

IMPLEMENTASI DECISION TREE UNTUK PREDIKSI SURVIVAL TITANIC

Tugas ini dikerjakan untuk memenuhi tugas UAS Machine Learning



Dosen Pengampuh: Agung Perdananto, S.Kom., M.Kom.

Dibuat oleh:

Abdul Hafizh Ar Rasyid

231011403352

Kelas 05TPLE004

**TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PAMULANG
2025**

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan tugas mata kuliah Machine Learning ini dengan baik. Tugas berjudul “Implementasi Decision Tree untuk Prediksi Survival Titanic” ini disusun sebagai salah satu syarat penilaian dalam mata kuliah Machine Learning.

Penulis menyadari bahwa penyusunan tugas ini tidak akan tercapai tanpa bantuan, dukungan, dan bimbingan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Agung Perdananto, S.Kom., M.Kom. selaku dosen pengampu mata kuliah Machine Learning, yang telah memberikan bimbingan, ilmu, motivasi, serta arahan yang sangat berharga selama proses pembelajaran dan penyusunan tugas ini.
2. Keluarga tercinta yang senantiasa memberikan dukungan moral dan doa dalam setiap langkah penulis.

Penulis menyadari bahwa tugas ini masih jauh dari sempurna dan masih terdapat banyak kekurangan. Oleh karena itu, saran dan kritik yang membangun sangat diharapkan untuk perbaikan di masa mendatang. Semoga hasil penelitian ini dapat bermanfaat bagi pengembangan ilmu pengetahuan, khususnya dalam bidang kecerdasan buatan dan aplikasinya dalam industri pangan.

Tangerang Selatan, 05 Januari 2026

Abdul Hafizh Ar Rasyid

DAFTAR ISI

IMPLEMENTASI DECISION TREE UNTUK PREDIKSI SURVIVAL TITANIC	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Tujuan Laporan	1
1.3 Dataset Titanic.....	2
BAB II KONSEP TEORI.....	3
2.1 Decision Tree	3
2.2 Konsep-konsep dalam Decision Tree.....	3
2.2.1 Node (Simpul)	3
2.2.2 Root (Akar).....	3
2.2.3 Leaf (Daun).....	3
2.2.4 Splitting (Pemecahan).....	3
2.2.5 Pruning (Pemangkasan).....	4
2.3 Perbedaan Decision Tree, Random Forest, dan Gradient Boosting.....	4
2.4 Kelebihan dan Kekurangan Tree-based Methods	4
BAB III METODOLOGI.....	6
3.1 Diaram Alur.....	6
3.2 Tahapan Preprocessing Data	7
3.3 Proses Modeling Decision Tree	7
BAB IV HASIL DAN ANALISIS.....	9
4.1 Hasil Preprocessing Data	9
4.2 Hasil Eksperimen Hyperparameter Tuning.....	9
4.3 Analisis Confusion Matrix	9
4.4 Feature Importance Analysis.....	10
4.5 Visualisasi Decision Tree.....	10
4.6 Diskusi Hasil	11
4.7 Evaluasi Kritis terhadap Model.....	11

BAB V KESIMPULAN DAN SARAN.....	12
5.1 Kesimpulan.....	12
5.2 Saran.....	12
LAMPIRAN.....	14

BAB I

PENDAHULUAN

1.1 Latar Belakang

Machine Learning merupakan salah satu teknologi yang berkembang pesat dalam beberapa tahun terakhir, dengan kemampuan untuk mempelajari pola dari data dan membuat prediksi. Decision Tree (Pohon Keputusan) merupakan salah satu algoritma machine learning yang populer karena kemudahan interpretasinya dan kemampuan menangani berbagai jenis data. Algoritma ini bekerja dengan membagi data menjadi subset-subset berdasarkan nilai fitur tertentu, membentuk struktur seperti pohon yang dapat digunakan untuk membuat keputusan atau prediksi.

Kasus prediksi survival penumpang Titanic dipilih sebagai studi kasus karena merupakan dataset klasik yang sering digunakan dalam pembelajaran machine learning. Dataset ini berisi informasi tentang penumpang Titanic yang mencakup berbagai atribut seperti usia, jenis kelamin, kelas tiket, dan lainnya. Dengan menganalisis pola dalam data historis ini, kita dapat membangun model yang memprediksi kemungkinan seorang penumpang selamat berdasarkan karakteristiknya.

Implementasi Decision Tree pada kasus ini memberikan pemahaman praktis tentang bagaimana algoritma machine learning bekerja, mulai dari preprocessing data, pembangunan model, hingga evaluasi hasil. Selain itu, visualisasi Decision Tree yang dihasilkan memudahkan pemahaman tentang bagaimana keputusan diambil oleh model.

