

**MODEL PEMBELAJARAN DAN LAPORAN AKHIR
PROJECT-BASED LEARNING
MATA KULIAH MACHINE LEARNING
KELAS B**



**“PERBANDINGAN KINERJA METODE KLASIFIKASI DALAM KEPUASAN
PENUMPANG PESAWAT TERBANG TAHUN 2019”**

DISUSUN OLEH KELOMPOK 06 :

- | | |
|----------------------------|-----------------|
| 1. MOHAMMAD NIZAR RISWANDA | (21083010015) |
| 2. MEISYA VIRA AMELIA | (21083010018) |
| 3. RHEINKA ELYANA SUPRAPTO | (21083010021) |
| 4. EDINA ALANA NABILA | (21083010022) |

DOSEN PENGAMPU:

AVIOLLA TERZA DAMALIANA, S.SI., M.Stat
Dr. Eng. Ir. ANGGRAINI PUSPITA SARI, ST., MT.

PROGRAM STUDI SAINS DATA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
JAWA TIMUR

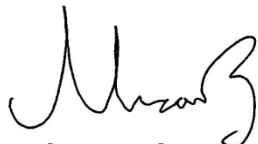
SURAT PERNYATAAN

Kami yang bertanda tangan di bawah ini:

1. NAMA : Mohammad Nizar Riswanda
NPM : 21083010015
2. NAMA : Meisya Vira Amelia
NPM : 21083010018
3. NAMA : Rheinka Elyana Suprpto
NPM : 21083010021
4. NAMA : Edina Alana Nabila
NPM : 21083010022

Dengan ini menyatakan bahwa hasil pekerjaan yang kami serahkan sebagai bagian dari penilaian mata kuliah machine learning adalah benar-benar karya orisinal kami, bukan milik orang lain, dan tidak pernah digunakan dalam penilaian tugas yang lain dalam mata kuliah apapun, baik secara keseluruhan ataupun sebagian, di Universitas Pembangunan Nasional “Veteran” Jawa Timur ataupun di institusi lainnya. Apabila di kemudian hari terbukti bahwa kami melakukan kecurangan maka kami bersedia menerima sanksi sesuai dengan aturan yang berlaku di Universitas Pembangunan Nasional “Veteran” Jawa Timur.

Surabaya, 14 Juni 2023



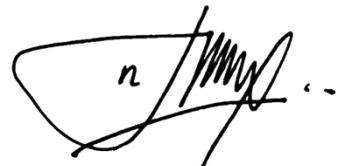
Mohammad Nizar R.



Meisya Vira Amelia



Rheinka Elyana S.



Edina Alana N.

KATA PENGANTAR

Puji Syukur ke Hadirat Tuhan Yang Maha Esa, yang telah memberikan rahmat-Nya seiring dengan selesainya penyusunan Laporan Model Pembelajaran dan Laporan Akhir Project-Based Learning Mata Kuliah Machine Learning jurusan Sains Data Fakultas Ilmu Komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur.

Pada dasarnya laporan ini merupakan bukti nyata pelaksanaan Praktikum Analisis Data menggunakan machine learning di jurusan Sains Data. Di samping itu, beberapa hal yang termuat di dalamnya dapat dijadikan referensi untuk memahami mata kuliah tertentu pada semester lanjut. Kami mengucapkan terima kasih kepada Ibu AVIOLLA TERZA DAMALIANA, S.SI., M.Stat dan Ibu Dr. Eng. Ir. ANGGRAINI PUSPITA SARI, ST., MT. selaku dosen pengampu mata kuliah machine learning.

Juga kepada rekan-rekan Sains Data Angkatan 2021 dan seluruh pihak yang mendukung penyempurnanaan laporan project-based learning kami. Laporan ini masih memiliki banyak kelemahan, oleh karena itu sangat dibutuhkan kritik dan saran yang membangun sebagai bahan bagi perbaikan laporan ini dimasa yang akan datang. Akhir kata, kami berharap laporan ini dapat bermanfaat bagi pembaca. Sekian dan Terima Kasih.

ABSTRACT

The United States is a constitutional federal republic consisting of fifty states and a federal district. Air transportation is one of the major modes of transportation in the country, serving various types of passengers including tourists, business travelers, and families. Therefore, airlines in the United States routinely conduct passenger satisfaction surveys to understand passengers' perceptions and opinions about airline services. Through the analysis of this dataset, this research aims to determine the appropriate machine learning method to be applied for analysis. The results of the discussion show that the XGBoost method yields the highest accuracy rate of 95.11% and performs the best based on precision, recall, and F1-score values. On the other hand, the SVM method has the lowest accuracy rate of 93.86%. The findings of this study provide insights for airlines to take appropriate steps in improving their services to passengers. By understanding the factors that influence passenger satisfaction, airlines can enhance the passenger experience and build a positive image for their company.

Keywords: XGBoost, SVM, Random Forest, Airplane Passenger

ABSTRAK

Amerika Serikat adalah sebuah negara republik konstitusional federal yang terdiri dari lima puluh negara bagian dan sebuah distrik federal. Transportasi udara menjadi salah satu transportasi utama di negara ini, dengan berbagai penumpang pesawat terbang termasuk wisatawan, pebisnis, dan keluarga. Oleh karena itu, maskapai di Amerika Serikat melakukan survei kepuasan penumpang secara rutin untuk memahami persepsi dan pendapat penumpang terhadap layanan maskapai. Melalui analisis dataset ini, penelitian ini dapat menjawab metode machine learning apa yang tepat untuk diterapkan sebagai analisis. Hasil pembahasan menunjukkan bahwa metode XGBoost memberikan hasil akurasi tertinggi dengan nilai 95,11. Metode SVM, di sisi lain, memiliki akurasi terendah sebesar 93,86%. Hasil penelitian ini dapat memberikan wawasan kepada maskapai penerbangan untuk mengambil langkah yang tepat dalam memperbaiki pelayanan kepada penumpang. Dengan memahami faktor-faktor yang berpengaruh pada kepuasan penumpang, maskapai dapat meningkatkan pengalaman penumpang dan membangun citra positif bagi perusahaan mereka.

Kata kunci: XGBoost, SVM, Random Fores, Airplane Passenger

DAFTAR ISI

SURAT PERNYATAAN	2
KATA PENGANTAR.....	3
ABSTRACT	4
ABSTRAK.....	4
DAFTAR ISI	1
DAFTAR GAMBAR.....	3
DAFTAR TABEL	4
BAB I PENDAHULUAN	5
1.1 Latar Belakang	5
1.2 Permasalahan	6
1.3 Tujuan	6
1.4 Manfaat	6
BAB II TINJAUAN PUSTAKA.....	7
2.1 Teori Penunjang	7
2.1.1 Passenger Satisfaction.....	7
2.1.2 Support Vector Machine	7
2.1.3 <i>XGBoost</i>	8
2.1.4 Random Forest.....	9
2.1.5 Python	11
2.1.6 Confusion Matrix	11
2.2 Penelitian Terkait	12
2.2.1 Prediksi Kepuasan Penumpang Maskapai Penerbangan dengan Algoritma Klasifikasi	12
2.2.2 Klasifikasi Berita Menggunakan Metode Support Vector Machine.....	12
2.2.3 Klasifikasi Pemegang Polis Menggunakan Metode <i>XGBoost</i>	13
BAB III METODOLOGI PENELITIAN	14
3.1 Pengumpulan Data	14
3.2 Pre-Processing.....	14
3.2.1 Informasi Dataset	15
3.2.2 Menghapus Kolom Id	15
3.2.3 Mengubah Nama Kolom.....	15
3.2.4 <i>Missing Value</i>	15
3.2.5 Duplikasi Data	15
3.2.6 Deteksi <i>Outlier</i>	15
3.3 Exploratory Data Analysis	16

3.4 Visualisasi	16
3.5 Linearitas.....	16
3.6 Mapping	16
3.7 Pemodelan Algoritma	16
3.8 <i>Feature Importance</i>	17
BAB IV HASIL DAN PEMBAHASAN.....	18
4.1 Dataset.....	18
4.2 Pre-Processing.....	18
4.2.1 Mengganti Nama Kolom.....	18
4.2.2 Menangani <i>Missing Value</i>	18
4.2.3 Menangani Duplikasi Data	19
4.2.4 Mendeteksi Data <i>Outlier</i>	19
4.3 Exploratory Data Analysis (EDA)	20
4.4 Visualisasi	21
4.5 Mapping Data.....	24
4.6 Splitting Data	24
4.7 Data Modelling	25
4.7.1 XGBoost	25
4.7.2 Support Vector Machine	25
4.7.3 Random Forest	26
4.8 Performance Analysis	26
4.8.1 Evaluasi Performa XGBoost.....	26
4.8.2 Evaluasi Performa SVM	29
4.8.3 Evaluasi Performa Random Forest	33
4.9 Pemilihan Akhir Model.....	36
4.10 Fitur Penting Model	37
BAB V KESIMPULAN.....	38
DAFTAR PUSTAKA	39
LAMPIRAN	40

DAFTAR GAMBAR

Gambar 1 Flowchart Support Vector Machine	7
Gambar 2 Flowchart XGBoost	8
Gambar 3 Flowchart Random Forest	9
Gambar 4 Diagram Alur Penelitian	14
Gambar 5 Boxplot dari Fitur Distance	19
Gambar 6 Boxplot dari Departure_Delay_Minutes	20
Gambar 7 Boxplot dari Arrival_Delay_Minutes	20
Gambar 8 Hasil Count Describe	20
Gambar 9 Hasil dari Exploratory Data Analysis	21
Gambar 10 Visualisasi Gender dengan Satisfaction_V2	21
Gambar 11 Visualisasi Satisfaction_V2 dengan Age	22
Gambar 12 Visualisasi Satisfaction_V2 dengan Dep_Val_Time_Convenient	22
Gambar 13 Visualisasi Satisfaction_V2 dengan Gate	22
Gambar 14 Scatter Plot Departure_Delay_Minutes dan Arrival_Delay_Minutes	23
Gambar 15 Visualisasi Scatter Plot Distance dan Departure_Delay_Minutes	23
Gambar 16 Confusion Mtarix XGBoost 90:10	28
Gambar 17 Confusion Matrix XGBoost 80:20	28
Gambar 18 Confusion Matrix XGBooxt 70:30	29
Gambar 19 Confusion Matrix SVM 90:10	31
Gambar 20 Confusion Matrix SVM 80:20	32
Gambar 21 Confusion Matrix SVM 70:30	32
Gambar 22 Confusion Matrix Random Forest 90:10	35
Gambar 23 Confusion Matrix Random Forest 80:20	35
Gambar 24 Confusion Matrix Random Forest 70:30	35
Gambar 25 Feature Importances	37

DAFTAR TABEL

Tabel 1 Confusion Matrix.....	11
Tabel 2 Perbandingan Error Tiap Rasio.....	29
Tabel 3 Perbandingan Error Tiap Rasio.....	32
Tabel 4 Perbandingan Error Tiap Rasio.....	36
Tabel 5 Pemilihan Akhir Model	36
Tabel 6 Perbandingan Nilai Error Rasio Tiap Model	36

BAB I

PENDAHULUAN

1.1 Latar Belakang

Amerika Serikat merupakan sebuah negara republik konstitusional federal yang terdiri lima puluh negara bagian dan sebuah distrik federal. Negara ini terletak di bagian tengah Benua Amerika Utara, yang menjadi lokasi dari empat puluh lima negara bagian yang saling bersebelahan, beserta distrik ibu kota Washington, D.C. Karena itu, transportasi udara menjadi salah satu transportasi utama di negara ini, Karena transportasi udara menjadi transportasi utama, maka terdapat berbagai penumpang pesawat terbang termasuk wisatawan, pebisnis, dan keluarga. Setiap penumpang memiliki kebutuhan berbeda yang harus dipertimbangkan oleh maskapai penerbangan dalam menyediakan layanan. Maka dari itu, maskapai berlomba-lomba memberikan pengalaman terbaik kepada penumpang mereka agar mempertahankan pelanggan dan membangun citra positif bagi maskapai penerbangan yang kuat.

