang benar dan tingkat positif yang salah pada semua ambang batas (*threshold*) klasifikasi. Untuk meringkas kurva ROC dalam suatu kuantitas disebut juga *Area Under the Curve* (AUC) yang merupakan suatu pengukuran seluruh area di bawah kurva ROC. Nilai AUC yang lebih tinggi atau semakin besar areanya menunjukkan model klasifikasi yang dihasilkan lebih baik, yang menjadikan nilai AUC sebagai tujuan dalam pemaksimalan. Sebab area memiliki interpretasi yang bagus sebagai probabilitas pengklasifikasi dalam memberikan peringkat [32]. Berikut formula dalam mencari nilai AUC yang didapatkan dari hasil *confusion matrix* [33] :

$$AUC = \frac{1}{2} \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right) \quad (2.12)$$

Nilai AUC selalu memiliki *range* dari 0 hingga 1. Rentang ini menggambarkan luas area di bawah kurva yang dibentuk oleh hubungan antara sumbu x dan sumbu y. Nilai diatas 0,5 menunjukkan hasil yang menarik, karena ini mengindikasikan bahwa prediksi yang dilakukan lebih baik daripada prediksi acak, yang akan menghasilkan sebuah garis diagonal antara (0,0) dan (1,1) dengan luas 0,5. Dengan kata lain, semakin tinggi nilai AUC, semakin baik kualitas klasifikasi dan akurasi tes diagnostiknya [16].

Tabel 2.4 Kategori nilai AUC

<i>AUC Value</i>	<i>Category</i>
0.90 – 1.00	<i>excellent classification</i>
0.80 – 0.90	<i>good classification</i>
0.70 – 0.80	<i>fair classification</i>
0.60 – 0.70	<i>poor classification</i>
0.50 – 0.60	<i>failure</i>

2.18 Perbandingan Algoritma *Logistic Regression*, *Random Forest*, dan *XGBoost*

Adapun perbandingan keunggulan dan kelemahan dari ketiga algoritma yaitu *Logistic Regression*, *Random Forest*, dan *XGBoost* dapat dilihat pada tabel [] berikut.

Tabel 2.5 Perbandingan Algoritma *Machine Learning*

No	Algoritma	Keunggulan	Kelemahan
1	<i>Logistic Regression</i>	<ul style="list-style-type: none"> - Sederhana dan mudah diinterpretasikan - Efisien dalam komputasi dan bekerja dengan baik pada dataset yang relatif kecil - Cocok untuk masalah biner 	<ul style="list-style-type: none"> - Hanya memprediksi pada batas linear dan tidak dapat menangani hubungan kompleks antara fitur - Tidak dapat menangani ketergantungan non-linear antara fitur - Rentan terhadap <i>overfitting</i> jika terdapat banyak fitur yang tidak relevan
2	<i>Random Forest</i>	<ul style="list-style-type: none"> - Mampu menangani masalah klasifikasi dan regresi dengan baik - Mampu menangani dataset dengan jumlah fitur yang besar - mampu mengatasi masalah <i>overfitting</i> dengan cara mengatur parameter yang digunakan - mampu mengidentifikasi fitur-fitur penting dalam prediksi - mampu menangani fitur numerik dan kategorikal 	<ul style="list-style-type: none"> - lebih kompleks - memiliki interpretasi yang sulit - lebih sulit untuk dilatih karena komputasi yang lebih tinggi dibandingkan dengan <i>Logistic Regression</i> - rentan terhadap <i>overfitting</i> jika parameter tidak diatur dengan baik

3	<i>XGBoost</i>	<ul style="list-style-type: none"> - mampu menangani masalah klasifikasi dan regresi dengan baik - menghasilkan model yang kuat dan memiliki performa yang baik dalam kompetisi data - memiliki mekanisme penanganan <i>overfitting</i> melalui penyesuaian parameter - dapat menangani hubungan non-linier dan interaksi fitur 	<ul style="list-style-type: none"> - memiliki kompleksitas komputasi yang lebih tinggi dibandingkan <i>Logistic Regression</i> dan <i>Random Forest</i> - rentan terhadap <i>overfitting</i> jika parameter tidak diatur dengan baik - lebih sulit diinterpretasikan dibandingkan dengan <i>Logistic Regression</i>
---	----------------	---	--

2.19 Google Looker Studio

Google Looker Studio adalah sebuah alat yang dirancang untuk membuat visualisasi data dengan kemudahan yang luar biasa, menghasilkan presentasi data yang dinamis, responsif, dan menarik. Peluncurannya pada tahun 2016 telah memperkaya dunia visualisasi data dengan pendekatan yang inovatif. Sebuah hal yang menarik adalah Google Looker Studio dapat diakses secara bebas oleh siapa pun yang memiliki akun Google, menawarkan aksesibilitas yang lebih luas untuk memanfaatkan alat ini dalam menjelajahi dan menampilkan data dengan cara yang lebih menarik dan berinteraksi [34].

Salah satu kelebihan mencolok dari Google Looker Studio adalah kemudahannya dalam penggunaan, bahkan bagi pengguna yang masih baru dalam dunia visualisasi data. Selain itu, integrasinya yang meluas dengan berbagai sumber data memungkinkan pengguna untuk mengakses informasi dari berbagai sumber dengan mudah. Fitur-fitur yang disediakan oleh platform ini juga sangat beragam dan bermanfaat dalam menciptakan visualisasi data yang menarik dan informatif. Tidak hanya itu, laporan dan *dashboard* yang dihasilkan memiliki sifat interaktif dan responsif, sehingga memungkinkan pengguna untuk menjelajahi data secara lebih mendalam. Kemampuan untuk kolaborasi tim yang efisien memungkinkan

berbagai anggota tim untuk berkontribusi dalam pengolahan dan penyajian data secara bersama-sama [35].



