Perbandingan Skor Prediksi dalam Pembelajaran Mesin pada Keselamatan Penumpang Kapal

Mochamad Althaf Pramasetya Perkasa[1],Alam Rahmatulloh[2]
[1,2] Universitas Siliwangi, Kota Tasikmalaya
Email : 207006080@student.unsil.ac.id[1], alam@unsil.ac.id[2]

Abstrak

Keselamatan penumpang kapal merupakan aspek penting dalam industri perkapalan, dan prediksi yang tepat mengenai situasi darurat dan keselamatan dapat memberikan keuntungan besar dalam mengelola risiko di laut. Prediksi yang tepat mengenai situasi darurat dan keselamatan dapat memberikan keuntungan besar dalam mengelola risiko di laut. Penelitian ini melibatkan pengumpulan dan analisis data historis mengenai insiden kecelakaan kapal dan faktor-faktor yang mempengaruhinya. Metode pembelajaran mesin, seperti logistic regression, KNN, Naïve Bayes Classifier, Decision Tree, Random Forest, dll. akan digunakan untuk mengidentifikasi pola-pola dalam data yang dapat digunakan untuk memprediksi keselamatan penumpang kapal. Data yang digunakan dalam penelitian ini mencakup variabel-variabel seperti cuaca, keadaan kapal, jumlah penumpang, dan faktor-faktor lain yang relevan. Data penelitian diambil dari penumpang titanic. Hasil penelitian ini menghasilkan model yang cocok untuk dijadikan prediksi keselamatan penumpang kapal, yaitu model random forest dan decision tree mendapatkan skor tertinggi dibandingkan model yang lainnya, skor prediksinya yaitu 84.29%.

Kata Kunci: Keselamatan Penumpang**,** Pembelajaran Mesin, Prediksi

PENDAHULUAN

Keselamatan penumpang kapal telah menjadi isu utama yang semakin mendapat perhatian. Kecelakaan di laut dapat berdampak serius, baik dalam hal hilangnya nyawa maupun dampak ekonomi.Oleh karena itu, menjadi penting untuk mengembangkan metode yang dapat membantu memprediksi keselamatan penumpang kapal dengan tingkat akurasi dan efektivitas yang tinggi(Rahmanita et al., 2023). Dataset diambil dari sebuah tragedi, yaitu titanic. Penggunaan dataset titanic ini dapat dijadikan parameter untuk mengetahui faktor yang dapat mempengaruhi keselamatan penumpang.

Metode pembelajaran mesin digunakan untuk membuat prediksi mengenai penumpang yang selamat pada saat Titanic tenggelam(Mosavi et al., 2018). Fitur-fitur seperti tarif tiket, usia, jenis kelamin, dan kelas akan digunakan untuk membuat prediksi-prediksi tersebut. Analisis prediktif adalah suatu prosedur yang mencakup penggunaan metode komputasi untuk menentukan pola-pola penting dan berguna dalam data besar. Dengan menggunakan metode pembelajaran mesin, kelangsungan hidup diprediksi pada berbagai kombinasi fitur(A. Singh et al., 2017).

Beberapa model pada pembelajaran mesin akan digunakan untuk membangun proses prediksi dataset, seperti model logistic regression, KNN, Naïve Bayes Classifier, Decision Tree, Random Forest, dll(Sarker, 2021) . Logistic regression memiliki kelebihan yaitu memiliki kelebihan yaitu mudah diinterpretasi, cocok untuk masalah klasifikasi biner, dan memiliki kelemahan yaitu terbatas pada kasus klasifikasi biner, tidak mampu menangani hubungan kompleks antar fitur(Ashenden, 2021). KNN memiliki kelebihan yaitu mudah diimplementasikan, efektif dalam data yang tidak linier , dapat digunakan untuk klasifikasi atau regresi, sedangkan kelemahannya sensitif terhadap data pencilan, memerlukan perhitungan yang intensif jika dataset besar, dan

membutuhkan pemilihan parameter k(Kenyhercz & Passalacqua, 2016). Kelebihan SVM yaitu cocok untuk dataset dengan jumlah fitur tinggi, efektif dalam menangani data yang tidak terpisah secara linier, dapat menggunakan fungsi kernel dan kelemahannya yaitu memerlukan tuning parameter yang tepat, rentan terhadap *overheating* jika tidak diatur dengan baik(Pratiwi & Setyawan, 2021). Naive Bayes Classifier adalah metode yang digunakan dengan cepat untuk mengelompokkan data teks dan data berkategori, walaupun asumsi bahwa setiap fitur dalam data bersifat independen tidak selalu benar(de Souza et al., 2022). Decision Tree mudah diinterpretasi dan mampu mengatasi berbagai jenis data, namun cenderung mengalami overfitting (Sanni et al., 2022). Random Forest mengatasi overfitting dengan menggabungkan banyak pohon keputusan, meskipun kompleks dan sulit diinterpretasi . Perceptron sederhana dan berkinerja baik pada data linier, tetapi terbatas pada masalah yang kompleks(Widrow & Lehr, n.d.). Stochastic Gradient Descent efisien pada data besar, meskipun rentan terhadap data tidak seimbang(Sharma, 2018). RVM (Relevance Vector Machine) efektif dalam mengatasi *overfitting* dan mengidentifikasi fitur yang relevan, meskipun memerlukan waktu komputasi yang lebih lama dan kurang umum digunakan (Tipping, 1999). Kelebihan dan kelemahan dari masing-masing model tersebut akan terjawab nantinya pada hasil perbandingan skor prediksi, dengan melihat model mana yang cocok untuk dijadikan bahan prediksi keselamatan penumpang kapal.

Penelitian (Kakde & Agrawal, 2018) membandingkan beberapa model pembelajaran mesin dengan menggunakan EDA yaitu proses analisis data yang digunakan untuk memahami karakteristik, pola, dan struktur dari data sebelum melakukan analisis statistik yang lebih mendalam. model yang dipilih dalam penelitian (Kakde & Agrawal, 2018)diantaranya logistic regression, random forest, decision tree, dan SVM. Setelah melakukan hasil analisis menggunakan EDA, pada didapat skor prediksi diantaranya logistic regression mendapatkan nilai 0,832761504, kemudian random forest sebesar 0.826031816, selanjutnya decision tree sebesar 0.817059484, dan yang terakhir yaitu model SVM dengan skor 0.831613817. Model terbaik yang diperoleh dari penelitian (Kakde & Agrawal, 2018)adalah SVM.

Penelitian ini akan menambahkan beberapa model dalam pembelajaran mesin yang sebelumnya tidak dimasukkan pada penelitian (Kakde & Agrawal, 2018) dan membuat perbandingan skor prediksinya. Penelitian ini berbeda dengan sebelumnya, penelitian ini tidak menggunakan R studio melainkan dibangun diatas python. Perbedaan penggunaan python akan sangat terasa, karena banyak berbagai library yang dapat mendukung proses prediksi keselamatan penumpang. Penggunaan python akan menghasilkan skor yang bentuknya konstan, tidak berupa decimal. Masing-masing fitur yang dimiliki dari model-model yang dipakai akan mempengaruhi nilai skor prediksi keselamatan. Hal yang berbeda juga yaitu pada penggunaan metode, penelitian ini menambahkan metode visualisasi data pada tahapan proses prediksi datasetnya. Peneliti melihat celah pada penelitian sebelumnya, celah yang akan diperbaiki yaitu pada hasil skor prediksi menggunakan algoritma-algoritma yang ada pada pembelajaran mesin. Penelitian ini membuat skor prediksi dengan nilai persen, pada penelitian sebelumnya hasil skor prediksi menunjukan nilai desimal.