Dalam upaya membangun citra positif dalam pelayanan, maka maskapai di Amerika Serikat secara rutin melakukan survei kepuasan penumpang untuk memahami persepsi dan pendapat penumpang terhadap layanan maskapai. Yang mana survei tersebut dapat digunakan sebagai evaluasi maskapai dan juga mempengaruhi keputusan penumpang dalam memilih maskapai penerbangan yang akan mereka gunakan.

Menggunakan teknik machine learning pada penelitian ini bertujuan untuk menganalisis dataset survei kepuasan penumpang maskapai penerbangan AS. Algoritma machine learning membantu untuk mengidentifikasi pola dan tren dari penumpang maskapai yang didapat dari dataset yang kompleks. Melalui analisis dataset ini, kita dapat menjawab berbagai pertanyaan yang relevan seperti faktor yang memiliki kontribusi lebih dalam kepuasan penumpang, adakah perbedaan tingkat kepuasan maskapai tertentu. Sehingga maskapai dapat mengambil langkah yang tepat untuk memperbaiki pelayanan kepada penumpang.

1.2 Permasalahan

- a. Sulitnya mengetahui pola dari konsumen atau penumpang disebuah maskapai penerbangan.
- b. Belum tersedianya analisis perbandingan kinerja metode machine learning pada dataset kepuasan penumpang pesawat terbang pada tahun 2019.

1.3 Tujuan

- a. Melakukan perbandingan metode machine learning dalam analisis klasifikasi dataset kepuasan penumpang pesawat terbang tahun 2019.
- b. Menghasilkan satu metode machine learning yang paling akurat diterapkan untuk melakukan analisis klasifikasi dataset kepuasan penumpang pesawat terbang tahun 2019.

1.4 Manfaat

- a. Manfaat bagi Masyarakat

Karena banyaknya maskapai yang menawarkan pelayanan yang baik, maka dengan adanya analisis dan dataset ini masyarakat dapat memperoleh manfaat yaitu memiliki kehati-hatian dalam memilih sebuah maskapai penerbangan. Karena pasti masyarakat juga menginginkan pelayanan dari sebuah maskapai yang paling baik dan maksimal. Selain itu, masyarakat dapat menghindari kekecewaan atas pelayanan yang dilakukan oleh sebuah maskapai.

- b. Manfaat bagi Maskapai

Bagi perusahaan sektor penerbangan atau maskapai mendapatkan manfaat yaitu dapat mengetahui pola dari konsumen atau penumpang maskapai penerbangan. Selain itu, maskapai juga mendapatkan metode yang cocok digunakan untuk melakukan analisis konsumen penerbangan dan dengan analisis tersebut maskapai mampu untuk mengembangkan strategi bisnis yang dapat diterapkan dalam perusahaan.

BAB II

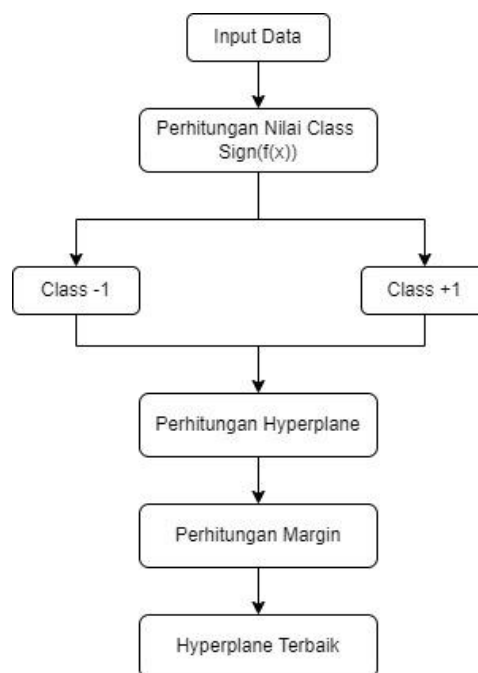
TINJAUAN PUSTAKA

2.1 Teori Penunjang

2.1.1 Passenger Satisfaction

Passenger satisfaction merupakan sebuah dataset yang dikumpulkan dari wilayah Negara Amerika Serikat. Dataset tersebut merupakan sebuah data yang berisi informasi tentang kepuasan penumpang maskapai penerbangan di Amerika Serikat. Dataset ini dikumpulkan dengan tujuan untuk memahami pola konsumen dan tingkat kepuasan penumpang terhadap layanan maskapai yang mereka terima selama perjalanan.

2.1.2 Support Vector Machine



Gambar 1 Flowchart Support Vector Machine

Support vector machine merupakan sebuah algoritma machine learning yang berfungsi untuk mengubah data training menjadi data yang memiliki dimensi lebih tinggi dengan tujuan membangun sebuah hyperplane atau batas keputusan yang dapat memisahkan dua kelas data dengan baik. Konsep klasifikasi dengan SVM adalah mencari hyperplane terbaik yang memiliki fungsi sebagai pemisah dari dua kelas data, opini positif (+1) dan opini negatif(-1). Persamaan SVM seperti pada rumus di bawah ini:

$$w \cdot x + b = 0$$

Dengan:

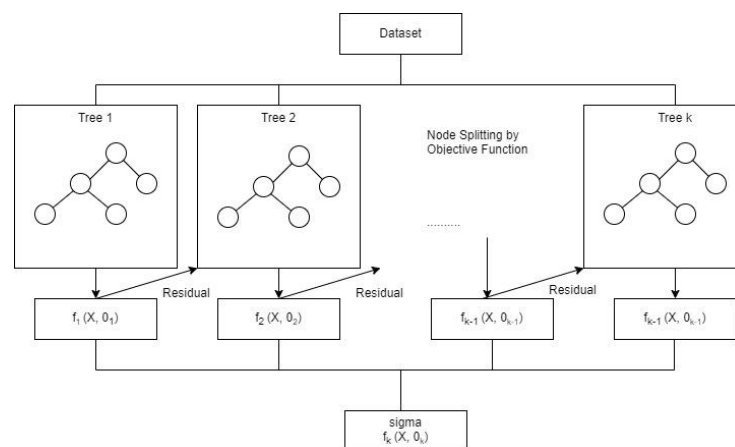
w = Parameter hyperplane yang dicari (garis yang tegak lurus antara garis hyper plane dan titik support vector)

x = Titik data masukan Support Vector Machine

b = Parameter hyperlane yang dicari (nilai bias)

Kelebihan dari algoritma *support vector machine* adalah sebagai berikut yaitu efektif dalam dataset yang memiliki dimensi tinggi, mampu mengatasi masalah klasifikasi yang tidak linier, mampu mengurangi resiko *overfitting* dan memiliki kemampuan penanganan data outliers yang baik. Untuk kekurangan dari algoritma *support vector machine* sendiri adalah sebagai berikut yaitu proses dari algoritma *support vector machine* berlangsung dalam waktu yang lama. Karena apabila pemilihan parameter yang tidak tepat dapat mempengaruhi kinerja model.

2.1.3 XGBoost



Gambar 2 Flowchart XGBoost

XGBoost merupakan algoritma berbasis *decision tree* yang efektif dalam menentukan sebuah solusi dari regresi hingga klasifikasi. Cara kerja dari algoritma ini adalah mengubah parameter *data training* secara berulang untuk mengurangi fungsi *loss*. Dengan kemampuan dan cara kerja *XGBoost* tersebut, kesalahan atau *overfitting* dapat dihindari

Pada metode ini diperlukan fungsi objektif yang berguna untuk menilai seberapa bagus model yang didapatkan sesuai dengan *data training*. Karakteristik

yang terpenting dari fungsi objektif terdiri dari 2 bagian yaitu nilai pelatihan yang hilang dan nilai regularisasi seperti pada persamaan berikut ini:

$$obj(\theta) = L(\theta) + \Omega(\theta)$$

Dengan:

L = Fungsi pelatihan yang hilang

Ω = Fungsi regularisasi

θ = Parameter model terkait

$$L(\theta) = \sum_{i=1}^n (y_i, \hat{y}_i)$$

Dengan:

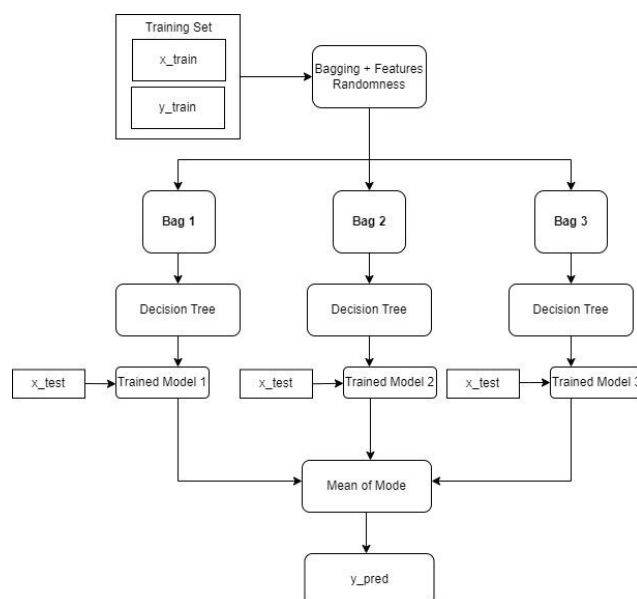
y_i = Nilai data sebenarnya yang dianggap benar

\hat{y} = Hasil nilai prediksi dari model

n = Jumlah iterasi nilai dari mode

Kelebihan dari algoritma *XGBoost* adalah sebagai berikut yaitu memiliki kinerja yang baik, algoritma *XGBoost* lebih stabil dalam penanganan data, dan mendukung *parallel processing* sehingga *training data* dan prediksi model dilakukan menjadi lebih cepat. Untuk kekurangan dari algoritma *XGBoost* adalah sebagai berikut yaitu *XGBoost* memiliki *hyperparameter* yang banyak untuk dikonfigurasi, sehingga membutuhkan waktu yang lama dan membutuhkan memori yang besar untuk melatih model.