Gambar 2.13 Contoh penggunaan Google Looker Studio [34]

2.20 Penelitian Terkait

Dalam penyusunan skripsi ini terdapat lima penelitian terkait yang dijadikan sebagai rujukan mengenai metode yang digunakan.

Penelitian yang dilakukan oleh Praveen Lalwani, dkk. [36] yang bertujuan untuk membandingkan prediksi *customer churn* di industri telekomunikasi menggunakan teknik *machine learning* terkenal yaitu *Logistic Regression*, *Naïve Bayes*, *Support Vector Machine*, *Decision Tree*, *Random Forest*, *XGBoost Classifier*, *CatBoost Classifier*, *AdaBoost Classifier* dan *Extra tree Classifier*. Hasil dari penelitian ini menunjukkan bahwa dua teknik ansambel yaitu *AdaBoost Classifier* dan *XGBoost Classifier* memiliki akurasi maksimum dengan skor AUC 84% untuk masalah

prediksi *churn* dan mengungguli algoritma lainnya dalam ukuran kinerja seperti akurasi, presisi, *f1-score*, *recall*, dan AUC.

Penelitian yang dilakukan oleh J. Pamina, dkk. [37] yang bertujuan untuk mengembangkan model prediksi *customer churn* dengan mengidentifikasi fitur-fitur yang sangat memengaruhi *churn* menggunakan teknik *machine learning* yaitu *KNN*, *Random Forest*, dan *XGBoost*. Hasil dari penelitian ini didapatkan model yang terbaik untuk diusulkan yaitu *XGBoost* dan menunjukkan bahwa pelanggan *fiber optic* dengan biaya bulanan yang lebih besar memiliki pengaruh yang lebih tinggi.

Penelitian yang dilakukan oleh Iqbal Hanif [38] yang bertujuan untuk membandingkan algoritma *XGBoost* dengan algoritma *Logistic Regression* dalam memprediksi *churn* dengan data kelas yang tidak seimbang. Hasil penelitian menunjukkan bahwa model *XGBoost* memiliki kemampuan yang lebih baik menangani data kelas yang tidak seimbang dan juga memiliki kemampuan yang lebih baik untuk memisahkan kelas *churn* dan kelas yang tidak *churn* daripada *Logistic Regression*.

Penelitian yang dilakukan oleh V. Kavitha [39] untuk memprediksi pelanggan perusahaan Telecom yang cenderung membatalkan langganan menggunakan algoritma *machine learning* agar dapat menawarkan layanan yang lebih baik dan mengurangi tingkat *churn*. Hasil dari penelitian ini menunjukkan bahwa *Random Forest* menghasilkan hasil yang lebih baik dibandingkan dengan algoritma *XGBoost* dan *Logistic Regression* dengan akurasi sebesar 80%.

Penelitian lain yang dilakukan oleh Sultan Abdulrahman Alshamsi [6] menggunakan CRISP-DM dan menerapkan 3 model *machine learning* yang berbeda yaitu *Decision tree*, *Logistic Regression*, dan *Random Forest*. Pada penelitian ini membandingkan 3 model *machine learning* yang berbeda untuk memprediksi *churn* pelanggan. Hasil dari penelitian ini menunjukkan bahwa model *Random Forest* lebih baik daripada model lainnya dengan akurasi dan *kappa score* masing-masing sebesar 93,5% dan 0,75.

Tabel 2.6 Penelitian Terkait

No	Peneliti	Data	Algoritma	Hasil
1	Praveen,dkk. (2021)	Data perusahaan telekomunikasi	<i>Logistic Regression, Naïve Bayes, Support Vector Machine, Decision Tree, Random Forest, Extra Tree Classifier, Adaboost, XGBoost, dan CatBoost</i>	Best model: AdaBoost dan <i>XGBoost</i> dengan akurasi 81,71% dan 80,8% , serta skor AUC tertinggi mencapai 84%
2	J. Pamina, dkk. (2019)	Data perusahaan telekomunikasi	<i>K-Nearest Neighbors, Random Forest, dan XGBoost</i>	Best model: <i>XGBoost</i> dengan akurasi sebesar 79,8% dan <i>f1-score</i> sebesar 58,2%.
3	Iqbal Hanif (2019)	Data Telkom Indonesia periode Oktober 2017 sampai Maret 2018	<i>XGBoost, Logistic Regression</i>	Best model: <i>XGBoost</i> dengan akurasi sebesar 97,9%, <i>sensitivity</i> sebesar 0,80 dan <i>specificity</i> sebesar 0.99
4	V. Kavitha (2020)	Data perusahaan telekomunikasi	<i>Random Forest, XGBoost, dan Logistic Regression</i>	Best model: <i>Random Forest</i> dengan akurasi 80%.
5	Sultan Abdulrahman Alshamsi (2022)	Data <i>e-commerce</i> website periode Juni 2021 sampai November 2021	<i>Decision tree, Logistic Regression, dan Random Forest</i>	Best model: <i>Random Forest</i> dengan akurasi 93,5% dan <i>kappa score</i> 0,75.