METODE

Metode yang digunakan pada penelitian ini yaitu menggunakan EDA. Analisis Data Eksploratori (Exploratory Data Analysis - EDA) adalah suatu pendekatan yang digunakan dalam statistik dan ilmu data untuk mengungkap struktur dan pola dalam dataset. EDA membantu

Anda memahami data yang Anda miliki sebelum menerapkan model atau statistik yang lebih lanjut. Visualisasi data adalah salah satu komponen penting dari EDA, karena grafik dan plot dapat membantu peneliti memahami data dengan lebih baik(J. Singh et al., 2022).

Metode visualisasi data termasuk pada bagian tahapan EDA. Metode visualisasi data menggunakan *python* dalam kasus prediksi dataset Titanic, memiliki beberapa kegunaan yang sangat penting. Pertama, visualisasi data memungkinkan kita untuk memahami karakteristik dan struktur dataset dengan lebih baik. Dengan melihat grafik dan plot visual, kita dapat mengidentifikasi pola, anomali, atau tren yang mungkin ada dalam data, seperti distribusi usia penumpang, hubungan antara kelas tiket dan kelangsungan hidup, atau sebaran data pada berbagai fitur. Selanjutnya, visualisasi data memungkinkan kita untuk membuat keputusan yang lebih baik dalam pemrosesan data. Misalnya, kita dapat menggunakan visualisasi untuk mengidentifikasi data yang hilang atau anomali yang perlu diatasi sebelum membangun model pembelajaran mesin. Hal ini membantu meningkatkan kualitas data yang akan digunakan dalam proses pembelajaran mesin, yang pada gilirannya dapat meningkatkan kinerja model.

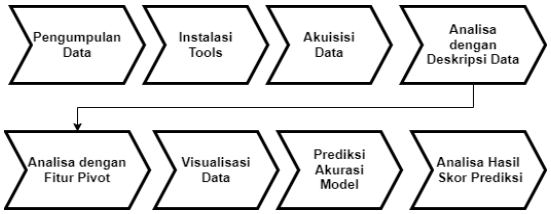
Penelitian Terkait

Penelitian (Kakde & Agrawal, 2018) melakukan penerapan prediksi menggunakan model-model yang ada pada pembelajaran mesin, kemudian dibangun menggunakan tools R studio. Dataset yang dijadikan objek prediksi yaitu menggunakan titanic dataset. Metode yang digunakan pada penelitian ini yaitu menggunakan EDA, merupakan proses analisis data yang digunakan untuk memahami karakteristik data secara mendalam sebelum melakukan analisis statistik atau membangun model prediktif. EDA melibatkan serangkaian teknik dan metode yang bertujuan untuk mengidentifikasi pola, anomali, dan informasi penting dalam dataset. Hasil dari akurasi pada penelitian tersebut didapat sebagai berikut logistic regression mendapatkan nilai 0,832761504, kemudian random forest sebesar 0.826031816, selanjutnya decision tree sebesar 0.817059484, dan yang terakhir yaitu model SVM dengan skor 0.831613817.

Penelitian yang dilakukan akan membuat tahapan prediksi menggunakan metode EDA dan visualisasi data. EDA adalah proses eksplorasi data yang lebih luas yang melibatkan sejumlah teknik dan metode untuk memahami karakteristik data sebelum membangun model. EDA mencakup visualisasi data sebagai salah satu komponennya, tetapi juga melibatkan perhitungan statistik deskriptif, identifikasi anomali, pemahaman distribusi variabel, dan eksplorasi hubungan antar variabel. Jadi dapat disimpulkan bahwa penelitian ini menambahkan metode visualisasi data juga dalam tahapan pengeksplorasian dataset yang dijadikan objek prediksi.

Perbedaan selanjutnya yaitu pada penggunaan tools pada saat proses prediksi, tools yang dipakai pada penelitian sebelumnya yaitu menggunakan R studio. Penelitian ini menggunakan python. Pada hasil skor pun akan berbeda dikarenakan tools yang digunakan pun berbeda. Kelemahan pada penelitian sebelumnya yaitu, didapatkannya hasil skor prediksi dari model yang digunakan bernilai desimal. Melihat hasil decimal tersebut, penelitian ini akan membangun skor prediksi model pembelajaran mesin yang bersifat konstan. Skor prediksi didapat dari hasil pembuatan model diatas python.

Tahapan Penelitian



Gambar 1. Alur Penelitian

A. Pengumpulan Data

Pengumpulan dilakukan dengan mengunduh data yang tersedia dalam format digital atau dengan mengumpulkan data dari berbagai sumber . Pada tahap ini, penting untuk memastikan bahwa data dikumpulkan dengan integritas yang baik, artinya data harus lengkap, akurat, dan terstruktur dengan benar. Dataset pada penelitian ini diambil dari Kaggle. Data yang dikumpulkan harus dimasukkan ke dalam format yang sesuai untuk analisis, biasanya dalam format DataFrame . Proses ini melibatkan pembersihan data, penghapusan data yang hilang atau tidak valid, dan penyesuaian format data jika diperlukan.

B. Instalasi Tools

Pada tahap ini akan melakukan instalasi library ataupun model pembelajaran mesin yang akan digunakan pada tahapan penggambaran dataset. Tahap ini juga akan dilakukan pemanggilan model dan library yang akan digunakan pada proses prediksi dataset.

C. Akuisisi Data

Akuisisi data bertujuan untuk mendapatkan informasi penting terkait individu yang terdapat pada dataset. Data-data tersebut mencakup detail individu penumpang seperti usia, jenis kelamin, kelas tiket, jumlah saudara kandung atau pasangan yang ikut serta dalam perjalanan, serta informasi seputar perjalanan seperti titik keberangkatan, tujuan, dan biaya tiket. Selain itu, data juga mencakup informasi kritis tentang kelangsungan hidup penumpang, yang merupakan label atau target dalam model prediksi.

D. Analisa Data dengan Deskripsi Data

Tujuan dari analisis data dengan cara mendeskripsikan data adalah untuk menyelidiki dan memahami karakteristik, pola, dan struktur data yang terkandung dalam dataset. Melalui analisis deskriptif, kita dapat mengidentifikasi statistik dasar seperti rata-rata, median, modus, dan deviasi standar dari setiap variabel, yang memberikan gambaran umum tentang data.

E. Analisa dengan Pivot

Analisis melalui fitur pivoting, bertujuan untuk menjelajahi dan memahami hubungan antara berbagai fitur atau variabel dalam dataset dengan tingkat kelangsungan hidup penumpang. Melalui teknik ini, peneliti dapat mengidentifikasi pola dan korelasi antara berbagai atribut seperti kelas tiket, jenis kelamin, usia, jumlah saudara kandung atau pasangan, dan lainnya dengan kelangsungan hidup penumpang. Hal ini penting karena membantu peneliti memahami faktor-faktor apa yang paling mempengaruhi peluang seseorang selamat dalam situasi bencana seperti yang terjadi pada kapal Titanic.