2.1.4 Random Forest



Gambar 3 Flowchart Random Forest

Random forest berasal dari metode *decision tree* yang telah mengalami pengembangan. Dalam *random forest* sendiri memiliki kelebihan yaitu saat terdapat data outlier, *random forest* mampu meningkatkan akurasi prediksi. Kelebihan *random forest* yang lain adalah memiliki kemampuan seleksi fitur yang baik. Untuk kekurangan dari *random forest* sendiri adalah ketika dataset tidak seimbang maka prediksi yang dihasilkan juga tidak seimbang, pemrosesan analisis membutuhkan waktu yang cukup lama, dan rentan terhadap *overfitting*.

Metode ini digunakan untuk membangun pohon keputusan yang terdiri dari *root node*, *internal node*, dan *leaf node* dengan mengambil atribut dan data secara acak sesuai ketentuan yang diberlakukan. *Root node* merupakan simpul yang terletak paling atas, atau biasa disebut sebagai akar dari pohon keputusan. *Internal node* adalah simpul percabangan, dimana node ini mempunyai output minimal dua dan hanya ada satu input. Sedangkan *leaf node* atau terminal node merupakan simpul terakhir yang hanya memiliki satu input dan tidak mempunyai output.

Pohon keputusan dimulai dengan cara menghitung nilai entropy sebagai penentu tingkat ketidakmurnian atribut dan nilai information gain. Untuk menghitung nilai entropy dan nilai information gain dapat menggunakan persamaan di bawah ini:

$$Entropy(Y) = -\sum_i p(c|Y) \log_2 p(c|Y)$$

Dengan:

Y = Himpunan kasus
 $p(c|Y)$ = Proporsi nilai Y terhadap kelas c

$$Information\ Gain(Y, a) = Entropy(Y) - \sum_{v \in Values} \frac{|Y_v|}{|Y_a|} Entropy(Y_v)$$

Dengan:

$Values(a)$ = Merupakan semua nilai yang mungkin dalam himpunan kasus a
 Y_v = Subkelas dari Y dengan kelas v yang berhubungan dengan kelas a
 Y_a = Semua nilai yang sesuai dengan Y

2.1.5 Python

Python merupakan bahasa pemrograman tingkat tinggi yang dibuat oleh Guido Van Rossum dan dirilis pada tahun 1991. Python juga merupakan bahasa yang sangat populer belakangan ini. Selain itu, Python juga merupakan bahasa pemrograman yang multi fungsi contohnya Python dapat digunakan untuk Machine Learning dan Deep Learning. Python dipilih sebagai penelitian karena Python memiliki penulisan sintaksis yang mudah selain itu Python juga memiliki library yang lengkap dan memiliki dukungan komunitas yang kuat karena Python bersifat open source. Untuk menuliskan source code Python anda dapat menggunakan IDE seperti VS Code, Sublime Text, PyCharm atau juga dapat menggunakan IDE online seperti Jupyter Notebook dan Google Colab.

2.1.6 Confusion Matrix

Confusion Matrix merupakan pengukuran performa untuk klasifikasi machine learning dimana keluaran dapat berupa dua kelas atau lebih. Confusion matrix terdiri dari tabel yang menyatakan klasifikasi jumlah data uji yang benar dan jumlah data uji yang salah.

Ada empat istilah yang merupakan representasi hasil proses klasifikasi pada confusion matrix yaitu True Positif, True Negatif, False Positif, dan False Negatif. Berikut adalah ilustrasi untuk confusion matrix:

Tabel 1 Confusion Matrix

Confusion Matrix		Kelas Aktual	
		1 (Positive)	0 (Negative)
Kelas	1 (Positive)	TP	FP
Prediksi	0 (Negative)	FN	TN

Dengan:

- True Positive (TP) = Merupakan data positif yang diprediksi benar
- True Negative (TN) = Merupakan data negatif yang diprediksi benar
- False Positive (FP) = Merupakan data negatif namun diprediksi sebagai data positif
- False Negative (FN) = Merupakan data positif namun diprediksi sebagai data negatif

Keempat istilah dari confusion matrix dapat digunakan untuk menghitung berbagai performance metrics untuk mengukur kinerja model yang telah dibuat. Beberapa performance metrics yang umum digunakan adalah accuracy, precision, recall (sensitivity), specificity dan F1-score. Untuk menghitung nilai dari beberapa performance metrics dapat dilihat pada persamaan-persamaan di bawah ini:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad Precision = \frac{TP}{TP + FP}$$

$$Sensitivity = \frac{TP}{TP + FN} \quad Specificity = \frac{TN}{TN + FP}$$

$$F1 - Score = \frac{2 \times precision \times recall}{precision + recall}$$

Dengan menggabungkan penggunaan beberapa performance metrics ini, dapat diperoleh pemahaman yang lebih komprehensif tentang performa model klasifikasi, baik dari segi keseluruhan akurasi maupun kemampuan dalam mengidentifikasi kelas positif dan negatif secara tepat.

2.2 Penelitian Terkait

2.2.1 Prediksi Kepuasan Penumpang Maskapai Penerbangan dengan Algoritma Klasifikasi

Pada Penelitian yang dilakukan oleh Herawan Hayadi, Jin-Mook Kim, Khodijah Hulliyah dan Husni Teja Sukmana pada tahun 2021, membahas tentang prediksi kepuasan penumpang maskapai penerbangan dengan menggunakan beberapa metode klasifikasi yaitu k-Nearest, Logistics Regression, Decision Tree dan Random Forest. Dari hasil penelitian didapatkan hasil bahwa metode Random Forest menjadi metode dengan akurasi paling baik sebesar 99% untuk klasifikasi kepuasan penumpang maskapai penerbangan dibanding metode lainnya.

2.2.2 Klasifikasi Berita Menggunakan Metode Support Vector Machine

Pada penelitian yang dilakukan oleh Robbi Nanda, Elin Haerani, Siska Kurnia Gusti dan Siti Ramadhani pada tahun 2022, membahas tentang klasifikasi data berita di Indonesia dengan menggunakan metode Support Vector Machine (SVM). Dari hasil penelitian didapatkan hasil akurasi tertinggi pada skenario pembagian

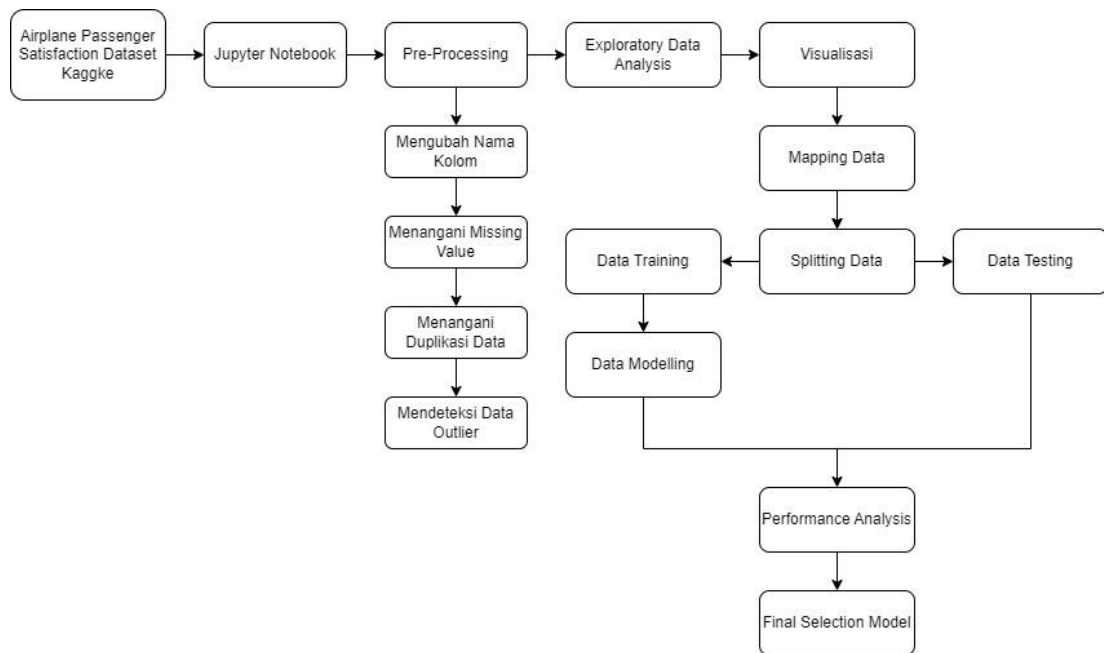
data 90% dan 10% yaitu sebesar 88% dengan data yang digunakan sebanyak 510 data berita.

2.2.3 Klasifikasi Pemegang Polis Menggunakan Metode XGBoost

Pada penelitian yang dilakukan oleh Aditya Adam Firdaus dan Aceng Komarudin Mutaqin pada tahun 2021, membahas tentang klasifikasi pemegang polis menggunakan metode XGBoost, dengan didapatkan hasil bahwa metode XGBoost dapat mengklasifikasikan klaim dengan akurat sebesar 80,87%. Sedangkan dalam hal presisi klasifikasi klaimnya sebesar 80,84%.

BAB III

METODOLOGI PENELITIAN



Gambar 4 Diagram Alur Penelitian

3.1 Pengumpulan Data

Pada penelitian ini dataset yang digunakan adalah dataset Airplane Passenger Satisfaction. Dataset ini merupakan hasil survey dari Negara Amerika Serikat atau US. Dataset tersebut diambil dari website terbuka bernama kaggle.com. Data yang tersedia adalah data tahun 2015 dan tahun 2019, akan tetapi setelah melalui penghitungan analisis dan mengetahui hasil dari analisis, maka dataset yang digunakan untuk menjadi acuan prediksi adalah dataset tahun 2019.