Berdasarkan tabel 2.5 diketahui bahwa sebelumnya telah dilakukan penelitian dalam membangun model untuk memprediksi pelanggan yang *churn* pada data *e-commerce* dengan membandingkan algoritma *decision tree*, *logistic regression*, dan *random forest* dalam mencari model yang terbaik. Sedangkan penelitian yang akan dilakukan mencoba membangun model prediksi pelanggan *churn* menggunakan data yang berbeda yaitu data *e-commerce* yang berfokus pada industri *fashion*. Selain itu, beberapa penelitian yang sama telah dilakukan menggunakan data perusahaan telekomunikasi yang menunjukkan bahwa algoritma *XGBoost* menjadi model terbaik dalam memprediksi pelanggan yang *churn*. Sehingga dalam penelitian ini akan mencoba membangun model menggunakan algoritma *XGBoost* untuk menganalisis performa model pada data *fashion e-commerce* dan membandingkannya dengan algoritma *logistic regression* dan *random forest* untuk mengetahui model terbaik dalam memprediksi pelanggan yang *churn*.

3.2 Alat dan Bahan Penelitian

3.2.1 Alat

Adapun penelitian ini menggunakan perangkat keras dan (*hardware*) dan perangkat lunak (*software*) dengan spesifikasi berikut.

Tabel 3.2 Alat dan Bahan Penelitian

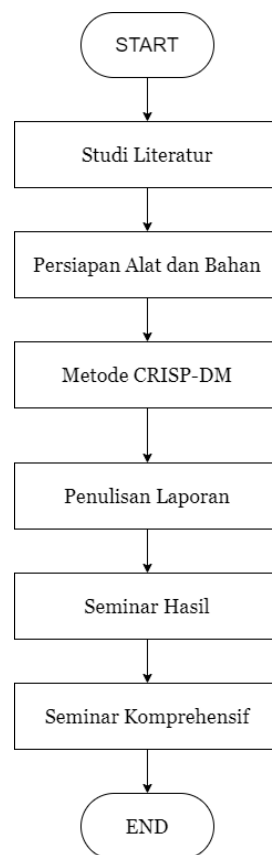
No	Perangkat	Spesifikasi	Deskripsi
1	Laptop	Lenovo Ideapad Slim 3 ryzen 3 4000 series, RAM 4 GB, SSD 512 dengan sistem operasi windows 10	Perangkat keras yang digunakan untuk pengembangan dan pengujian model <i>machine learning</i>
2	Python	Python 3.10.11	Bahasa pemrograman yang digunakan untuk membuat model <i>machine learning</i>
3	Google Colab		Perangkat lunak berupa <i>notebook</i> yang digunakan untuk menulis program python
4	Google Sheet		Perangkat lunak yang digunakan untuk menyimpan data yang sudah dilakukan proses <i>preprocessing</i>
5	Google Looker Studio		Perangkat lunak yang digunakan untuk memvisualisasi data dalam bentuk <i>dashboard</i> informatif agar mudah untuk dipahami

3.2.2 Bahan

Dalam penelitian ini bahan penelitian yang digunakan yaitu data *fashion e-commerce* yang didapatkan saat mengikuti Studi Independen Data Science Batch 3 di Startup Campus. Terdapat 4 dataset yang didapatkan yaitu data *customer*, data produk, data transaksi, dan data *click stream*.