F. Visualisasi Data

Analisa menggunakan visualisasi data, tujuannya adalah untuk mengidentifikasi pola, tren, dan anomali dalam data dengan cara yang lebih jelas dan memudahkan pemahaman. Selain itu, visualisasi juga membantu dalam menjawab pertanyaan-pertanyaan yang mungkin timbul, seperti bagaimana sebaran usia penumpang Titanic, bagaimana hubungan antara variabel jenis kelamin dan tingkat kelangsungan hidup, atau apakah ada perbedaan signifikan dalam kelangsungan hidup antara kelas tiket

G. Membangun Model untuk Prediksi

Beberapa model akan dimasukkan pada python, diantaranya KNN, Naïve Bayes Classifier, Decision Tree, Random Forest, dll. Model yang dimasukkan akan digunakan untuk melakukan proses prediksi skor.

H. Analisa Hasil Skor Prediksi

Tahap terakhir yaitu analisa hasil prediksi, hasil prediksi akan digunakan untuk dianalisa yang nantinya akan ditemukan model yang memiliki skor

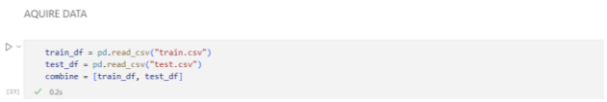
prediksi yang lebih baik diantara model-model lainnya.

HASIL DAN PEMBAHASAN

Berikut adalah hasil dari penelitian proses prediksi dataset menggunakan python. Pada bagian hasil terdapat tahapan-tahapan pada saat proses pengeksplorasian dan penggambaran data. Bagian pembahasan akan menjelaskan tentang perbandingan skor prediksi menggunakan model pembelajaran mesin yang dipakai pada penumpang kapal dengan dataset titanic.

Hasil

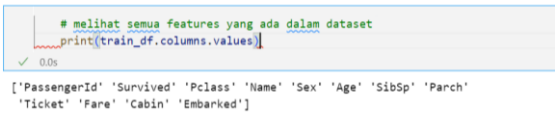
A. Akuisisi Data



Gambar 2. Akuisisi Data

Gambar 2 menunjukkan tahap awal dalam memproses data Titanic, yaitu membaca dua file CSV, yaitu "train.csv" dan "test.csv," menggunakan library Pandas. Setelah membaca kedua file tersebut, data dari masing-masing file dimuat ke dalam dataframe train_df dan test_df. Selanjutnya, kedua dataframe tersebut dimasukkan ke dalam list combine. Dengan mengimpor data dari file CSV ke dalam dataframe, analis data dapat mulai menjelajahi, membersihkan, dan mempersiapkan data untuk analisis lebih lanjut serta pemodelan machine learning.

B. Analisa Data dengan Deskripsi Data



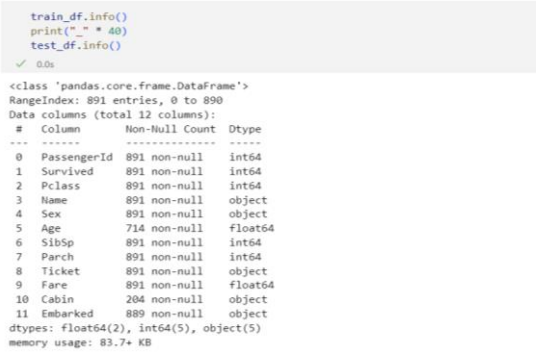
Gambar 3. Analisa Data dengan Deskripsi data

Pada gambar 3 dapat dilihat bahwa tahapan tersebut bertujuan untuk memunculkan daftar nama kolom dalam dataset train_df, berguna untuk mengidentifikasi fitur yang tersedia dan memberikan panduan untuk analisis data.

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Gambar 4. Analisa Data dengan Deskripsi data

Perintah pada gambar 4 adalah langkah awal yang umum dalam analisis data, digunakan untuk menampilkan lima baris pertama dari dataframe train_df. Dengan melihat beberapa baris awal data, dapat dengan cepat mendapatkan gambaran tentang struktur dataset dan isi dari kolom-kolomnya.



Gambar 5. Analisa Data dengan Deskripsi Data

Gambar 5 mengilustrasikan bahwa pemeriksaan informasi seperti ini adalah tahap penting dalam eksplorasi awal data. Langkah ini tidak hanya membantu dalam pemahaman data dan pelaksanaan tahap awal pembersihan, tetapi juga membangun dasar yang diperlukan untuk analisis yang lebih mendalam dan pengembangan model machine learning. Tahapan ini memberikan kesempatan bagi analis data untuk mengidentifikasi potensi masalah dalam data dan merencanakan langkah-langkah selanjutnya dalam pengolahan dan analisis data yang lebih akurat

```
train_df.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

Gambar 6. Analisa Data dengan Deskripsi Data

Perintah pada gambar 6 digunakan untuk menghasilkan statistik deskriptif dari dataset train_df. Hasil dari kompilasi kodingan ini adalah tabel yang memuat informasi statistik terkait setiap kolom numerik dalam dataset, seperti rata-rata (mean), standar deviasi (standard deviation), nilai minimum (minimum), nilai kuartil pertama (25th percentile), median (50th percentile), nilai kuartil ketiga (75th percentile), dan nilai maksimum (maximum).

```
train_df.describe(include=['O'])
```

	Name	Sex	Ticket	Cabin	Embarked
count	891	891	891	204	889
unique	891	2	681	147	3
top	Braund, Mr. Owen Harris	male	347082	B96 B98	S
freq	1	577	7	4	644

Gambar 7. Analisa Data dengan Deskripsi Data

Dalam Gambar 7, tahapannya dirancang untuk membantu memahami karakteristik dari kolom-kolom kategori dalam dataset Titanic. Tahap ini bermanfaat dalam menjelajahi distribusi kategori, mengidentifikasi nilai yang paling umum, serta merencanakan langkah-langkah pemrosesan data yang sesuai, seperti mengatasi nilai yang hilang atau mengkodekan kategori dalam format yang cocok untuk pemodelan machine learning.

C. Analisa dengan Pivot

Analyze by pivoting features

```
train_df[['Pclass', 'Survived']].groupby(['Pclass']
                                         , as_index=False).mean().sort_values(by='Survived',
                                                                              ascending=False)
```

	Pclass	Survived
0	1	0.629630
1	2	0.472826
2	3	0.242363

Gambar 8. Analisa dengan Pivot

Gambar 8 menunjukkan perintah yang digunakan untuk mengurutkan daftar kelas tiket berdasarkan tingkat kelangsungan hidup rata-rata. Ini bertujuan memberikan wawasan apakah kelas tiket berpengaruh pada kelangsungan hidup penumpang dalam dataset Titanic. Dalam konteks ini, kelas tiket dengan tingkat kelangsungan hidup rata-rata tertinggi akan muncul sebagai yang pertama dalam hasil.

```
train_df[['Sex', 'Survived']].groupby(['Sex'],
                                      as_index=False).mean().sort_values(by='Survived',
                                                                         ascending=False)
```

	Sex	Survived
0	female	0.742038
1	male	0.188908

Gambar 9. Analisa dengan Pivot

Pada Gambar 9, perintah tersebut digunakan untuk menganalisis dampak jenis kelamin (kolom 'Sex') terhadap tingkat kelangsungan hidup (kolom 'Survived') dalam dataset Titanic. Kode ini melibatkan serangkaian langkah, termasuk pengelompokan data berdasarkan jenis kelamin, perhitungan rata-rata, dan pengurutan hasil. Hasil yang dihasilkan adalah sebuah tabel yang memuat informasi tentang rata-rata tingkat kelangsungan hidup berdasarkan jenis kelamin, yang diurutkan dari yang tertinggi hingga yang terendah.