3.2 Pre-Processing

Pre-processing adalah sebuah proses yang dilakukan sebelum melaksanakan analisis dengan sebuah metode atau algoritma. Tujuan dari dilakukannya *pre-processing* adalah untuk menghasilkan data yang benar-benar bersih dan konsisten. *Pre-processing* terdiri dari beberapa langkah yaitu penghapusan beberapa kolom, mengubah nama kolom, mengatasi *missing value*, memeriksa duplikasi data, dan deteksi outlier.

3.2.1 Informasi Dataset

Informasi dataset adalah sebuah langkah yang digunakan untuk memeriksa type data pada tiap variabel dataset. Apakah telah sesuai dengan yang seharusnya atau bahkan tidak sesuai sehingga tipe data perlu diubah. Tujuan dilakukannya pengubahan tipe data adalah untuk mempermudah proses analisis dan mendapatkan data yang seragam atau konsisten.

3.2.2 Menghapus Kolom Id

Menghapus kolom id dilakukan karena pada kolom tersebut tidak memuat informasi penting mengenai topik dataset. Kolom tersebut hanya menyimpan id atau urutan data dari penumpang saja. Maka dari itu untuk menghindari error pada analisis, kolom tersebut perlu dihapus.

3.2.3 Mengubah Nama Kolom

Mengubah nama kolom dilakukan dengan menjadikan kata dari setiap kolom menjadi tidak kapital. Hal ini dilakukan untuk membantu memastikan konsistensi penggunaan nama kolom dalam dataset.

3.2.4 *Missing Value*

Missing value adalah baris dalam dataset yang tidak memiliki nilai atau kosong. *Missing value* sangat berpengaruh dalam hasil analisis data. Sehingga *missing value* perlu diatasi dengan tepat, yaitu dengan mengganti nilai kosong dengan rata-rata dari data atau dapat menghapus nilai kosong apabila frekuensinya tidak lebih dari 50%.

3.2.5 Duplikasi Data

Duplikasi data adalah kondisi dimana data tersebut muncul sebanyak lebih dari satu kali. Duplikasi data juga sangat berpengaruh dalam hasil analisis, karena algoritma dapat mendeteksi inkonsistensi data sehingga hasil analisis tidak maksimal. Untuk mengatasi duplikasi data dapat dilakukan dengan menghapus salah satu data yang terduplikasi.

3.2.6 Deteksi *Outlier*

Deteksi *outlier* adalah langkah dimana algoritma *machine learning* memeriksa apakah terdapat data yang memiliki perilaku atau nilai yang

menyimpang dari obyek-obyek lainnya. Data *outlier* dapat tetap digunakan untuk analisis ketika jumlah dari nilai *outlier* terlalu banyak dalam dataset. Hal tersebut mengindikasikan bahwa memang terdapat keanehan natural yang dihasilkan saat melakukan pengambilan atau observasi data. Dan data *outlier* juga dapat dibuang ketika data yang tersedia memiliki frekuensi yang besar dan jumlah dari nilai *outlier* kecil.

3.3 Exploratory Data Analysis

Exploratory data analysis atau EDA adalah sebuah langkah analisis dengan mendeskripsikan data melalui nilai rata-rata, nilai median, dan jumlah data. Fungsi dari EDA adalah untuk mendeteksi kesalahan yang terdapat pada data dan mengetahui hubungan antar data.

3.4 Visualisasi

Visualisasi data adalah langkah untuk merepresentasikan sebuah dataset dalam bentuk gambar seperti grafik dan diagram. Visualisasi dilakukan untuk mengetahui informasi lain dari dataset, mengidentifikasi permasalahan yang ada pada dataset, serta mengetahui pola dan tren dari dataset.

3.5 Linearitas

Linearitas digunakan untuk mengetahui hubungan antar dua variabel. Apakah dua variabel tersebut memiliki hubungan yang linear atau tidak. Tujuan dari memeriksa linearitas adalah untuk memilih variabel atau parameter yang akan digunakan dalam analisis selanjutnya.

3.6 Mapping

Mapping data digunakan untuk menumerisasi data kategorik menjadi data numerik. Tujuan dari mengubah data kategorik menjadi data numerik adalah untuk mempermudah analisis, karena algoritma tidak dapat memproses sebuah data yang berbentuk kategorik. Sehingga perlu diubah dalam *range* angka tertentu seperti mengubah data kategorik menjadi 0,1, dan 2.

3.7 Pemodelan Algoritma

Pemodelan algoritma adalah langkah yang digunakan untuk persiapan analisis menggunakan metode yang telah dipilih. Langkah pemodelan dilakukan

dengan membagi data menjadi variabel target dan atribut serta membagi dataset menjadi data training dan data testing. Setelah itu, melakukan inisiasi model menggunakan *support vector machine*, *random forest*, dan *xgboost*.

3.8 Feature Importance

Feature importance adalah sebuah metode yang diterapkan dalam analisis yang memiliki tujuan untuk mengidentifikasi sebuah variabel berkontribusi terhadap prediksi atau hasil model.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Dataset

Dataset yang akan dianalisis adalah dataset Airplane Passenger Satisfaction yang diambil dari Kaggle. Dataset ini memiliki jumlah baris sebanyak 129880 baris dan jumlah kolom adalah 24 kolom. Pada kolom 'id', dari awal sudah kita hapuskan karena tidak akan berpengaruh pada analisis maupun pembuatan model, sehingga terjadi perubahan jumlah kolom dari yang mulanya 24 kolom menjadi 23 kolom.

4.2 Pre-Processing

4.2.1 Mengganti Nama Kolom

```
#mengganti nama kolom

df = df.rename(columns={'satisfaction_v2':'satisfaction', 'Gender':'gender',
'Customer Type':'customer_type', 'Age':'age', 'Type of Travel':'travel_type',
'Class':'class', 'Flight Distance':'distance', 'Seat comfort':'seat_comfort',
'Departure/Arrival time convenient':'dep_val_time_convenient', 'Food and
drink':'food_drink', 'Gate location':'gate','Inflight wifi service':'wifi_service',
'Inflight entertainment':'entertainment','Online support':'online_support',
'Ease of Online booking':'online_booking_service','On-board
service':'onboard_service', 'Leg room service':'leg_room_service','Baggage
handling':'baggage_handling', 'Checkin
service':'checkin_service','Cleanliness':'cleanliness','Online
boarding':'online_boarding', 'Departure Delay in
Minutes':'departure_delay_minutes','Arrival Delay in
Minutes':'arrival_delay_minutes'})
```

Gambar di atas merupakan potongan script yang berfungsi untuk mengubah nama kolom agar saat memanggil atau mengolah kolom tersebut lebih mudah, terlebih lagi dataset memiliki banyak kolom dengan nama yang panjang. Sehingga dalam studi kasus ini, nama kolom diubah dengan nama yang lebih pendek.

4.2.2 Menangani *Missing Value*

Pada studi kasus ini, ditemukan 393 missing value pada kolom 'arrival_delay_minutes', sehingga terjadi perubahan terhadap jumlah data. Data yang sebelumnya sebanyak 129880 menjadi 129487. Missing value ini kami hapuskan, karena data kami berjumlah sangat banyak sehingga tidak berpengaruh secara signifikan.

```
df.isna().sum().sort_values(ascending=False)
```

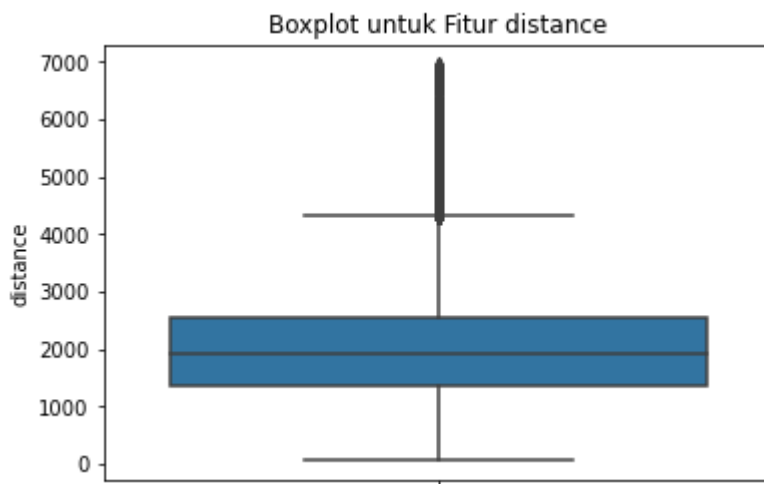
4.2.3 Menangani Duplikasi Data

Gambar di atas merupakan potongan script untuk memeriksa data yang terduplikasi. Pada dataset yang digunakan tidak ditemukan adanya duplikasi data, sehingga tidak ada perubahan terhadap jumlah data.

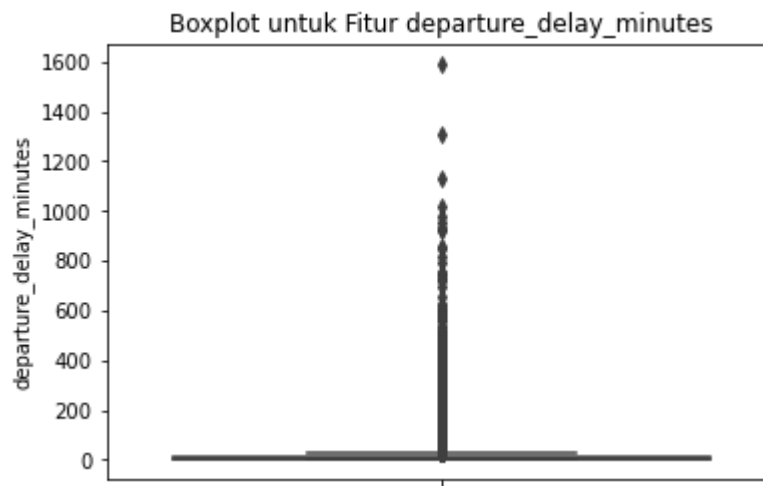
```
df.drop_duplicates(keep = 'first', inplace = True)
```