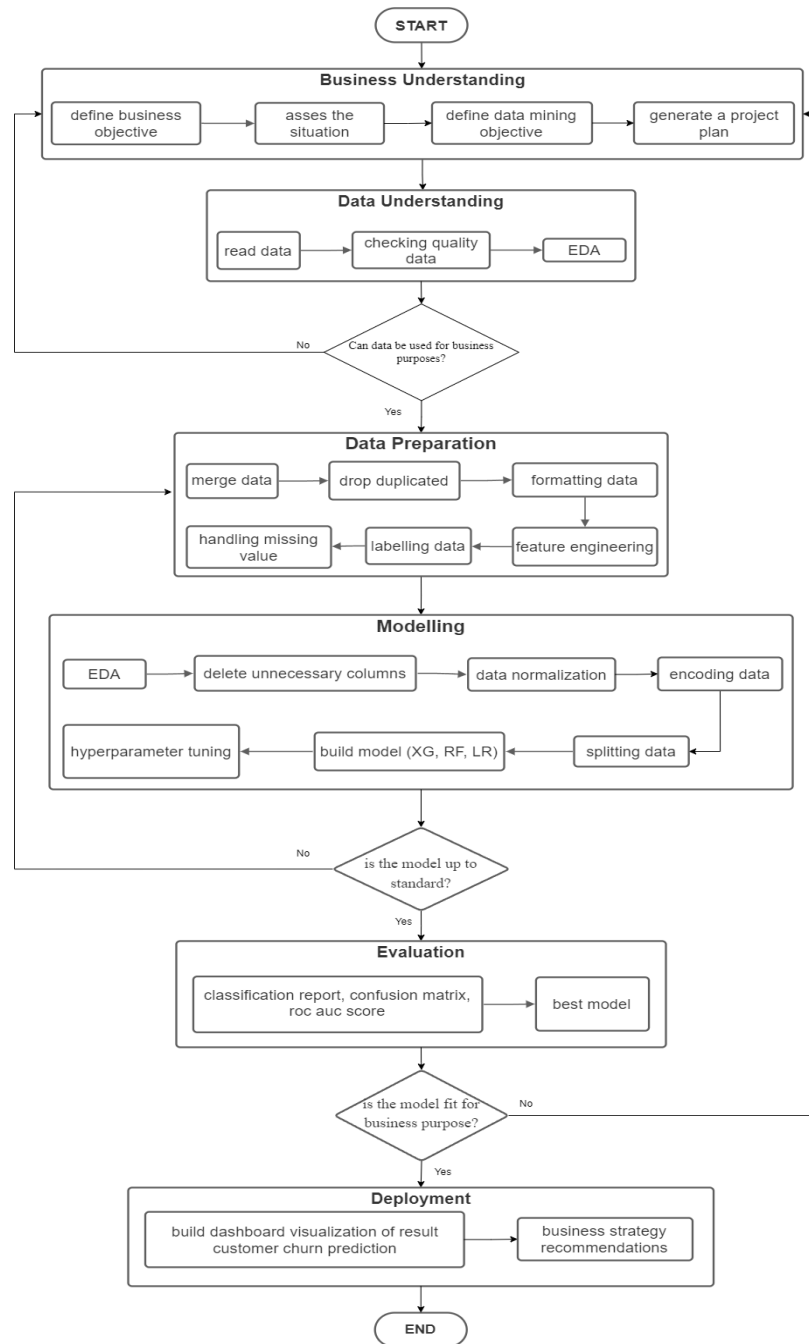
3.3 Tahapan Penelitian

Tahapan penelitian yang dilakukan dimulai dengan melakukan studi literatur yaitu mengumpulkan serta mempelajari ilmu pengetahuan dan penelitian terdahulu yang terkait bersumber dari jurnal, buku, maupun artikel. Ilmu yang dipelajari pada tahap studi literatur digunakan untuk mendukung penelitian. Kemudian mempersiapkan alat dan bahan yang akan digunakan dalam penelitian. Pada proses data mining yang dilakukan menggunakan metode CRISP-DM yang terdiri dari pemahaman bisnis, pemahaman data, persiapan data, pemodelan, evaluasi, dan penyebaran. Setelah proses yang ada pada metode CRISP-DM telah selesai dilakukan, selanjutnya menulis laporan. Laporan ini akan berisi hasil penelitian yang telah dilakukan. Selanjutnya dilakukan seminar hasil dan seminar komprehensif. Tahapan penelitian yang dilakukan dapat dilihat pada gambar diagram alir atau *flowchart* berikut.



Gambar 3.1 Tahapan Penelitian

Adapun tahapan penelitian ini untuk proses data mining menggunakan metode *Cross Industry Standard Process for Data Mining (CRISP-DM)*. Berikut adalah tahapan pada metode CRISP-DM yang dilakukan yaitu :



Gambar 3.2 Tahapan penelitian metode CRISP-DM

3.3.1 Tahap Pemahaman Bisnis (*Business Understanding Phase*)

Pada tahap pemahaman bisnis ini melakukan beberapa kegiatan, dimulai dari menentukan tujuan dan persyaratan dari perspektif bisnis, menilai situasi, menentukan tujuan data mining, dan menghasilkan rencana proyek. Pada penelitian yang akan dilakukan memiliki tujuan bisnis yaitu untuk mengetahui probabilitas pelanggan terhadap *churn* pada bulan yang akan datang. Hal tersebut dilakukan sebagai upaya perusahaan untuk mempertahankan loyalitas pelanggan sehingga dapat memiliki wawasan pelanggan yang lebih baik, mengurangi biaya pemasaran, dan meningkatkan pendapatan perusahaan. Selanjutnya menilai situasi yang sedang terjadi pada perusahaan dan dampak yang dapat ditimbulkan dari situasi tersebut. Kemudian menentukan tujuan dari data mining yang dilakukan yaitu mengolah dan menganalisis data yang telah dikumpulkan untuk membangun model prediksi pelanggan *churn*. Dari analisis data tersebut, dilakukan juga pengumpulan informasi mengenai karakteristik pelanggan untuk memahami pelanggan dengan lebih baik. Dari tujuan bisnis yang ditetapkan, hasil dari menilai situasi yang sedang terjadi di perusahaan, dan tujuan data mining yang dilakukan dapat dihasilkan rencana proyek yang akan dilakukan.

3.3.2 Tahap Pemahaman Data (*Data Understanding Phase*)

Tahapan pemahaman data merupakan tahapan setelah kita menentukan rencana proyek yang dihasilkan. Tahap pemahaman data ini melakukan beberapa kegiatan yang dimulai dari mengumpulkan data yang dibutuhkan, menjelaskan data, memeriksa kualitas data, dan melakukan eksplorasi terhadap data. Langkah yang dilakukan untuk memahami data adalah mencari tahu atribut-atribut dan tipe data apa saja yang ada dalam setiap dataset. Selanjutnya, memeriksa kualitas data dengan melakukan pemeriksaan data yang hilang secara keseluruhan pada dataset, serta melakukan penanganan yang tepat terhadap data yang hilang karena dapat memengaruhi analisis maupun model yang akan dibangun. Setelah itu, dilakukan eksplorasi setiap dataset dalam bentuk visualisasi untuk memahami data secara keseluruhan, mencari informasi yang tersembunyi di dalam data, mengidentifikasi masalah potensial, dan membantu dalam mempersiapkan dataset secara optimal untuk membangun model.