```
train_df[['SibSp', 'Survived']].groupby(['SibSp'], as_index=False).mean().sort_values(by='Survived', ascending=False)
```

	SibSp	Survived
1	1	0.535885
2	2	0.464286
0	0	0.345395
3	3	0.250000
4	4	0.166667
5	5	0.000000
6	8	0.000000

Gambar 10. Analisa dengan Pivot

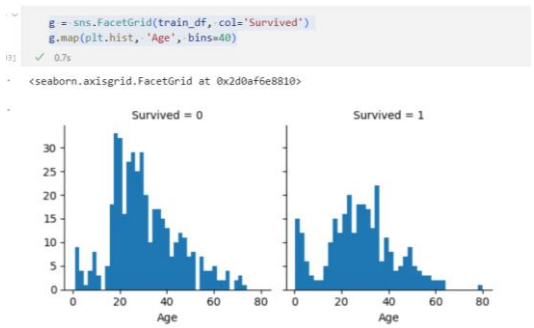
Dapat dilihat pada gambar 10, perintah tersebut digunakan untuk menganalisis pengaruh jumlah saudara kandung/pasangan (kolom 'SibSp') terhadap tingkat kelangsungan hidup (kolom 'Survived') dalam dataset Titanic. Langkah-langkah analisis yang dijalankan melibatkan pengelompokan data berdasarkan jumlah saudara kandung/pasangan, perhitungan rata-rata

kelangsungan hidup, dan pengurutan hasil berdasarkan tingkat kelangsungan hidup. Hasil dari perintah tersebut adalah tabel yang memuat informasi tentang rata-rata tingkat kelangsungan hidup berdasarkan jumlah saudara kandung/pasangan. Tabel tersebut diurutkan mulai dari yang tertinggi hingga yang terendah berdasarkan tingkat kelangsungan hidup.

Gambar 11. Analisa dengan Pivot

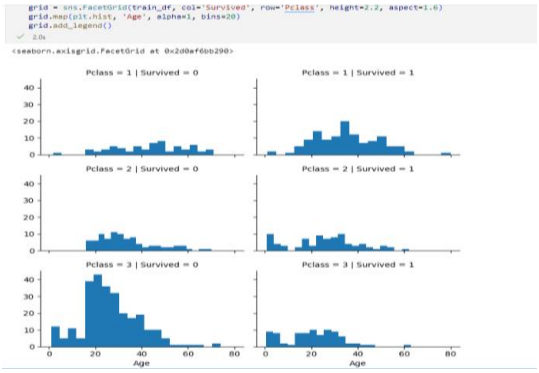
Perintah pada Gambar 11 digunakan untuk menganalisis dampak jumlah orang tua/anak (kolom 'Parch') terhadap tingkat kelangsungan hidup (kolom 'Survived') dalam dataset Titanic. Proses ini melibatkan serangkaian langkah, termasuk pengelompokan data berdasarkan jumlah orang tua/anak, perhitungan rata-rata kelangsungan hidup, dan pengurutan hasil berdasarkan tingkat kelangsungan hidup. Hasil dari perintah tersebut adalah sebuah tabel yang berisi informasi rata-rata tingkat kelangsungan hidup berdasarkan jumlah orang tua/anak. Tabel tersebut diurutkan mulai dari yang tertinggi hingga yang terendah berdasarkan tingkat kelangsungan hidup.

D. Visualisasi Data



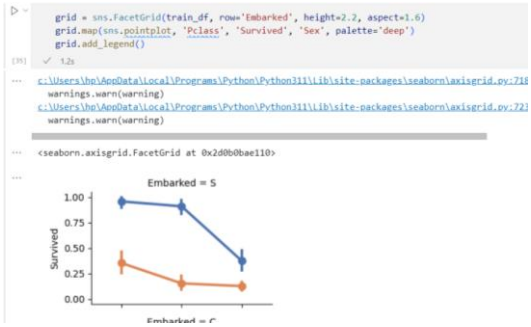
Gambar 12. Visualisasi Data

Pada Gambar 12, digunakan library Seaborn (disingkat sebagai 'sns') untuk membuat visualisasi yang menggambarkan distribusi usia penumpang Titanic berdasarkan kelangsungan hidup (kolom 'Survived'). Hasil dari visualisasi ini adalah dua histogram yang ditampilkan berdampingan, yang masing-masing merepresentasikan distribusi usia penumpang yang selamat dan yang tidak selamat. Visualisasi ini memungkinkan kita untuk mengeksplorasi sejauh mana distribusi usia mempengaruhi kelangsungan hidup penumpang. Informasi ini dapat membantu dalam menentukan apakah usia memiliki peran signifikan dalam tingkat kelangsungan hidup penumpang di kapal Titanic.



Gambar 13. Visualisasi Data

Gambar 13 menampilkan hasil dari grid visualisasi yang terdiri dari sejumlah subplot. Setiap subplot menggambarkan histogram yang merepresentasikan distribusi usia penumpang dalam kelas tiket tertentu berdasarkan kelangsungan hidup. Visualisasi ini memungkinkan kita untuk melakukan analisis terhadap hubungan antara distribusi usia dengan faktor kelas tiket dan kelangsungan hidup penumpang. Informasi yang dihasilkan dari visualisasi ini dapat memberikan wawasan yang berharga dan lebih mendalam terkait peran usia dan kelas tiket dalam tingkat kelangsungan hidup penumpang di kapal Titanic



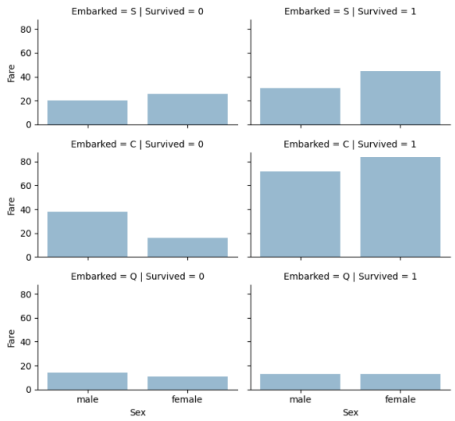
Gambar 14. Visualisasi Data

Perintah pada Gambar 14 menciptakan sebuah grid visualisasi yang terdiri dari sejumlah subplot. Setiap subplot memvisualisasikan data menggunakan pointplot, yang memungkinkan kita untuk membandingkan hubungan antara kelas tiket, kelangsungan hidup, dan jenis kelamin penumpang berdasarkan pelabuhan embarkasi. Visualisasi ini memungkinkan analisis interaksi antara berbagai faktor dalam dataset Titanic, seperti bagaimana kelas tiket, kelangsungan hidup, dan jenis kelamin berkaitan dengan pelabuhan embarkasi. Informasi yang dihasilkan dari visualisasi ini memberikan wawasan yang berharga terkait perbedaan dalam distribusi penumpang berdasarkan faktor-faktor tersebut, dan memungkinkan kita untuk lebih memahami bagaimana variabel-variabel ini saling berhubungan dalam konteks peristiwa Titanic.