4.2.4 Mendeteksi Data *Outlier*

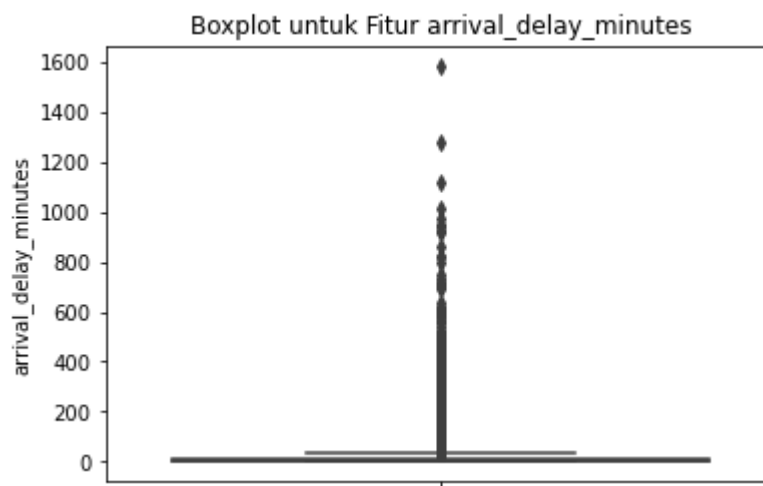
Outlier kita deteksi dengan visualisasi boxplot dan metode IQR. *Outlier* pada dataset ini ditemukan pada 3 kolom yaitu 'distance', 'departure_delay_minutes', dan 'arrival_delay_minutes'. Banyaknya outlier pada kolom 'distance' sebanyak 2575, pada kolom 'arrival_delay_minutes' sebanyak 17492, dan pada kolom 'departure_delay_minutes' sebanyak 17970. *Outlier-outlier* tersebut dipertahankan agar data yang digunakan untuk dianalisis dapat mewakili variasi yang sebenarnya dari data yang ada dan tidak menghilangkan informasi berharga.



Gambar 5 Boxplot dari Fitur Distance



Gambar 6 Boxplot dari Departure_Delay_Minutes



Gambar 7 Boxplot dari Arrival_Delay_Minutes

4.3 Exploratory Data Analysis (EDA)

Pada EDA didapatkan informasi baru yaitu rata-rata keterlambatan keberangkatan dan tiba pesawat adalah 15 menit dengan standar deviasinya 38. Median dari keterlambatannya adalah 0, artinya 50% penerbangan dalam data ini tidak mengalami keterlambatan

```
df.describe(exclude = ['float', 'int64']).T
```

	count	unique	top	freq
satisfaction	129487	2	satisfied	70882
gender	129487	2	Female	65703
customer_type	129487	2	Loyal Customer	105773
travel_type	129487	2	Business travel	89445
class	129487	3	Business	61990

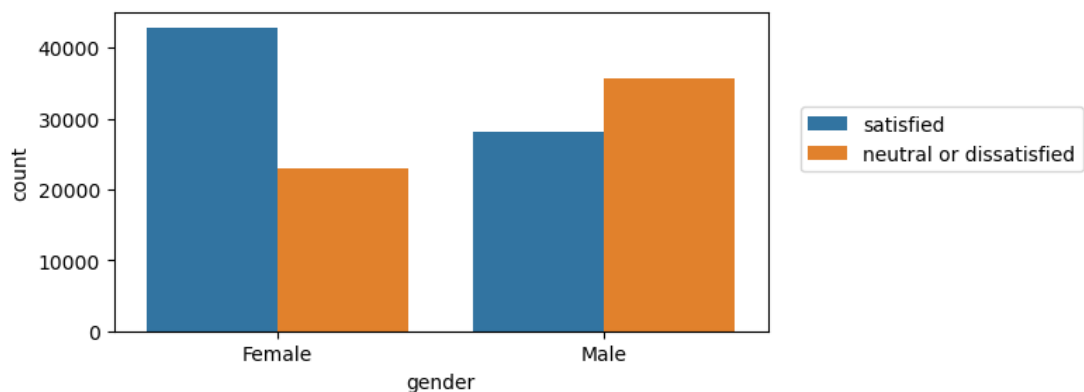
Gambar 8 Hasil Count Describe

	count	mean	std	min	25%	50%	75%	max
age	129487.0	39.428761	15.117597	7.0	27.0	40.0	51.0	85.0
distance	129487.0	1981.008974	1026.884131	50.0	1359.0	1924.0	2543.0	6951.0
seat_comfort	129487.0	2.838586	1.392873	0.0	2.0	3.0	4.0	5.0
dep_val_time_convenient	129487.0	2.990277	1.527183	0.0	2.0	3.0	4.0	5.0
food_drink	129487.0	2.852024	1.443587	0.0	2.0	3.0	4.0	5.0
gate	129487.0	2.990377	1.305917	0.0	2.0	3.0	4.0	5.0
wifi_service	129487.0	3.249160	1.318765	0.0	2.0	3.0	4.0	5.0
entertainment	129487.0	3.383745	1.345959	0.0	2.0	4.0	4.0	5.0
online_support	129487.0	3.519967	1.306326	0.0	3.0	4.0	5.0	5.0
online_booking_service	129487.0	3.472171	1.305573	0.0	2.0	4.0	5.0	5.0
onboard_service	129487.0	3.465143	1.270755	0.0	3.0	4.0	4.0	5.0
leg_room_service	129487.0	3.486118	1.292079	0.0	2.0	4.0	5.0	5.0
baggage_handling	129487.0	3.695460	1.156487	1.0	3.0	4.0	5.0	5.0
checkin_service	129487.0	3.340729	1.260561	0.0	3.0	3.0	4.0	5.0
cleanliness	129487.0	3.705886	1.151683	0.0	3.0	4.0	5.0	5.0
online_boarding	129487.0	3.352545	1.298624	0.0	2.0	4.0	4.0	5.0
departure_delay_minutes	129487.0	14.643385	37.932867	0.0	0.0	0.0	12.0	1592.0
arrival_delay_minutes	129487.0	15.091129	38.465650	0.0	0.0	0.0	13.0	1584.0

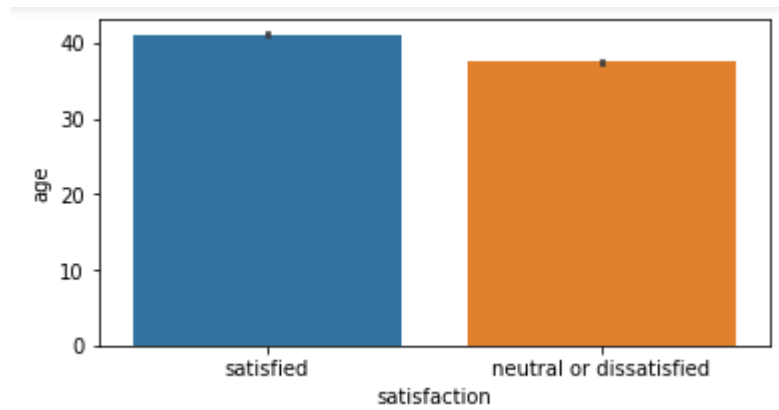
Gambar 9 Hasil dari Exploratory Data Analysis

4.4 Visualisasi

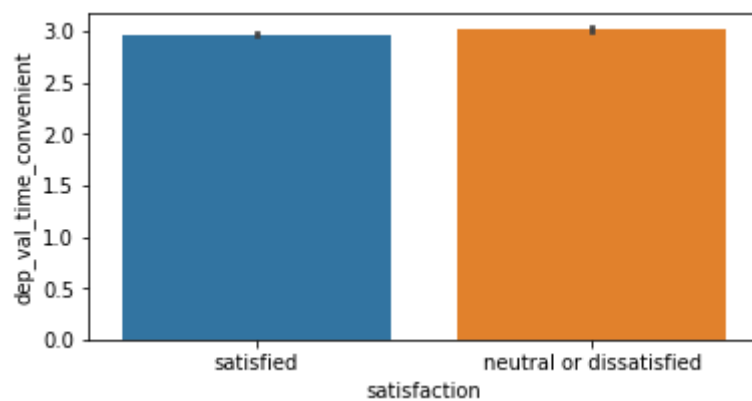
Pada studi kasus ini, ditemukan terdapat 4 fitur atau kolom yang ketika divisualisasikan berdasarkan data jenis numerikal, jumlah data pada kategori satisfied dan neutral or dissastified hampir sama. Hal ini berarti fitur atau kolom tersebut tidak terlalu mempengaruhi dan memberi banyak informasi. Sehingga kolom yang tidak terlalu memberikan banyak informasi, akan dilakukan penghapusan agar konsistensi data tetap terjaga.



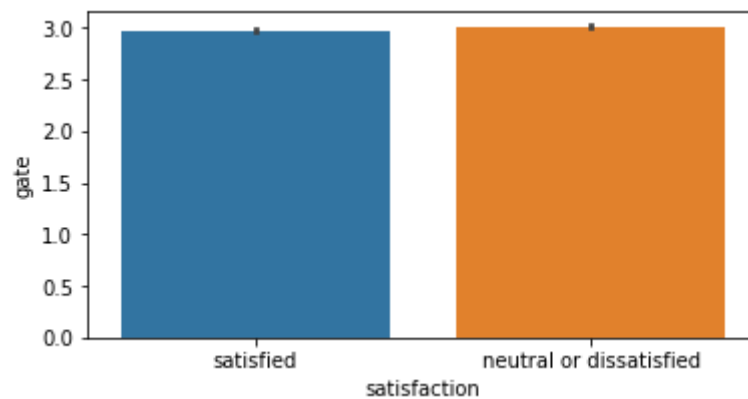
Gambar 10 Visualisasi Gender dengan Satisfaction_V2