3.3.3 Tahap Persiapan Data (*Data Preparation Phase*)

Tahap persiapan data ini memiliki beberapa kegiatan yaitu dimulai dengan memilih dataset yang digunakan, melakukan penggabungan (*merge*) terhadap beberapa dataset yang berelasi, membersihkan data yang memiliki *missing value*, membuat atribut baru dari atribut – atribut yang sudah ada (*feature engineering*), dan memformat data untuk dapat diproses dengan data mining. Tahap persiapan data ini dilakukan dengan menggunakan *software* Google Colab dan bahasa pemrograman python dengan library Pandas yang digunakan untuk membaca dan mengintegrasikan dataset, membersihkan data seperti mengatasi *missing value* dengan mengganti, memodifikasi, atau menghapusnya, serta digunakan untuk menganalisis data.

Setelah data dipersiapkan, selanjutnya dilakukan proses memberikan label atau *class* pada data secara manual. Pada penelitian ini terdapat dua label yaitu *churn* dan *not churn*. Proses memberikan label dilakukan dengan mengelompokkan pembelian dalam periode tiga bulan dan dengan mengklasifikasikan pelanggan sebagai *churn* apabila pada periode berikutnya menghabiskan kurang dari 40% dari jumlah yang dibelanjakan pada periode referensi [40].

100 €	80 €	90 €	40 €	60 €	Non-Churner
Trimester 1	Trimester 2	Trimester 3	Trimester 4	Trimester 5	
100 €	80 €	90 €	35 €	30 €	Churner
Trimester 1	Trimester 2	Trimester 3	Trimester 4	Trimester 5	

Gambar 3.3 Contoh derivasi variabel *partial churning* [40]

Gambar 3.3 merupakan contoh kasus dalam mendefinisikan seorang pelanggan termasuk *churn* atau tidak dengan membandingkan total pembelian setiap trisemester. Berikut cara perhitungan untuk pendefinisian pelanggan *churn* dengan menghitung 40% total pembelian setiap trimester sebelumnya dan membandingkan dengan total pembelian trimester saat ini.

Adapun proses perhitungan dan perbandingan total pembelian dalam mendefinisikan pelanggan yang *churn*.

- 1) Menghitung 40% dari total pembelian pada setiap trimester

Kasus pertama

$$\text{Trimester 1 : } 40\% \times 100\text{€} = 40\text{€}$$

$$\text{Trimester 2 : } 40\% \times 80\text{€} = 32\text{€}$$

$$\text{Trimester 3 : } 40\% \times 90\text{€} = 36\text{€}$$

$$\text{Trimester 4 : } 40\% \times 40\text{€} = 16\text{€}$$

$$\text{Trimester 5 : } 40\% \times 60\text{€} = 24\text{€}$$

Kasus kedua

$$\text{Trimester 1 : } 40\% \times 100\text{€} = 40\text{€}$$

$$\text{Trimester 2 : } 40\% \times 80\text{€} = 32\text{€}$$

$$\text{Trimester 3 : } 40\% \times 90\text{€} = 36\text{€}$$

$$\text{Trimester 4 : } 40\% \times 35\text{€} = 14\text{€}$$

$$\text{Trimester 5 : } 40\% \times 30\text{€} = 12\text{€}$$

- 2) Membandingkan total pembelian terkini dengan 40% dari total pembelian sebelumnya

Misalkan:

- P_n adalah total pembelian yang dibelanjakan pada trimester ke-n
- D_1 adalah distribusi pembelian pertama (pelanggan yang tidak *churn*)
- D_2 adalah distribusi pembelian pertama (pelanggan yang *churn*)

Kasus pertama:

$$D_1 = (P_2 \text{ vs } 0.4P_1) \& (P_3 \text{ vs } 0.4P_2) \& (P_4 \text{ vs } 0.4P_3) \& (P_5 \text{ vs } 0.4P_4)$$

$$D_1 = (80\text{€} > 40\text{€}) \& (90\text{€} > 32\text{€}) \& (40\text{€} > 36\text{€}) \& (60\text{€} > 16\text{€})$$

Kasus kedua:

$$D_2 = (P_2 \text{ vs } 0.4P_1) \& (P_3 \text{ vs } 0.4P_2) \& (P_4 \text{ vs } 0.4P_3) \& (P_5 \text{ vs } 0.4P_4)$$

$$D_2 = (80\text{€} > 40\text{€}) \& (90\text{€} > 32\text{€}) \& (35\text{€} < 36\text{€}) \& (30\text{€} > 14\text{€})$$

Berdasarkan perhitungan yang telah dilakukan menunjukkan bahwa distribusi pembelian pertama (D_1) mewakili pelanggan yang tidak *churn* dikarenakan tidak ada kuartal dimana nilai yang dibelanjakan pada semua kuartal berikutnya kurang dari 40% dari jumlah yang dibelanjakan di kuartal tersebut. Sedangkan distribusi pembelian kedua (D_2) mewakili pelanggan yang *churn* dengan mempertimbangkan referensi kuartal ketiga dimana dapat disimpulkan bahwa pada periode berikutnya pelanggan ini membeli kurang dari 40% dari jumlah yang dibelanjakan pada kuartal ketiga. Sehingga dapat diasumsikan pelanggan ini *churn* pada awal kuartal keempat.

3.3.4 Tahap Pemodelan (*Modelling Phase*)

Tahap pemodelan ini dimulai dengan melakukan *Exploratory Data Analysis* (EDA). Kemudian menghilangkan atribut yang tidak digunakan dalam pemodelan seperti atribut *customer id*. Selanjutnya melakukan normalisasi data pada data numerik dan melakukan *encoding* data pada data kategorik. Selanjutnya memilih teknik pemodelan, membangun model berdasarkan data yang telah diproses sebelumnya dan menilai model serta memperbaiki model sehingga menghasilkan model *machine learning* yang memiliki akurasi yang paling baik. Dalam tahapan ini menggunakan Google Colab dengan bahasa pemrograman python dan untuk *machine learning* yang digunakan adalah algoritma *XGBoost*, *Random Forest*, dan *Logistic Regression*.