```
grid = sns.FacetGrid(train_df, row='Embarked', col='Survived', height= 2.2, aspect=1.6)
grid.map(sns.barplot, 'Sex', 'Fare', alpha=.5, ci=None)
grid.add_legend()
```

✓ 1.3s

Gambar 15. Perintah pada Pemrograman Python



Gambar 16. Hasil dari Perintah Gambar(15)

Gambar 16 menggambarkan hasil dari perintah pada Gambar 15, yang menghasilkan sebuah grid visualisasi yang terdiri dari beberapa subplot. Setiap subplot menampilkan barplot yang membandingkan hubungan antara biaya tiket dan jenis kelamin penumpang dalam konteks pelabuhan embarkasi serta kelangsungan hidup. Melalui visualisasi ini, kita dapat melakukan analisis mendalam terhadap cara di mana biaya tiket dan jenis kelamin berhubungan dengan faktor pelabuhan embarkasi dan kelangsungan hidup penumpang di kapal Titanic. Hasil visualisasi ini memberikan wawasan yang berharga dan mendalam dalam pemahaman interaksi kompleks antara variabel-variabel tersebut dalam konteks peristiwa Titanic.

Gambar 17. Visualisasi Data

Pada Gambar 17, perintah tersebut mencetak dimensi dataset setelah menghapus kolom yang tidak diperlukan dan menampilkan hasilnya sebagai "After". Tindakan ini bertujuan untuk memastikan bahwa dataset telah dibersihkan dari kolom yang tidak relevan sehingga siap digunakan dalam tahap analisis dan pemodelan selanjutnya. Pemilihan kolom yang relevan dan penghapusan kolom yang tidak

diperlukan merupakan langkah kritis dalam persiapan data untuk analisis data dan pengembangan model machine learning.

```
for dataset in combine:
    dataset['Title'] = dataset.Name.str.extract('([A-Za-z]+)\.', expand=False)
pd.crosstab(train_df['Title'], train_df['Sex'])
for dataset in combine:
    dataset['Title'] = dataset['Title'].replace([
        'Lady', 'Countess', 'Capt', 'Col','Don', 'Major',
        'Rev', 'Sir', 'Jonkheer', 'Dona', 'Dr'], 'Rare')
    dataset['Title'] = dataset['Title'].replace(['Mlle', 'Ms'], 'Miss')
    dataset['Title'] = dataset['Title'].replace('Mme', 'Mrs')
train_df[['Title', 'Survived']].groupby(['Title'], as_index=False).mean()
```

	Title	Survived
0	Master	0.575000
1	Miss	0.702703
2	Mr	0.156673
3	Mrs	0.793651
4	Rare	0.347826

Gambar 18. Visualisasi Data

Perintah pada Gambar 18 merupakan serangkaian langkah pengolahan data dalam dataset Titanic yang bertujuan untuk mengekstrak gelar penumpang dari kolom 'Name'. Hasil dari langkah-langkah ini adalah penambahan kolom baru 'Title' yang mencerminkan gelar penumpang. Hal ini memiliki manfaat dalam analisis lebih lanjut untuk memahami peran gelar dalam kelangsungan hidup penumpang di Titanic, sambil juga melakukan penyederhanaan data dengan menggantikan gelar-gelar yang jarang muncul dengan 'Rare'.

```
title_mapping = {"Mr": 1, "Miss": 2, "Mrs": 3, "Master":4, "Rare":5}
for dataset in combine:
    dataset['Title'] = dataset['Title'].map(title_mapping)
    dataset['Title'] = dataset['Title'].fillna(0)
train_df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked	Title
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	7.2500	S	1
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	71.2833	C	3
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	7.9250	S	2
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	53.1000	S	3
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	8.0500	S	1

Gambar 19. Visualisasi Data

Pada Gambar 19, terlihat bahwa langkah-langkah penggantian gelar-gelar dalam bentuk teks dengan representasi numerik telah berhasil dilakukan. Proses ini memiliki manfaat penting dalam konteks pemodelan machine learning, karena kebanyakan algoritma model memerlukan data dalam bentuk numerik. Dengan demikian, dataset telah disiapkan dengan baik untuk tahap analisis dan pemodelan selanjutnya. Penggunaan perintah terakhir, train_df.head(), bertujuan untuk memeriksa hasil perubahan tersebut dengan menampilkan lima baris pertama dari dataset pelatihan.

```
train_df = train_df.drop(['Name', 'PassengerId'], axis=1)
test_df = test_df.drop(['Name'], axis=1)
combine = [train_df, test_df]
```

Gambar 20. Visualisasi Data

Perintah pada Gambar 20 menghapus kolom-kolom yang dianggap tidak relevan dalam dataset, memfokuskan data pada informasi yang lebih relevan untuk menganalisis faktor-faktor yang mempengaruhi kelangsungan hidup penumpang Titanic.

```
for dataset in combine:
    dataset['Sex'] = dataset['Sex'].map({'female':1, 'male':0}).astype(int)
train_df.head()
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	Title
0	0	3	0	22.0	1	0	7.2500	S	1
1	1	1	1	38.0	1	0	71.2833	C	3
2	1	3	1	26.0	0	0	7.9250	S	2
3	1	1	1	35.0	1	0	53.1000	S	3
4	0	3	0	35.0	0	0	8.0500	S	1

Gambar 21. Visualisasi Data

Gambar 21 menunjukkan langkah-langkah yang mengubah kolom 'Sex' dari format teks menjadi format numerik, yang merupakan tahap penting dalam persiapan data untuk pemodelan machine learning. Jenis kelamin penumpang sekarang direpresentasikan sebagai angka 1 (untuk wanita) dan 0 (untuk pria), yang memudahkan penggunaan data ini dalam model-model prediksi. Perintah terakhir, train_df.head(), digunakan untuk memeriksa hasil transformasi ini dengan menampilkan lima baris pertama dari dataset pelatihan.

```
guess_ages = np.zeros((2,3))
for dataset in combine:
    for i in range(0, 2):
        for j in range(0, 3):
            guess_df = dataset[(dataset['Sex'] == i) & \
                               (dataset['Pclass'] == j+1)]['Age'].dropna()
            age_guess = guess_df.median()

            #convert random age float to nearest .5 age
            guess_ages[i, j] = int( age_guess/0.5 + 0.5) * 0.5

    for i in range(0,2):
        for j in range(0, 3):
            dataset.loc[(dataset.Age.isnull()) & (dataset.Sex == i) & (dataset.Pclass == j+1),'Age'] = guess_ages[i, j]

    dataset['Age'] = dataset['Age'].astype(int)
train_df.head()
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	Title
0	0	3	0	22	1	0	7.2500	S	1
1	1	1	1	38	1	0	71.2833	C	3
2	1	3	1	26	0	0	7.9250	S	2
3	1	1	1	35	1	0	53.1000	S	3
4	0	3	0	35	0	0	8.0500	S	1

Gambar 22. Visualisasi Data

Pada gambar 22, dapat dilihat bahwa kode yang diberikan digunakan untuk mengisi nilai-nilai yang hilang dalam

kolom "Age" dalam dataset Titanic. Proses ini dilakukan dengan menghitung nilai median usia berdasarkan kombinasi jenis kelamin (Sex) dan kelas penumpang (Pclass) kemudian menggantikan nilai-nilai yang hilang dengan perkiraan usia tersebut. Hasil dari pelaksanaan kode ini adalah dataset yang telah diperbarui, dengan kolom "Age" yang sudah terisi dengan perkiraan usia berdasarkan jenis kelamin dan kelas penumpang, sehingga tidak ada lagi nilai-nilai yang kosong dalam kolom "Age." Penambahan informasi ini dapat mendukung analisis data yang lebih lanjut, terutama jika usia merupakan fitur yang penting dalam analisis tersebut.