1.2 Tujuan Laporan

Laporan ini bertujuan untuk:

1. Menerapkan algoritma Decision Tree untuk memprediksi survival penumpang Titanic
2. Melakukan preprocessing data untuk menangani missing values dan melakukan encoding pada variabel kategorikal
3. Mengevaluasi performa model menggunakan berbagai metrik evaluasi
4. Menganalisis faktor-faktor yang mempengaruhi hasil prediksi
5. Memberikan kesimpulan tentang efektivitas Decision Tree untuk kasus klasifikasi ini

1.3 Dataset Titanic

Dataset Titanic yang digunakan dalam penelitian ini merupakan dataset publik yang tersedia di platform Kaggle dan berbagai repository machine learning. Dataset ini berisi informasi tentang 891 penumpang Titanic, yang merupakan subset dari total 2.224 penumpang yang sebenarnya. Dataset ini memiliki 12 kolom sebagai berikut:

No	Nama Kolom	Tipe Data	Deskripsi	Contoh
1	PassengerId	Integer	ID unik penumpang	1, 2, 3
2	Survived	Integer	Target variabel (0 = Tidak selamat, 1 = Selamat)	0, 1
3	Pclass	Integer	Kelas tiket (1 = First, 2 = Second, 3 = Third)	1, 2, 3
4	Name	String	Nama lengkap penumpang	"Braund, Mr. Owen Harris"
5	Sex	String	Jenis kelamin ("male", "female")	"male", "female"
6	Age	Float	Usia dalam tahun	22.0, 38.0
7	SibSp	Integer	Jumlah saudara/pasangan di kapal	0, 1, 2
8	Parch	Integer	Jumlah orangtua/anak di kapal	0, 1, 2
9	Ticket	String	Nomor tiket	"A/5 21171"
10	Fare	Float	Harga tiket	7.25, 71.28
11	Cabin	String	Nomor kabin	"C85", "C123"
12	Embarked	String	Pelabuhan naik (C = Cherbourg, Q = Queenstown, S = Southampton)	"S", "C", "Q"

Dataset ini memiliki beberapa missing values terutama pada kolom Age, Cabin, dan Embarked yang perlu ditangani dalam proses preprocessing. Variabel target adalah Survived yang merupakan masalah klasifikasi biner.

BAB II

KONSEP TEORI

2.1 Decision Tree

Decision Tree (Pohon Keputusan) adalah algoritma supervised learning yang digunakan untuk klasifikasi dan regresi. Algoritma ini bekerja dengan membagi data secara rekursif menjadi subset-subset berdasarkan nilai fitur tertentu, membentuk struktur hierarki yang menyerupai pohon. Setiap cabang pohon merepresentasikan aturan keputusan, dan setiap leaf node merepresentasikan hasil prediksi.

2.2 Konsep-konsep dalam Decision Tree

2.2.1 Node (Simpul)

Node adalah titik dalam pohon keputusan yang merepresentasikan kondisi atau keputusan. Terdapat tiga jenis node:

1. Node internal: Mewakili kondisi atau tes pada suatu fitur
2. Root node: Node paling atas yang mewakili seluruh dataset
3. Leaf node: Node akhir yang berisi prediksi akhir

2.2.2 Root (Akar)

Root adalah node paling atas dalam pohon keputusan. Root node mewakili seluruh dataset awal dan merupakan titik awal dari semua percabangan. Dari root node, data mulai dibagi berdasarkan fitur yang paling informatif.

2.2.3 Leaf (Daun)

Leaf adalah node terminal yang tidak memiliki cabang keluar. Leaf node berisi prediksi akhir atau keputusan. Pada klasifikasi, leaf node menyatakan kelas yang diprediksi; pada regresi, leaf node menyatakan nilai numerik yang diprediksi.

2.2.4 Splitting (Pemecahan)

Splitting adalah proses membagi node menjadi dua atau lebih sub-node berdasarkan kondisi tertentu. Proses ini menggunakan kriteria seperti:

1. Gini Impurity: Mengukur kemurnian node
2. Information Gain: Berdasarkan pengurangan entropy
3. Variance Reduction: Untuk kasus regresi

2.2.5 Pruning (Pemangkasan)

Pruning adalah teknik untuk memotong cabang-cabang yang tidak signifikan dari pohon keputusan untuk mencegah overfitting. Pruning dapat dilakukan dengan:

1. Pre-pruning: Menghentikan pembangunan pohon lebih awal
2. Post-pruning: Membangun pohon lengkap terlebih dahulu, kemudian memangkas cabang