Setelah berhasil membangun model dan mengevaluasi ketiga algoritma *machine learning* tersebut, selanjutnya menentukan fitur penting (*feature importance*) yang mempengaruhi model dalam menentukan kategori untuk pelanggan yang *churn* atau pelanggan yang tidak *churn*. Fitur-fitur yang teridentifikasi penting akan digunakan dalam membangun ulang model. Kemudian hasilnya akan dibandingkan dengan hasil bila menggunakan seluruh fitur. Ada beberapa pendekatan yang dapat diterapkan untuk mengetahui nilai penting suatu fitur. Pendekatan untuk menganalisis *feature importance* pada model *Logistic Regression* berfokus pada koefisien yang diberikan kepada setiap fitur dalam persamaan regresi. Koefisien ini mengukur seberapa besar perubahan dalam variabel target yang terkait dengan

perubahan satu unit dalam fitur tertentu, dengan asumsi semua fitur lainnya tetap konstan. Pendekatan ini memberikan gambaran yang jelas tentang arah dan besarnya pengaruh setiap fitur terhadap prediksi target. Koefisien fitur dapat diakses melalui atribut “.coef_” pada objek model *Logistic Regression* setelah dilatih.

Pendekatan untuk menganalisis *feature importance* pada model Random Forest diukur dengan menghitung seberapa banyak rata-rata pengurangan kerapatan impuritas (*Gini Impurity*) yang dihasilkan oleh setiap fitur dalam semua pohon keputusan. Fitur yang mampu secara efektif memisahkan kelas target memiliki skor *feature importance* yang lebih tinggi. *Feature importance* pada model *Random Forest* dapat diakses menggunakan atribut “.feature_importances_” pada objek model. Kemudian dapat divisualisasikan hasilnya dalam bentuk grafik batang.

Pendekatan untuk menganalisis *feature importance* pada model *XGBoost* diukur berdasarkan seberapa sering fitur tersebut digunakan untuk memisahkan data selama pembentukan pohon keputusan dalam *ensemble*. Proses ini melibatkan dua komponen utama yaitu *gain importance* untuk mengukur sejauh mana penggunaan suatu fitur telah meningkatkan kualitas pemisahan pada setiap langkah dalam pembangunan pohon dan *cover importance* untuk mengukur seberapa banyak data yang dipengaruhi oleh pembagian yang melibatkan suatu fitur. *Feature importance* pada model *XGBoost* dapat diakses menggunakan atribut “.feature_importances_”, yang merupakan cara untuk mengakses informasi mengenai kontribusi relatif setiap fitur dalam model terhadap pembentukan prediksi. Semakin besar skor *importance*, semakin besar kontribusi fitur dalam mempengaruhi prediksi model.

3.3.5 Tahap Evaluasi (*Evaluation*)

Tahap evaluasi dilakukan dengan menilai sejauh mana model memenuhi tujuan bisnis dan evaluasi ini juga menilai kualitas dan efektivitas model yang dihasilkan. Evaluasi dilakukan dengan mempertimbangkan hasil dari pengukuran kinerja klasifikasi yaitu *classification report* dan *confusion matrix*, serta nilai ROC AUC pada model *Logistic Regression*, *Random Forest* dan *XGBoost*. Selain itu, tahapan ini menentukan langkah yang akan dilakukan pada tahap *deployment* atau bila

terdapat tahapan yang terlewat maka dapat kembali pada tahapan awal yaitu *business understanding*.

3.3.6 Tahap Penyebaran (*Deployment*)

Tahap penyebaran ini dilakukan dengan membuat *dashboard* menggunakan *tool* Google Looker Studio dalam memvisualisasikan hasil prediksi yang telah diperoleh sehingga lebih mudah untuk dipahami. Hasil visualisasi data yang diperoleh akan dianalisis untuk mendapatkan *insight* dari data yang disajikan. *Insight* yang didapat dikombinasikan dengan analisis dari eksplorasi data-data perusahaan XYZ, serta melibatkan pendapat *expert* untuk dijadikan acuan dalam penyusunan rekomendasi strategi pemasaran. Rekomendasi-rekomendasi strategi pemasaran yang diusulkan dapat diterapkan tim marketing dan tim bisnis ke depannya dengan harapan dapat mengurangi *customer* yang akan *churn* demi membangun loyalitas pelanggan dan meningkatkan profitabilitas perusahaan.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan yang diperoleh dari penelitian prediksi pelanggan *churn* menggunakan algoritma *Logistic Regression*, *Random Forest*, dan *XGBoost* adalah sebagai berikut.

1. Berdasarkan hasil yang diperoleh dari penelitian ini, algoritma yang memiliki performa terbaik untuk model prediksi pelanggan *churn* adalah *XGBoost* dengan nilai akurasi 97% , *precision* sebesar 97%, *recall* sebesar 98%, *f1-score* sebesar 98%, dan nilai AUC mencapai 0.995.
2. Berdasarkan *feature importance* dari model *XGBoost*, diperoleh 7 fitur yang memiliki dampak signifikan terhadap kecenderungan pelanggan untuk *churn* atau tetap berlangganan, yaitu keaktifan pelanggan, jumlah item yang ada dalam keranjang belanja, frekuensi transaksi, jumlah variasi produk yang dibeli, *recency*, *tenure*, dan status pelanggan.
3. Berdasarkan percobaan membangun ulang model menggunakan fitur-fitur penting *XGBoost* terbukti bahwa pendekatan ini mampu mengoptimalkan kinerja model dengan lebih efisien dari segi waktu komputasi dan tetap efektif.
4. Hasil prediksi yang digunakan dalam tahap penyebaran dibuat dengan menerapkan model *XGBoost*, kemudian data hasil prediksi divisualisasikan dalam bentuk *column chart*, *pie chart*, dan tabel menggunakan penerapan Google Looker Studio.