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	Title
0	0	3	0	1	1	0	7.2500	S	1
1	1	1	1	2	1	0	71.2833	C	3
2	1	3	1	1	0	0	7.9250	S	2
3	1	1	1	2	1	0	53.1000	S	3
4	0	3	0	2	0	0	8.0500	S	1

Gambar 23. Visualisasi Data

Pada Gambar 23, kode ini digunakan untuk mengelompokkan nilai usia (Age) dalam dataset Titanic menjadi beberapa kelompok umur guna menyederhanakan analisis. Saat melakukan iterasi dataset dalam variabel combine, kode ini mengkategorikan nilai-nilai usia ke dalam kelompok umur yang relevan. Dengan tindakan ini, kolom "Age" dalam dataset diperbarui sesuai dengan kelompok usia yang telah ditentukan, memberikan dasar yang lebih mudah dipahami untuk analisis selanjutnya. Pengelompokan usia ini memudahkan identifikasi pola-pola terkait usia dalam konteks kelangsungan hidup atau faktor-faktor lain dalam dataset Titanic. Hasil dari perubahan ini dapat diamati dengan menjalankan train_df.head(), yang menampilkan contoh beberapa baris data teratas dalam dataset yang telah diperbarui.

	FamilySize	Survived
3	4	0.724138
2	3	0.578431
1	2	0.552795
6	7	0.333333
0	1	0.303538
4	5	0.200000
5	6	0.136364
7	8	0.000000
8	11	0.000000

Gambar 24. Visualisasi Data

Pada Gambar 24, perintah ini digunakan untuk menghasilkan informasi statistik terkait ukuran keluarga (FamilySize) dalam dataset Titanic. Saat melalui setiap iterasi dataset dalam combine, kolom 'FamilySize' dibentuk dengan menggabungkan jumlah 'SibSp' (saudara atau pasangan) dan 'Parch' (orang tua atau anak), ditambah satu, untuk merepresentasikan total anggota dalam keluarga seseorang di atas kapal. Dengan kata lain, 'FamilySize' mengindikasikan jumlah individu dalam keluarga. Hasil dari perintah ini memberikan wawasan tentang keterkaitan antara ukuran keluarga dan peluang bertahan hidup di atas kapal Titanic

	IsAlone	Survived
0	0	0.505650
1	1	0.303538

Gambar 25. Visualisasi Data

Pada Gambar 25, perintah ini digunakan untuk membuat dan mengisi kolom 'IsAlone' dalam dataset Titanic dengan tujuan mengklasifikasikan apakah seorang penumpang bepergian sendirian atau tidak. Saat melalui dataset dalam variabel 'combine', kolom 'IsAlone' awalnya diinisialisasi dengan nilai 0, mengindikasikan asumsi awal bahwa penumpang tidak bepergian sendirian. Menggunakan .loc, kode mengubah nilai 'IsAlone' menjadi 1 untuk penumpang yang memiliki 'FamilySize' (ukuran keluarga) setara dengan 1, menunjukkan bahwa mereka tidak memiliki saudara, pasangan, orang tua, atau anak yang menemani mereka di atas kapal.

Hasil dari perintah ini memberikan informasi apakah penumpang yang bepergian sendirian memiliki tingkat kelangsungan hidup yang berbeda dibandingkan dengan penumpang yang bepergian bersama keluarga atau teman. Dengan demikian, analisis ini membantu mengidentifikasi apakah faktor kesendirian berpengaruh terhadap peluang bertahan hidup di atas kapal Titanic.

```
train_df = train_df.drop(['Parch', 'SibSp', 'FamilySize'], axis=1)
test_df = test_df.drop(['Parch', 'SibSp', 'FamilySize'], axis=1)
combine = [train_df, test_df]
train_df.head()
```

✓ 0.0s

	Survived	Pclass	Sex	Age	Fare	Embarked	Title	IsAlone
0	0	3	0	1	7.2500	S	1	0
1	1	1	1	2	71.2833	C	3	0
2	1	3	1	1	7.9250	S	2	1
3	1	1	1	2	53.1000	S	3	0
4	0	3	0	2	8.0500	S	1	1

Gambar 26. Visualisasi Data

Pada Gambar 25, terdapat perintah yang bertujuan untuk menghapus sejumlah kolom tertentu dari dataset Titanic yang disimpan dalam variabel train_df dan test_df. Kolom-kolom yang dihapus mencakup 'Parch', 'SibSp', dan 'FamilySize', yang sebelumnya telah digunakan untuk menghitung ukuran keluarga dan mengklasifikasikan penumpang apakah bepergian sendirian atau tidak. Setelah kolom-kolom ini dihapus, dataset yang telah diperbarui kembali disimpan dalam variabel train_df dan test_df. Selanjutnya, kedua dataset ini dimasukkan ke dalam variabel combine.

Dampak dari eksekusi perintah ini adalah train_df yang kini hanya berisi kolom-kolom yang telah dipilih setelah penghapusan. Hasil ini dapat dilihat melalui perintah train_df.head(). Penghapusan kolom-kolom yang tidak lagi dibutuhkan dapat membantu mempermudah analisis dataset dan mengurangi kompleksitas data, sehingga memungkinkan untuk fokus pada fitur-fitur yang lebih relevan dalam proses analisis lebih lanjut.

```
for dataset in combine:
    dataset['Age*Class'] = dataset.Age * dataset.Pclass
train_df.loc[:,['Age*Class', 'Age', 'Pclass']].head(10)
```

✓ 0.0s

	Age*Class	Age	Pclass
0	3	1	3
1	2	2	1
2	3	1	3
3	2	2	1
4	6	2	3
5	3	1	3
6	3	3	1
7	0	0	3
8	3	1	3
9	0	0	2

Gambar 27. Visualisasi Data

Pada Gambar 27, terdapat perintah yang bertujuan untuk membuat kolom baru yang disebut 'AgeClass' dalam dataset Titanic yang disimpan dalam variabel train_df dan test_df. Kolom 'AgeClass' ini akan mengandung hasil perkalian antara nilai usia (Age) dan kelas penumpang (Pclass) untuk setiap penumpang dalam dataset. Perintah ini digunakan untuk menciptakan fitur baru yang menggabungkan usia dan kelas penumpang menjadi satu variabel, yang dapat memberikan wawasan lebih dalam dalam analisis data.

Hasil dari perintah ini memungkinkan kita untuk melihat bagaimana hasil perkalian usia dan kelas penumpang memengaruhi data dalam dataset Titanic. Fitur 'Age*Class' memiliki potensi untuk memberikan informasi tambahan yang berguna dalam analisis data, dan dapat membantu dalam menentukan apakah ada hubungan antara kombinasi usia dan kelas penumpang dengan tingkat kelangsungan hidup atau faktor-faktor lain dalam dataset tersebut.

```
freq_port = train_df.Embarked.dropna().mode()[0]
for dataset in combine:
    dataset['Embarked'] = dataset['Embarked'].fillna(freq_port)
train_df[['Embarked', 'Survived']].groupby(['Embarked'], as_index=False).mean().sort_values(by='Survived', ascending=False)
```

✓ 0.0s

	Embarked	Survived
0	C	0.553571
1	Q	0.389610
2	S	0.339009

Gambar 28. Visualisasi Data

Pada Gambar 28, terdapat kode yang memiliki tujuan untuk mengatasi nilai-nilai yang hilang dalam kolom 'Embarked' (pelabuhan embarkasi) dalam dataset Titanic serta untuk mengidentifikasi hubungannya dengan tingkat kelangsungan hidup penumpang.Langkah pertama dalam kode ini adalah mencari pelabuhan embarkasi yang paling umum (mode) dalam dataset pelatihan (train_df). Ini

dilakukan dengan menggunakan train_df.Embarked.dropna().mode()[0], yang akan memberikan nilai pelabuhan yang paling sering muncul dalam data pelatihan. Selanjutnya, selama perulangan melalui dataset dalam variabel combine, nilai-nilai yang hilang dalam kolom 'Embarked' digantikan dengan nilai pelabuhan yang paling umum yang telah ditemukan sebelumnya. Dengan kata lain, langkah ini mengisi nilai-nilai yang kosong dengan nilai mode pelabuhan embarkasi.