Gambar 11 Visualisasi Satisfaction_V2 dengan Age

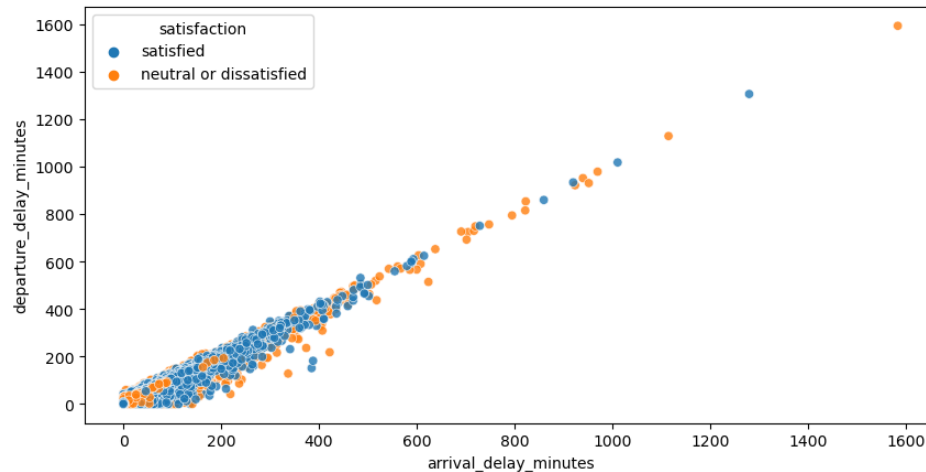


Gambar 12 Visualisasi Satisfaction_V2 dengan Dep_Val_Time_Convenient



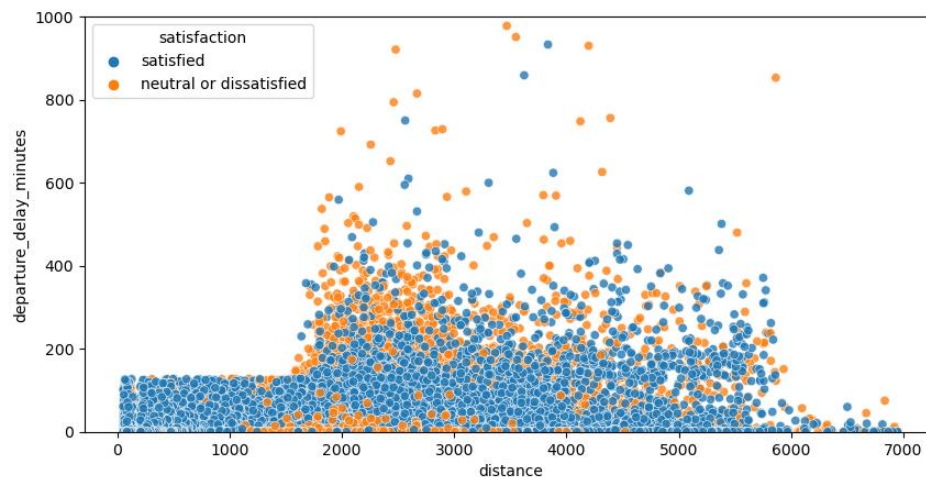
Gambar 13 Visualisasi Satisfaction_V2 dengan Gate

Keempat fitur atau kolom yang tidak banyak memberikan informasi dari dataset adalah kolom 'age', 'gender', 'gate', dan 'dep_val_time_convenient'. Maka dari itu, fitur-fitur atau kolom-kolom tersebut akan dihapuskan. Sehingga untuk dianalisis selanjutnya hanya terdapat 19 kolom.



Gambar 14 Scatter Plot Departure_Delay Minutes dan Arrival_Delay_Minutes

Gambar di atas merupakan visualisasi hubungan penundaan kedatangan dan keberangkatan terhadap kepuasan penumpang. Ternyata didapatkan informasi bahwa keterlambatan atau penundaan kedatangan dan keberangkatan, kepuasan penumpang memiliki hubungan yang linear. Linear disini berarti semakin lama penundaannya maka penumpang semakin merasa tidak puas atau dirugikan.



Gambar 15 Visualisasi Scatter Plot Distance dan Departure_Delay_Minutes

Sedangkan untuk hubungan keterlambatan keberangkatan terhadap jarak yang akan ditempuh didapatkan informasi bahwa semakin jauh jarak penerbangan, banyak penumpang tidak keberatan dengan penundaan sebentar dalam keberangkatan, sedangkan penumpang yang memiliki penerbangan jarak dekat tidak terlalu puas. Dari hal ini bisa diketahui bahwa penundaan keberangkatan bukan menjadi faktor kepuasan penumpang pesawat terbang.

4.5 Mapping Data

```
df['satisfaction'] = df['satisfaction'].map({'neutral or dissatisfied':0 ,
'satisfied':1})
df['customer_type'] = df['customer_type'].map({'Loyal Customer':1, 'disloyal
Customer':0})
df['travel_type'] = df['travel_type'].map({'Personal Travel':0, 'Business
travel':1})
df['class'] = df['class'].map({'Eco':0, 'Eco Plus':1, 'Business':2})
```

Gambar di atas merupakan potongan script untuk mapping atau mengkonversi data kategorik menjadi data numerik. Alasan lebih memilih melakukan mapping manual dibanding menumerasi data dengan `LabelEncoder()`, `OrdinalEncoder`, dan `OneHotEncoder` dikarenakan hasil pemetaan dengan fungsi-fungsi tersebut nilainya tidak tepat dan tidak sesuai dengan urutannya yang benar.

4.6 Splitting Data

```
target = 'satisfaction'
X = df.drop('satisfaction', axis=1)
y = df[target]
```

Pertama yang dilakukan adalah memisahkan data menjadi variabel target dan atribut. Variabel 'X' sebagai atribut yang berisi 19 kolom dan 'y' sebagai variabel target dengan 1 kolom yaitu kolom 'satisfaction'. Pembagian data ini kami lakukan dengan *percentage split*. Rasio yang digunakan ada 3 yaitu 90:10, 80:20, dan 70:30.

Pada rasio 90:10, jumlah data training sebanyak 116538 dan jumlah data testing sebanyak 12949. Sedangkan pada rasio 80:20, jumlah data training sebanyak 103589 dan jumlah data testingnya 25898. Dan pada rasio 70: 30, jumlah data trainingnya 90640 dan jumlah data testingnya 38847. Pada tahap pemisahan ini, juga dilakukan scaling fitur dengan metode pipeline, dan scaler standar. Tujuannya adalah untuk melakukan penskalaan fitur atau kolom dengan menggunakan metode `StandardScaler`.

4.7 Data Modelling

4.7.1 XGBoost

```
model_xgb = XGBClassifier(random_state =2)

model_xgb.fit(scaled_X_train,y_train)


pred_xgb = model_xgb.predict(scaled_X_test)

accuracy_score(y_test,pred_xgb)
```

Di atas, disajikan kode untuk melakukan klasifikasi menggunakan algoritma XGBoost dengan menggunakan library XGBClassifier. Langkah pertama adalah mendefinisikan model dengan variabel bernama model_xgb yang kemudian dilatih menggunakan data pelatihan, scaled_X_train dan y_train. Setelah itu, model akan diuji menggunakan data pengujian, scaled_X_test dan y_test. Untuk mengevaluasi kinerja model, digunakan metode Confusion Matrix.

4.7.2 Support Vector Machine

```
model_svm = SVC(random_state =2)
model_svm.fit(scaled_X_train,y_train)


pred_svm = model_svm.predict(scaled_X_test)
accuracy_score(y_test,pred_svm)
```

Di atas, disajikan kode untuk melakukan klasifikasi menggunakan algoritma SVM dengan menggunakan library SVC. Langkah pertama adalah mendefinisikan model dengan variabel bernama model_svm yang kemudian dilatih menggunakan data pelatihan, scaled_X_train dan y_train. Setelah itu, model akan diuji menggunakan data pengujian, scaled_X_test dan y_test. Untuk mengevaluasi kinerja model, digunakan metode Confusion Matrix.

4.7.3 Random Forest

```
model_rf = RandomForestClassifier(random_state =2)
model_rf.fit(scaled_X_train,y_train)

pred_rf = model_rf.predict(scaled_X_test)
accuracy_score(y_test,pred_rf)
```

Di atas, disajikan kode untuk melakukan klasifikasi menggunakan algoritma RandomForest dengan menggunakan library RandomForestClassifier. Langkah pertama adalah mendefinisikan model dengan variabel bernama model_rf yang kemudian dilatih menggunakan data pelatihan, scaled_X_train dan y_train. Setelah itu, model akan diuji menggunakan data pengujian, scaled_X_test dan y_test. Untuk mengevaluasi kinerja model, digunakan metode Confusion Matrix.

4.8 Performance Analysis

4.8.1 Evaluasi Performa XGBoost

```
conf_matrix = confusion_matrix(y_test, pred_xgb)

sns.heatmap(conf_matrix, annot=True, xticklabels=['not admitted', 'admitted'],
yticklabels=['not admitted', 'admitted'])

plt.figure(figsize=(5,5))

plt.show()
```

Gambar di atas adalah script untuk mencetak visualisasi confusion matrix dengan heatmap. Confusion matrix dihitung oleh fungsi confusion_matrix dengan parameter y_test dan pred_xgb. Confusion matrix didefinisikan sebagai variabel conf_matrix. Label pada sumbu x dan y heatmap, yaitu 'not admitted' dan 'admitted', menunjukkan kelas yang diprediksi oleh model. 'not admitted' menyatakan kelas negatif, sedangkan 'admitted' merupakan kelas positif.

```
# Menghitung metrik evaluasi

TP = conf_matrix[1, 1]

FP = conf_matrix[0, 1]

TN = conf_matrix[0, 0]

FN = conf_matrix[1, 0]


print('TP:', TP)

print('FP:', FP)

print('TN:', TN)

print('FN:', FN)


accuracy = (TP + TN) / (TP + FP + TN + FN)

precision = TP / (TP + FP)

recall = TP / (TP + FN)

f1_score = 2 * (precision * recall) / (precision + recall)


# Mencetak metrik evaluasi

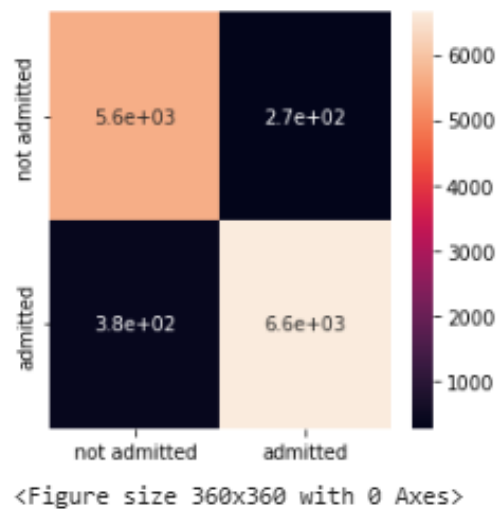
print('Accuracy:', accuracy)

print('Precision:', precision)
```

```
print('Recall:', recall)

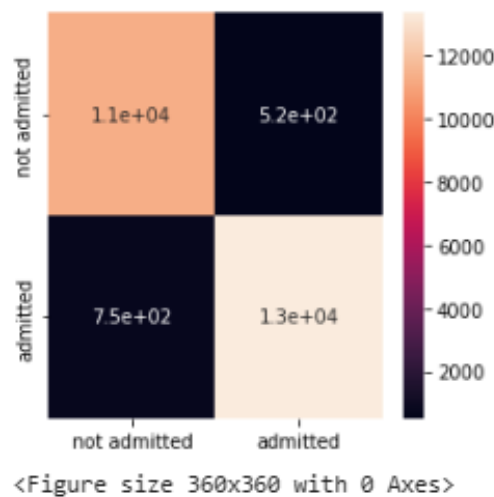
print('F1 Score:', f1_score)
```

Selanjutnya yang dapat kita ambil insightnya dari script di atas adalah dilakukan perhitungan matriks evaluasi dari hasil confusion matrix yang telah didefinisikan sebelumnya. Pada rasio 90:10 didapatkan hasil accuracy sebesar 0.9497, precision sebesar 0.9604, recall sebesar 0.9463, dan F1 score sebesar 0.9533.