5.2 Saran

Saran yang dapat diberikan untuk penelitian selanjutnya berdasarkan penelitian yang telah dilakukan adalah sebagai berikut.

1. Melakukan analisis lebih lanjut mengenai alasan pelanggan yang *churn* melalui survei kepuasan pelanggan dan memperkaya data perusahaan mengenai ulasan pelanggan untuk menilai produk yang dibeli.
2. Melakukan segmentasi pelanggan terlebih dahulu untuk memahami karakteristik pelanggan dengan lebih baik dan dapat digunakan juga sebagai referensi fitur sebelum dilakukan proses pemodelan.
3. Membangun model prediksi *churn* menggunakan fitur-fitur yang memiliki pengaruh yang tinggi terhadap kecenderungan pelanggan untuk *churn* atau tetap berlangganan.

DAFTAR PUSTAKA

- [1] S. R. Mege, N. I. Kurniawati, R. E. Werdani, S. Suwandi, and F. U. Nida, "Sustainability of E Commerce Business through Logistic System in the COVID 19 Pandemic," *South Asian Res J Bus Manag*, vol. 4, no. 3, pp. 122–132, Jun. 2022, doi: 10.36346/sarjbm.2022.v04i03.005.
- [2] Google, TEMASEK, and BAIN & COMPANY, "e-Conomy SEA 2020," Nov. 2020. Accessed: May 28, 2023. [Online]. Available: <https://www.bain.com/insights/e-conomy-sea-2020/>
- [3] H. Limanseto, "Akselerasi Ekonomi Digital pada e-Commerce dan Online Travel Menjadi Salah Satu Strategi Efektif Mendorong Kinerja Perekonomian Nasional - Kementerian Koordinator Bidang Perekonomian Republik Indonesia," Kementerian Koordinator Bidang Perekonomian Republik Indonesia. Accessed: May 28, 2023. [Online]. Available: <https://www.ekon.go.id/publikasi/detail/3978/akselerasi-ekonomi-digital-pada-e-commerce-dan-online-travel-menjadi-salah-satu-strategi-efektif-mendorong-kinerja-perekonomian-nasional>
- [4] A. Ahdiat, "Banyak Konsumen Lebih Pilih E-Commerce untuk Belanja Fashion | Databoks," databoks. Accessed: Aug. 23, 2023. [Online]. Available: <https://databoks.katadata.co.id/infografik/2022/09/08/banyak-konsumen-lebih-pilih-e-commerce-untuk-belanja-fashion>
- [5] C. Gold and T. Tzuo, *Fighting churn with data: the science and strategy of customer retention*. Shelter Island: Manning, 2020.
- [6] A. Alshamsi, "Customer Churn prediction in ECommerce Sector," Rochester Institute of Technology of Dubai, 2022. [Online]. Available: <https://scholarworks.rit.edu/cgi/viewcontent.cgi?article=12319&context=theses>
- [7] A. Hartman, J. Sifonis, and J. Kador, *Summary Net Ready : Strategies For Success in the E-conomy*. Primento Digital, 2014.
- [8] K. C. Laudon and C. G. Traver, *E-commerce: business, technology, society*, Tenth edition. Upper Saddle River, New Jersey: Pearson, 2014.
- [9] W. Buckinx and D. Van Den Poel, "Customer base analysis: partial defection of behaviourally loyal clients in a non-contractual FMCG retail setting,"

European Journal of Operational Research, vol. 164, no. 1, pp. 252–268, Jul. 2005, doi: 10.1016/j.ejor.2003.12.010.

[10] M. Saghir, Z. Bibi, S. Bashir, and F. H. Khan, “Churn Prediction using Neural Network based Individual and Ensemble Models,” in *2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, Islamabad, Pakistan: IEEE, Jan. 2019, pp. 634–639. doi: 10.1109/IBCAST.2019.8667113.

[11] F. A. Buttle, *Customer relationship management: concepts and technologies*, 2. ed. London: Routledge, 2012.

[12] R. Wirth and J. Hipp, “CRISP-DM: Towards a standard process model for data mining,” *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, Jan. 2000.

[13] P. Chapman *et al.*, “CRISP-DM 1.0: Step-by-step data mining guide,” 2000. Accessed: Jul. 20, 2023. [Online]. Available: <https://www.semanticscholar.org/paper/CRISP-DM-1.0%3A-Step-by-step-data-mining-guide-Chapman-Clinton/54bad20bbc7938991bf34f86dde0babfbd2d5a72>

[14] J. Han and M. Kamber, *Data mining: concepts and techniques*, 3rd ed. Burlington, MA: Elsevier, 2012.

[15] J. Kazil and K. Jarmul, *Data wrangling with Python*, First edition. Sebastopol, CA: O’Reilly Media, 2016.

[16] A. Zheng and A. Casari, *Feature engineering for machine learning: principles and techniques for data scientists*, First edition. Beijing: Boston: O’Reilly, 2018.