Hasil dari perintah ini adalah data yang telah diperbarui dengan nilai pelabuhan embarkasi yang sesuai dan lengkap. Perintah selanjutnya, train_df[['Embarked', 'Survived']].groupby(['Embarked'], as_index=False).mean().sort_values(by='Survived', ascending=False), digunakan untuk menghitung rata-rata tingkat kelangsungan hidup (Survived) berdasarkan pelabuhan embarkasi ('Embarked'). Hasilnya disajikan dalam urutan menurun berdasarkan tingkat kelangsungan hidup, sehingga kita dapat memahami sejauh mana pelabuhan embarkasi dapat mempengaruhi nasib penumpang Titanic dan faktor-faktor yang mungkin berkaitan dengannya.

```
for dataset in combine:
    dataset['Embarked'] = dataset['Embarked'].map({'S':0,
                                                   'C': 1,
                                                   'Q': 2}).astype(int)
train_df.head()
```

0.0s

	Survived	Pclass	Sex	Age	Fare	Embarked	Title	IsAlone	Age*Class
0	0	3	0	1	7.2500	0	1	0	3
1	1	1	1	2	71.2833	1	3	0	2
2	1	3	1	1	7.9250	0	2	1	3
3	1	1	1	2	53.1000	0	3	0	2
4	0	3	0	2	8.0500	0	1	1	6

Gambar 29. Visualisasi Data

Pada Gambar 29, ini adalah tahap terakhir dalam proses visualisasi data yang digunakan untuk mengubah nilai-nilai dalam kolom 'Embarked' (pelabuhan embarkasi) dalam dataset Titanic menjadi representasi numerik, yang lebih mudah diproses dalam analisis data. Selama perulangan melalui dataset dalam variabel 'combine', nilai-nilai dalam kolom 'Embarked' mengalami transformasi melalui metode pemetaan (mapping). Nilai 'S' digantikan dengan 0, 'C' digantikan dengan 1, dan 'Q' digantikan dengan 2. Setelah tahap pemetaan ini selesai, kolom 'Embarked' dalam dataset berisi angka-angka yang mewakili pelabuhan embarkasi.

Hasil dari perintah ini adalah dataset yang telah diperbarui dengan kolom 'Embarked' yang sekarang memuat nilai-nilai numerik sebagai pengganti pelabuhan embarkasi. Hal ini memudahkan analisis data karena algoritma dan metode analisis cenderung lebih efisien dalam memproses data numerik dibandingkan dengan data berjenis teks atau kategori. Perintah train_df.head() digunakan untuk melihat sepuluh baris pertama dari dataset train_df yang sudah diperbarui dengan perubahan ini.

E. Membuat model Menggunakan Pembelajaran Mesin

Berikut adalah perintah untuk membangun model pembelajaran mesin, yang nantinya akan digunakan untuk membuat skor prediksi pada dataset titanic:

```
X_train = train_df.drop("Survived", axis=1)
Y_train = train_df["Survived"]
X_test = test_df.drop("PassengerId", axis=1).copy()
X_train.shape, Y_train.shape, X_test.shape
# Logistic Regression
logreg = LogisticRegression()
logreg.fit(X_train, Y_train)
Y_pred = logreg.predict(X_test)
acc_log = round(logreg.score(X_train, Y_train) * 100, 2)
print(acc_log, '%')
# SVM (Support Vector Machines)
svc = SVC()
svc.fit(X_train, Y_train)
Y_pred = svc.predict(X_test)
acc_svc = round(svc.score(X_train, Y_train)*100, 2)
acc_svc
# K-Nearest Neighbors
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(X_train, Y_train)
Y_pred = knn.predict(X_test)
acc_knn = round(knn.score(X_train, Y_train) * 100, 2)
acc_knn
# Gaussian Naive Bayes
gaussian = GaussianNB()
gaussian.fit(X_train, Y_train)
Y_pred = gaussian.predict(X_test)
acc_gaussian = round(gaussian.score(X_train, Y_train) * 100, 2)
acc_gaussian
# Perceptron
perceptron = Perceptron()
perceptron.fit(X_train, Y_train)
Y_pred = perceptron.predict(X_test)
acc_perceptron = round(perceptron.score(X_train, Y_train)*100,2)
acc_perceptron
# Linear SVC
linear_svc = LinearSVC()
linear_svc.fit(X_train, Y_train)
Y_pred = linear_svc.predict(X_test)
acc_linear_svc = round(linear_svc.score(X_train, Y_train)*100, 2)
acc_linear_svc
# Stochastic Gradient Descent
sgd = SGDClassifier()
sgd.fit(X_train, Y_train)
Y_pred = sgd.predict(X_test)
acc_sgd = round(sgd.score(X_train, Y_train) * 100, 2)
acc_sgd
# Decision Tree
decision_tree = DecisionTreeClassifier()
decision_tree.fit(X_train, Y_train)
Y_pred = decision_tree.predict(X_test)
acc_decision_tree = round(decision_tree.score(X_train, Y_train) * 100, 2)
acc_decision_tree
# Random Forest
random_forest = RandomForestClassifier(n_estimators=100)
random_forest.fit(X_train, Y_train)
Y_pred = random_forest.predict(X_test)
random_forest.score(X_train, Y_train)
acc_random_forest = round(random_forest.score(X_train, Y_train) * 100, 2)
acc_random_forest
```

Gambar 30. Membuat Model

Kode di atas menggunakan beberapa model pembelajaran mesin untuk tugas klasifikasi pada dataset Titanic. Model-model yang digunakan termasuk Logistic Regression, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Gaussian Naive Bayes, Perceptron, Linear Support Vector Classification (Linear SVC), Stochastic Gradient Descent (SGD), Decision Tree, dan Random Forest. Setiap model digunakan untuk memprediksi kelangsungan hidup penumpang berdasarkan fitur-fitur dalam dataset. Hasil akurasi dari masing-masing model diukur menggunakan data pelatihan dan digunakan untuk memilih model terbaik untuk tugas klasifikasi Titanic.

```
models = pd.DataFrame({
    'Model': ['Support Vector Machine', 'KNN', 'Logistic Regression',
              'Random Forest', 'Naive Bayes', 'Perceptron',
              'Stochastic Gradient Decent', 'Linear SVC', 'Decision Tree'],
    'Score': [acc_svc, acc_knn, acc_log, acc_random_forest,
              acc_gaussian, acc_perceptron, acc_sgd, acc_linear_svc, acc_decision_tree]
})
models.sort_values(by="Score", ascending=False)
```

	Model	Score
3	Random Forest	84.29
8	Decision Tree	84.29
1	KNN	83.05
2	Logistic Regression	80.36
7	Linear SVC	78.79
0	Support Vector Machine	78.23
6	Stochastic Gradient Decent	76.66
4	Naive Bayes	72.17
5	Perceptron	69.02

Gambar 31.Membuat Model

Pada Gambar 31, perintah tersebut menjalankan berbagai model machine learning untuk tugas klasifikasi Titanic dan kemudian menyajikan hasil akurasi masing-masing model dalam dataframe "models." Akurasi ini diperoleh melalui pelatihan setiap model menggunakan data pelatihan Titanic dan pengukuran kinerja model pada data yang sama. Hasil akurasi dari berbagai model ini kemudian disajikan dalam urutan menurun (descending) berdasarkan skor akurasi tertinggi. Informasi ini memudahkan kita untuk dengan cepat mengidentifikasi model yang memberikan kinerja terbaik dalam memprediksi kelangsungan hidup penumpang Titanic.