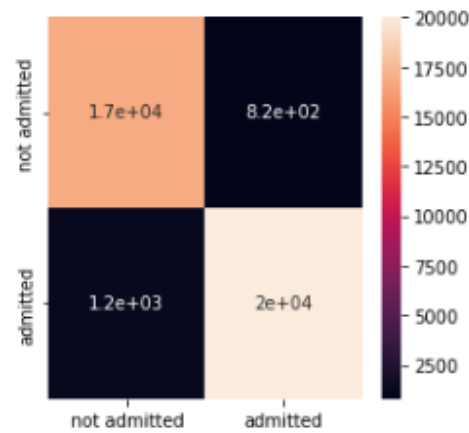


Gambar 16 Confusion Mtarix XGBoost 90:10

Sedangkan pada rasio 80:20, didapatkan hasil accuracy sebesar 0.9511, precision sebesar 0.9625, recall sebesar 0.9468, dan F1 score sebesar 0.9546. Dan yang terakhir yaitu rasio 70:30 didapatkan hasil accuracy sebesar 0.9481, precision sebesar 0.9606, recall sebesar 0.9436, dan F1 score sebesar 0.9520.



Gambar 17 Confusion Matrix XGBoost 80:20



<Figure size 360x360 with 0 Axes>

Gambar 18 Confusion Matrix XGBoost 70:30

Setelah dilakukan pengecekan dengan menggunakan *confusion matrix*, selanjutnya adalah memeriksa nilai *error* pada masing-masing model dengan menggunakan *codescript* berikut:

```
err_train = np.mean(y_train != model_xgb.predict(X_train))

err_test = np.mean(y_test != model_xgb.predict(X_test))
```

Didapatkan hasil sebagai berikut:

Tabel 2 Perbandingan Error Tiap Rasio

Rasio	Nilai Error Data Training	Nilai Error Data Testing
90:10	0.4520671368995521	0.45733261255695423
80:20	0.45163096467771674	0.45163096467771674
70:30	0.45196381288614296	0.4540633768373362

4.8.2 Evaluasi Performa SVM

```
conf_matrix = confusion_matrix(y_test, pred_svm)

sns.heatmap(conf_matrix, annot=True, xticklabels=['not admitted', 'admitted'],
yticklabels=['not admitted', 'admitted'])

plt.figure(figsize=(5,5))
plt.show()
```

Gambar di atas merupakan script untuk mencetak visualisasi confusion matrix dengan heatmap. Confusion matrix dihitung oleh fungsi `confusion_matrix` dengan parameter `y_test` dan `pred_xgb`. Confusion matrix didefinisikan sebagai variabel `conf_matrix`. Label pada sumbu x dan y heatmap, yaitu 'not admitted' dan 'admitted', label tersebut menunjukkan kelas yang diprediksi oleh model. 'not admitted' menyatakan kelas negatif, sedangkan 'admitted' menyatakan kelas positif.

```
# Menghitung metrik evaluasi

TP = conf_matrix[1, 1]

FP = conf_matrix[0, 1]

TN = conf_matrix[0, 0]

FN = conf_matrix[1, 0]


print('TP:', TP)

print('FP:', FP)

print('TN:', TN)

print('FN:', FN)


accuracy = (TP + TN) / (TP + FP + TN + FN)

precision = TP / (TP + FP)

recall = TP / (TP + FN)

f1_score = 2 * (precision * recall) / (precision + recall)
```

```
# Mencetak metrik evaluasi

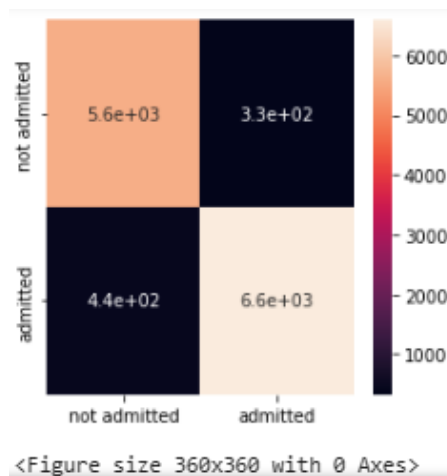
print('Accuracy:', accuracy)

print('Precision:', precision)

print('Recall:', recall)

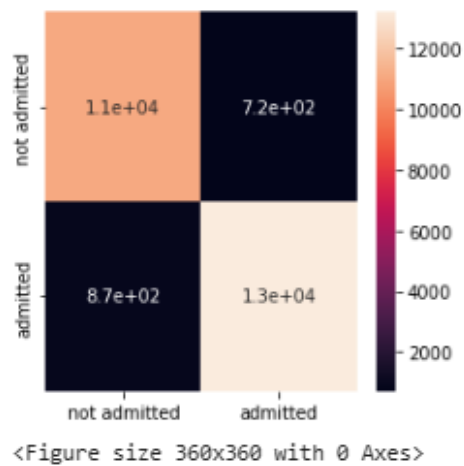
print('F1 Score:', f1_score)
```

Selanjutnya yang bisa dilihat dari script di atas, dilakukan perhitungan metrik evaluasi dari hasil confusion matrix yang telah didefinisikan sebelumnya. Pada rasio 90:10 didapatkan hasil accuracy sebesar 0.9406, precision sebesar 0.9524, recall sebesar 0.9373, dan F1 score sebesar 0.9448.

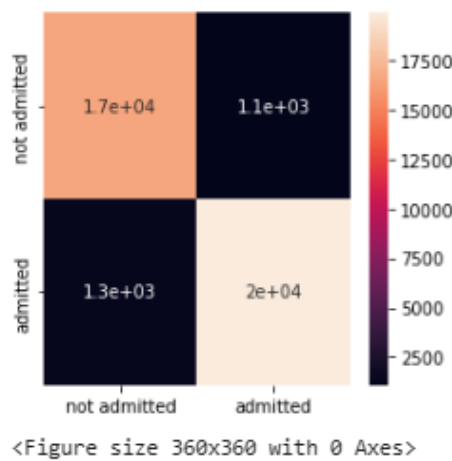


Gambar 19 Confusion Matrix SVM 90:10

Sedangkan pada rasio 80:20, didapatkan hasil accuracy sebesar 0.9386, precision sebesar 0.9484, recall sebesar 0.9380, dan F1 score sebesar 0.9432. Dan yang terakhir yaitu rasio 70:30 didapatkan hasil accuracy sebesar 0.9381, precision sebesar 0.9483, recall sebesar 0.9377, dan F1 score sebesar 0.9430.



Gambar 20 Confusion Matrix SVM 80:20



Gambar 21 Confusion Matrix SVM 70:30

Setelah dilakukan pengecekan dengan menggunakan *confusion matrix*, selanjutnya adalah memeriksa nilai *error* pada masing-masing model dengan menggunakan *codescript* berikut:

```
err_train = np.mean(y_train != model_xgb.predict(X_train))
err_test = np.mean(y_test != model_xgb.predict(X_test))
```

Didapatkan hasil sebagai berikut:

Tabel 3 Perbandingan Error Tiap Rasio

Rasio	Nilai Error Data Training	Nilai Error Data Testing
90:10	0.4520671368995521	0.45733261255695423

80:20	0.45163096467771674	0.4564445130898139
70:30	0.45196381288614296	0.4540633768373362

4.8.3 Evaluasi Performa Random Forest

```

conf_matrix = confusion_matrix(y_test, pred_rf)

sns.heatmap(conf_matrix, annot=True, xticklabels=['not admitted', 'admitted'],
yticklabels=['not admitted', 'admitted'])

plt.figure(figsize=(5,5))
plt.show()

```

Gambar di atas adalah script untuk mencetak visualisasi confusion matrix dengan heatmap. Confusion matrix dihitung oleh fungsi `confusion_matrix` dengan parameter `y_test` dan `pred_xgb`. Confusion matrix didefinisikan sebagai variabel `conf_matrix`. Label pada sumbu x dan y heatmap, yaitu 'not admitted' dan 'admitted', menunjukkan kelas yang diprediksi oleh model. 'not admitted' menyatakan kelas negatif, sedangkan 'admitted' merupakan kelas positif.

```

# Menghitung metrik evaluasi

TP = conf_matrix[1, 1]

FP = conf_matrix[0, 1]

TN = conf_matrix[0, 0]

FN = conf_matrix[1, 0]

print('TP:', TP)

print('FP:', FP)

```

```
print('TN:', TN)

print('FN:', FN)


accuracy = (TP + TN) / (TP + FP + TN + FN)

precision = TP / (TP + FP)

recall = TP / (TP + FN)

f1_score = 2 * (precision * recall) / (precision + recall)


# Mencetak metrik evaluasi

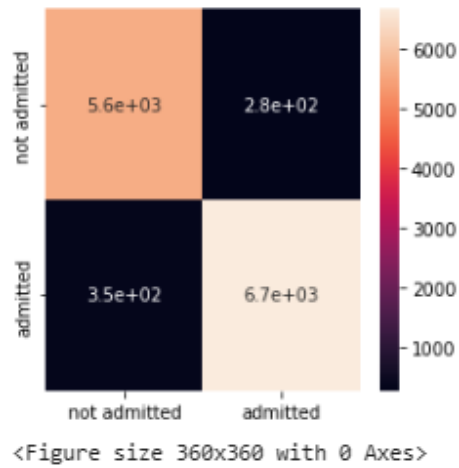
print('Accuracy:', accuracy)

print('Precision:', precision)

print('Recall:', recall)

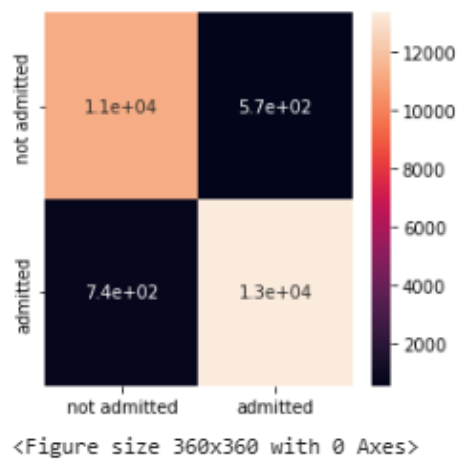
print('F1 Score:', f1_score)
```

Selanjutnya yang bisa dilihat dari script di atas, dilakukan perhitungan metrik evaluasi dari hasil confusion matrix yang telah didefinisikan sebelumnya. Pada rasio 90:10 didapatkan hasil accuracy sebesar 0.9508, precision sebesar 0.9590, recall sebesar 0.9500, dan F1 score sebesar 0.9545.