[17] T. Rawat and Dr. V. Khemchandani, “Feature Engineering (FE) Tools and Techniques for Better Classification Performance,” *IJIET*, vol. 8, no. 2, 2017, doi: 10.21172/ijiet.82.024.

[18] S. Ozdemir, *Feature engineering bookcamp*. Shelter Island, NY: Manning Publications Co, 2022.

[19] B. Mahesh, “Machine Learning Algorithms - A Review,” *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381–386, 2018, [Online]. Available: <https://www.ijsr.net/archive/v9i1/ART20203995.pdf>

[20] O. Theobald, *MACHINE LEARNING FOR ABSOLUTE BEGINNERS: a plain english introduction*, Second Edition. Scatterplot Press, 2017.

[21] R. Hans, “Variasi Jenis Algoritma Machine Learning, Sudah Tahu?,” DQLab. Accessed: May 28, 2023. [Online]. Available: <https://dqlab.id/variiasi-jenis-algoritma-machine-learning-sudah-tahu>

- [22] I. H. Witten, E. Frank, and M. A. Hall, *Data mining: practical machine learning tools and techniques*, 3rd ed. in in Morgan Kaufmann series in data management systems. Burlington, MA: Morgan Kaufmann, 2011.
- [23] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. in in Machine Learning & Pattern Recognition Series. Boca Raton, Fla.: CRC Press, Taylor & Francis, 2012.
- [24] D. Martins, “XGBoost: A Complete Guide to Fine-Tune and Optimize your Model,” Medium. Accessed: May 28, 2023. [Online]. Available: <https://towardsdatascience.com/xgboost-fine-tune-and-optimize-your-model-23d996fab663>
- [25] T. Yiu, “Understanding Random Forest,” Medium. Accessed: May 28, 2023. [Online]. Available: <https://towardsdatascience.com/understanding-random-forest-58381e0602d2>
- [26] “Random Forest Hyperparameter Tuning in Python,” *GeeksforGeeks*, Dec. 2022, Accessed: May 28, 2023. [Online]. Available: <https://www.geeksforgeeks.org/random-forest-hyperparameter-tuning-in-python/>
- [27] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [28] “XGBoost Parameters — xgboost 1.7.5 documentation.” Accessed: May 28, 2023. [Online]. Available: <https://xgboost.readthedocs.io/en/stable/parameter.html>
- [29] “sklearn.linear_model.LogisticRegression,” scikit-learn. Accessed: May 28, 2023. [Online]. Available: https://scikit-learn/stable/modules/generated/sklearn.linear_model.LogisticRegression.html
- [30] Y. Ali, E. Awwad, M. Al-Razgan, and A. Maarouf, “Hyperparameter Search for Machine Learning Algorithms for Optimizing the Computational Complexity,” *Processes*, vol. 11, no. 2, p. 349, Jan. 2023, doi: 10.3390/pr11020349.
- [31] I. Ismiguzel, “Hyperparameter Tuning with Grid Search and Random Search,” Medium. Accessed: May 28, 2023. [Online]. Available: <https://towardsdatascience.com/hyperparameter-tuning-with-grid-search-and-random-search-6e1b5e175144>
- [32] G. Florin, *Data mining: concepts, models and techniques*. in in Intelligent systems reference library, no. 12. Berlin, Heilelberg: Springer-Verlag, 2011.
- [33] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Information Processing & Management*, vol. 45, no. 4, pp. 427–437, Jul. 2009, doi: 10.1016/j.ipm.2009.03.002.

- [34] G. Snipes, “Google Data Studio,” *Journal of Librarianship and Scholarly Communication*, vol. 6, no. 1, Feb. 2018, doi: 10.7710/2162-3309.2214.
- [35] L. Hurst, *Hands on with google data studio: a data citizens survival guide*. Indianapolis: John Wiley and Sons, 2020.
- [36] P. Lalwani, M. K. Mishra, J. S. Chadha, and P. Sethi, “Customer churn prediction system: a machine learning approach,” *Computing*, vol. 104, no. 2, pp. 271–294, Feb. 2022, doi: 10.1007/s00607-021-00908-y.
- [37] B. Raja and P. Jeyakumar, “An Effective Classifier for Predicting Churn in Telecommunication,” *Journal of Advanced Research in Dynamical and Control Systems*, vol. 11, pp. 221–229, Jun. 2019.
- [38] I. Hanif, “Implementing Extreme Gradient Boosting (XGBoost) Classifier to Improve Customer Churn Prediction,” in *Proceedings of the Proceedings of the 1st International Conference on Statistics and Analytics, ICSA 2019, 2-3 August 2019, Bogor, Indonesia*, Bogor, Indonesia: EAI, 2020. doi: 10.4108/eai.2-8-2019.2290338.
- [39] V. Kavitha, G. Hemanth Kumar, S. V Mohan Kumar, M. Harish, and JNTUA College of Engineering, “Churn Prediction of Customer in Telecom Industry using Machine Learning Algorithms,” *IJERT*, vol. V9, no. 05, p. IJERTV9IS050022, May 2020, doi: 10.17577/IJERTV9IS050022.
- [40] V. L. Miguéis, D. Van den Poel, A. S. Camanho, and J. Falcão e Cunha, “Modeling partial customer churn: On the value of first product-category purchase sequences,” *Expert Systems with Applications*, vol. 39, no. 12, pp. 11250–11256, Sep. 2012, doi: 10.1016/j.eswa.2012.03.073.