Pembahasan

Paper ini bertujuan untuk membandingkan kinerja beberapa model pembelajaran mesin yang diterapkan pada dataset Titanic. Hasil akurasi dari berbagai model dievaluasi dan disajikan dalam tabel di atas. Model-model yang diuji termasuk Random Forest, Decision Tree, K-Nearest Neighbors (KNN), Logistic Regression, Linear Support Vector Classification (Linear SVC), Support Vector Machine (SVM), Stochastic Gradient Descent (SGD), Naive Bayes, dan Perceptron.

Berikut adalah hasil dari perhitungan skor akurasi yang dilakukan menggunakan model-model pada pembelajaran mesin dengan dataset titanic:

Tabel 1.Perbandingan Model

No	Model	Score
1	Random Forest	84.29
2	Decision Tree	84.29
3	KNN	83.05
4	Logistic Regression	80.36
5	Linear SVC	78.79
6	Support Vector Machine	78.23
7	Stochastic Gradient Decent	76.66
8	Naive Bayes	72.17
9	Perceptron	69.02

Dari Tabel 1, terlihat bahwa hasilnya memberikan gambaran yang jelas tentang kinerja berbagai model dalam memprediksi kelangsungan hidup penumpang Titanic berdasarkan fitur-fitur dataset. Model-model tersebut diurutkan berdasarkan skor akurasi tertinggi hingga terendah. Oleh karena itu, hasil ini memungkinkan peneliti untuk dengan cepat mengidentifikasi model yang menghasilkan hasil terbaik dalam konteks analisis yang sedang dilakukan. Model seperti Random Forest dan Decision Tree mencapai akurasi tertinggi, sedangkan model-model lain memiliki tingkat akurasi yang lebih rendah. Temuan ini memberikan panduan dalam pemilihan model yang paling sesuai untuk tugas klasifikasi pada dataset Titanic.

KESIMPULAN

Penelitian ini memberikan pengetahuan dalam memahami peran

pemilihan model machine learning dalam tugas klasifikasi, khususnya dalam konteks prediksi kelangsungan hidup penumpang Titanic. Hasil analisis menunjukkan bahwa pemilihan model yang tepat sangat penting, dengan model-model seperti Random Forest dan Decision Tree muncul sebagai pilihan yang kuat. Kontribusi utama penelitian ini adalah memberikan pandangan tentang bagaimana berbagai model dapat berperilaku dalam analisis data dan bagaimana pemilihan model yang bijak dapat mengarah pada hasil yang lebih baik.

Kelebihan penelitian ini adalah pemahaman mendalam tentang karakteristik model-model yang diuji dan kemampuan untuk membandingkannya secara langsung dalam konteks dataset Titanic. Pada penelitian ini juga memungkinkan pemilihannya yang lebih cerdas dan sesuai dengan situasi tertentu. Namun, ada beberapa kekurangan, terutama dalam hal pertimbangan praktis. Meskipun beberapa model mencapai akurasi tinggi, kompleksitas dan sumber daya yang dibutuhkan harus dipertimbangkan dalam penggunaan praktis.

Penelitian lanjutan dari penelitian ini akan mencakup eksplorasi teknik ensemble learning, pengembangan fitur engineering yang lebih canggih, dan integrasi data tambahan yang relevan seperti data kru kapal atau data penumpang yang lebih rinci. Hal ini diharapkan dapat meningkatkan akurasi prediksi dan memperkaya dataset dengan fitur-fitur informatif. Selain itu, penelitian akan fokus pada optimisasi model yang lebih efisien, yang mencakup pemilihan model yang lebih tepat dan pertimbangan praktis. Penelitian ini akan memberikan landasan yang kuat untuk pengembangan lebih lanjut dalam analisis data menggunakan machine learning, terutama dalam konteks peristiwa sejarah seperti bencana Titanic.

DAFTAR PUSTAKA

Ashenden, S. K. (2021). *The era of artificial intelligence, machine learning, and data science in the pharmaceutical industry*. Academic Press.

de Souza, G. F. M., Caminada Netto, A., de Andrade Melani, A. H., de Carvalho Michalski, M. A., & da Silva, R. F. (2022). Chapter 6 - Engineering systems' fault diagnosis methods. In G. F. M. de Souza, A. Caminada Netto, A. H. de Andrade Melani, M. A. de Carvalho Michalski, & R. F. da Silva (Eds.), *Reliability Analysis and Asset Management of Engineering Systems* (pp. 165–187). Elsevier. https://doi.org/https://doi.org/10.1016/B978-0-12-823521-8.00006-2

Kakde, Y., & Agrawal, S. (2018). Predicting survival on Titanic by applying exploratory data analytics and machine learning techniques. *International Journal of Computer Applications*, *179*(44), 32–38.

Kenyhercz, M. W., & Passalacqua, N. V. (2016). Missing data imputation methods and their performance with biodistance analyses. In *Biological Distance Analysis* (pp. 181–194). Elsevier.

Mosavi, A., Ozturk, P., & Chau, K. (2018). Flood prediction using machine learning models: Literature review. *Water*, *10*(11), 1536.

Pratiwi, N., & Setyawan, Y. (2021). Analisis Akurasi dari Perbedaan Fungsi Kernel dan Cost Pada Support Vector Machine Studi Kasus Klasifikasi Curah Hujan di Jakarta. *Journal of Fundamental Mathematics and Applications (JFMA)*, *4*(2), 203–212.

Rahmanita, M., Ricardianto, P., Wijayanti, R., Agusinta, L., Asmaniati, F., Djati, S., Tatiana, Y., Arafah, W., Amsyari, I., & Endri, E. (2023). The impact of the safety of passenger ship services on the development of water

recreation: evidence from Indonesia. *Uncertain Supply Chain Management*, *11*(3), 1121–1132.

Sanni, S. E., Okoro, E. E., Sadiku, E. R., & Oni, B. A. (2022). Advances in data-centric intelligent systems for air quality monitoring, assessment, and control. In *Current Trends and Advances in Computer-Aided Intelligent Environmental Data Engineering* (pp. 25–58). Elsevier.

Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. *SN Computer Science*, *2*(3), 160.

Sharma, A. (2018). Guided stochastic gradient descent algorithm for inconsistent datasets. *Applied Soft Computing*, *73*, 1068–1080.

Singh, A., Saraswat, S., & Faujdar, N. (2017). Analyzing Titanic disaster using machine learning algorithms. *2017 International Conference on Computing, Communication and Automation (ICCCA)*, 406–411.

Singh, J., Singh, J., Singh, G., & Kaur, N. (2022). Exploratory Data Analysis for Interpreting Model Prediction using Python. *2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON)*, 1–6.

Tipping, M. (1999). The relevance vector machine. *Advances in Neural Information Processing Systems*, *12*.

Widrow, B., & Lehr, M. A. (n.d.). *ARTIFIcıAL NEURAL NETwoRKs OF THE PERCEPTRON, MADALINE, AND*.