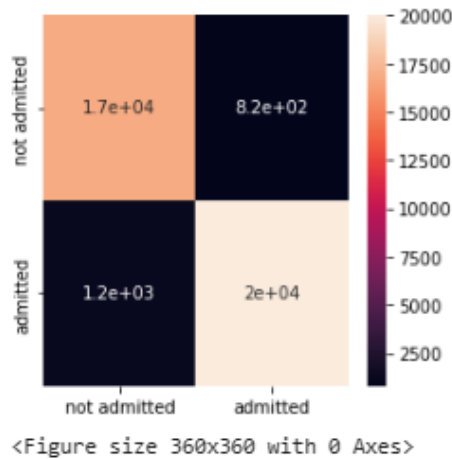
Gambar 22 Confusion Matrix Random Forest 90:10

Sedangkan pada rasio 80:20, didapatkan hasil accuracy sebesar 0.9494, precision sebesar 0.9590, recall sebesar 0.9475, dan F1 score sebesar 0.9532.



Gambar 23 Confusion Matrix Random Forest 80:20

Dan yang terakhir yaitu rasio 70:30 didapatkan hasil accuracy sebesar 0.9487, precision sebesar 0.9594, recall sebesar 0.9461, dan F1 score sebesar 0.9527.



Gambar 24 Confusion Matrix Random Forest 70:30

Setelah dilakukan pengecekan dengan menggunakan *confusion matrix*, selanjutnya adalah memeriksa nilai *error* pada masing-masing model dengan menggunakan *codescript* berikut:

```
err_train = np.mean(y_train != model_svm.predict(X_train))

err_test = np.mean(y_test != model_svm.predict(X_test))
```

Didapatkan hasil sebagai berikut:

Tabel 4 Perbandingan Error Tiap Rasio

Rasio	Nilai Error Data Training	Nilai Error Data Testing
90:10	0.4519984897629958	0.4572553865163333
80:20	0.45163096467771674	0.4564445130898139
70:30	0.45196381288614296	0.4540633768373362

4.9 Pemilihan Akhir Model

Tabel 5 Pemilihan Akhir Model

Algoritma	Akurasi Model		
	Rasio 90:10	Rasio 80:20	Rasio 70:30
XGBoost	94,97%	95,11%	94,81%
SVM	94,06%	93,86%	93,81%
Random Forest	95,08%	94,94%	94,87%

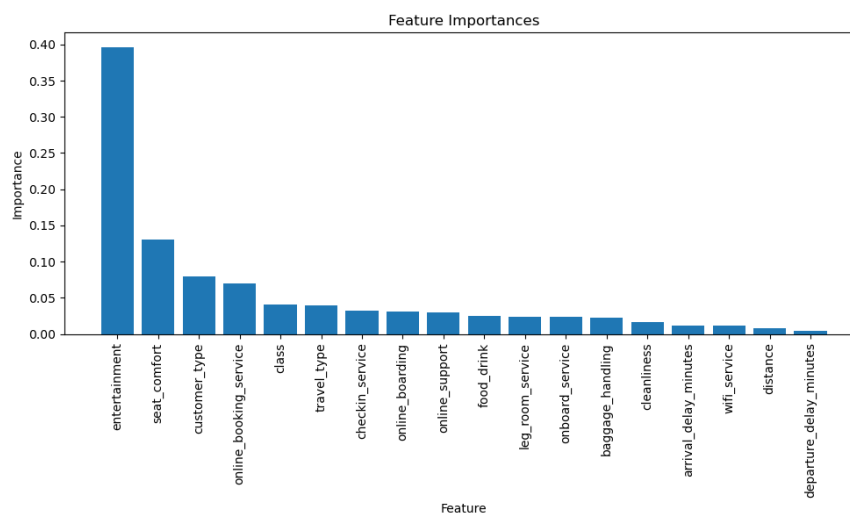
Tabel 6 Perbandingan Nilai Error Rasio Tiap Model

Model	Rasio	Data Testing
XGBoost	90:10	0.45733261255695423
	80:20	0.45163096467771674
	70:20	0.4540633768373362
SVM	90:10	0.45733261255695423
	80:20	0.4564445130898139
	70:20	0.4540633768373362
Random Forest	90:10	0.4572553865163333

	80:20	0.4564445130898139
	70:20	0.4540633768373362

Dari Hasil Perbandingan yang ditunjukkan pada tabel 1, didapatkan perbedaan hasil akurasi yang tidak terlalu signifikan. Diperoleh bahwa metode XGBoost merupakan metode terbaik pada dataset ini karena mampu mencapai nilai akurasi tertinggi di antara ketiga metode lainnya dengan nilai akurasi sebesar 95,11%. Metode XGBoost juga memberikan kinerja terbaik berdasarkan nilai precision, recall, dan F1-score yang di mana merupakan nilai terbesar di antara metode klasifikasi lainnya. Selain itu, jika dilihat pada table 7, model XGBoost memiliki nilai error terendah dengan nilai 0,451. Hal tersebut juga menandakan bahwa XGBoost memiliki performa yang baik dalam melakukan prediksi. Sedangkan metode SVM memiliki akurasi paling rendah sebesar 93,86% yang merupakan nilai terendah di antara ketiga metode klasifikasi lainnya.

4.10 Fitur Penting Model



Gambar 25 Feature Importances

Dari hasil *bar plot* di atas, dapat diketahui bahwa fitur “entertainment” memiliki pengaruh yang besar, diikuti dengan “seat_comfort”, “customer_type”, dan “online_booking_service”. Sedangkan “wifi_service”, “distance”, dan “departure_delay_minutes” tidak memiliki banyak pengaruh dalam prediksi model XGBoost.

BAB V

KESIMPULAN

Industri penerbangan mengalami pertumbuhan yang pesat dengan penumpang yang terus meningkat di seluruh dunia. Maskapai penerbangan tentu berlomba-lomba memberikan pengalaman terbaik kepada penumpang mereka agar mempertahankan pelanggan, memperluas pangsa pasar, dan membangun citra positif bagi maskapai penerbangan yang kuat. Untuk mencapai hal tersebut, dibutuhkan analisis komparatif untuk menemukan metode klasifikasi yang paling tepat terhadap kepuasan penumpang pesawat terbang.

Hasil algoritma XGBoost pada rasio 80:20 memiliki akurasi sebesar 95,11% sedangkan Support Vector Machine (SVM) sebesar 93,86%, dan Random Forest sebesar 94,94%. Walaupun didapatkan perbedaan hasil akurasi yang tidak terlalu signifikan, dari hasil perbandingan, algoritma XGBoost lebih unggul dibandingkan yang lain. Algoritma XGBoost merupakan algoritma terbaik dalam membangun model klasifikasi kepuasan penumpang pesawat terbang karena mampu mencapai nilai akurasi tertinggi di antara ketiga metode lainnya dengan nilai akurasi sebesar 95,11%.

DAFTAR PUSTAKA

- Alfarizi, M. S., Al-Farish, M. Z., Taufiqurrahman, M., Ardiansah, G., & Elgar, M. (2023). Penggunaan Python sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning.
- Haryadi, B. H., Kim, J.-M., Hulliyah, K., & Sukmana, H. T. (2021). Predicting Airline Passenger Satisfaction with Classification Algorithms. *Journal of Informatics and Information System*.
- Normawati, D., & Prayogi, S. A. (2021). Implementasi Naive Bayes Classifier dan Confusion Matrix pada Analisis Sentimen Berbasis Teks pada Twitter. *Jurnal Sains Komputer & Informatika*.
- Pangestu, S. Y., Astuti, Y., & Farida, L. D. (2019). Algoritma Support Vector Machine untuk Klasifikasi Sikap Politik Terhadap Partai Politik Indonesia. *Jurnal Mantik Penusa*.
- Ritonga, A. S., & Purwaningsih, E. S. (2018). Penerapan Metode Support Vector Machine (SVM) dalam Klasifikasi Kualitas Pengelasan SMAW (Shield Metal ARC Welding). *Jurnal Ilmiah Edutic*.
- Siburian, V. W., & Mulyana, I. E. (2018). Prediksi Harga Ponsel Menggunakan Metode Random Forest. *Prosiding Annual Research Seminar 2018*.
- Supriyadi, R., Gata, W., Maulidah, N., & Fauzi, A. (2020). Penerapan Algoritma Random Forest untuk Menentukan Kualitas Anggur Merah. *Jurnal Ilmiah Ekonomi dan Bisnis*.
- Yulianti, S. E., Soesanto, O., & Sukmawaty, Y. (2022). Penerapan Metode Extreme Gradient Boosting (XGBoost) pada Klasifikasi Nasabah Kartu Kredit. *Journal of Mathematics*.

LAMPIRAN

Dataset Airplane Passenger Satisfaction

	id	satisfaction_v2	Gender	Customer Type	Age	Type of Travel	Class	Flight Distance	Seat comfort	Departure/Arrival time convenient	...	Online support
0	11112	satisfied	Female	Loyal Customer	65	Personal Travel	Eco	265	0	0	...	2
1	110278	satisfied	Male	Loyal Customer	47	Personal Travel	Business	2464	0	0	...	2
2	103199	satisfied	Female	Loyal Customer	15	Personal Travel	Eco	2138	0	0	...	2
3	47462	satisfied	Female	Loyal Customer	60	Personal Travel	Eco	623	0	0	...	3
4	120011	satisfied	Female	Loyal Customer	70	Personal Travel	Eco	354	0	0	...	4

5 rows × 24 columns

Dataset Airplane Passenger Satisfaction

Online support	Ease of Online booking	On-board service	Leg room service	Baggage handling	Checkin service	Cleanliness	Online boarding	Departure Delay in Minutes	Arrival Delay in Minutes
2	3	3	0	3	5	3	2	0	0.0
2	3	4	4	4	2	3	2	310	305.0
2	2	3	3	4	4	4	2	0	0.0
3	1	1	0	1	4	1	3	0	0.0
4	2	2	0	2	4	2	5	0	0.0