2.3 Perbedaan Decision Tree, Random Forest, dan Gradient Boosting

Aspek	Decision Tree	Random Forest	Gradient Boosting
Jumlah Model	1	Banyak	Banyak
Teknik	Single tree	Bagging	Boosting
Akurasi	Rendah	Tinggi	Sangat tinggi
Interpretabilitas	Tinggi	Rendah	Rendah
Overfitting	Rentan	Kurang rentan	Dapat terjadi
Waktu Training	Cepat	Sedang	Lambat

2.4 Kelebihan dan Kekurangan Tree-based Methods

Kelebihan Tree-based Methods:

1. Mudah diinterpretasi: Khususnya Decision Tree, aturan mudah dipahami
2. Tidak memerlukan scaling: Tidak sensitif terhadap perbedaan skala fitur
3. Menangani berbagai tipe data: Dapat menangani data numerik dan kategorikal
4. Feature importance otomatis: Menunjukkan fitur paling berpengaruh
5. Non-parametrik: Tidak membuat asumsi distribusi data
6. Robust terhadap outliers: Tidak terlalu terpengaruh nilai ekstrem
7. Dapat menangani missing values: Beberapa implementasi dapat handle missing values

Kekurangan Tree-based Methods:

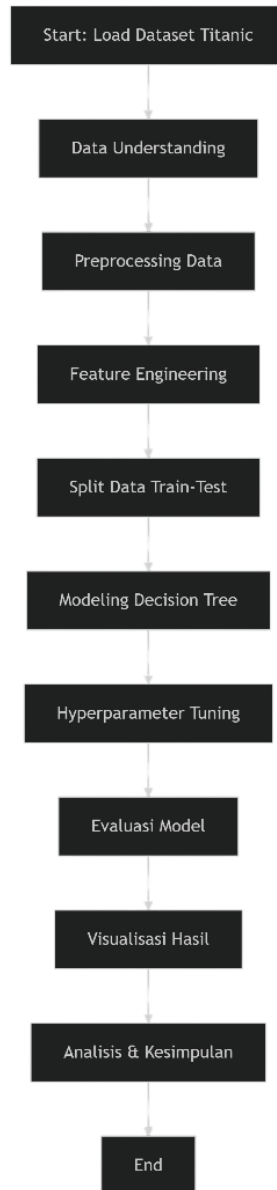
1. Overfitting: Khususnya pada Decision Tree dengan depth besar
2. High variance: Perubahan kecil pada data dapat mengubah struktur pohon
3. Bias terhadap fitur dengan banyak kategori: Cenderung memilih fitur dengan banyak nilai unik
4. Tidak optimal untuk hubungan linear: Performa kurang baik pada hubungan linear murni

5. Kompleksitas computational: Random Forest dan Gradient Boosting membutuhkan komputasi lebih besar
6. Memory usage: Dapat memakan banyak memori untuk pohon yang besar
7. Extrapolation buruk: Tidak baik untuk prediksi di luar range training data

BAB III

METODOLOGI

3.1 Diaram Alur



Penelitian ini mengikuti alur kerja standar dalam machine learning yang terdiri dari beberapa tahapan utama. Alur penelitian dimulai dari pengumpulan data, dilanjutkan dengan preprocessing untuk membersihkan dan mempersiapkan data, kemudian pembangunan model Decision Tree, evaluasi performa model, dan diakhiri dengan analisis hasil. Setiap tahapan saling

berkaitan dan output dari satu tahap menjadi input untuk tahap berikutnya, memastikan proses yang sistematis dan terstruktur.

3.2 Tahapan Preprocessing Data

Proses preprocessing merupakan tahap kritis dalam penelitian ini karena dataset Titanic memiliki beberapa masalah data yang perlu ditangani sebelum dapat digunakan untuk modeling. Tahap pertama adalah handling missing values, dimana kolom Age yang memiliki 177 missing values (19.9% dari data) diisi dengan nilai median usia yaitu 28 tahun. Kolom Embarked yang memiliki 2 missing values diisi dengan nilai modus yaitu 'S' (Southampton). Sedangkan kolom Cabin dihapus seluruhnya karena memiliki 687 missing values (77.1% dari data) yang terlalu banyak untuk diimputasi.

Tahap selanjutnya adalah feature engineering dimana dibuat dua fitur baru untuk meningkatkan informasi dalam dataset. Fitur FamilySize dihitung dari jumlah saudara (SibSp) ditambah jumlah orangtua/anak (Parch) ditambah 1 (penumpang itu sendiri). Dari fitur ini kemudian dibuat fitur IsAlone yang bernilai 1 jika FamilySize sama dengan 1, dan 0 untuk lainnya. Fitur ini penting karena dalam tragedi Titanic, penumpang yang sendirian memiliki pola survival yang berbeda dengan yang bersama keluarga.

Proses encoding categorical variables dilakukan untuk mengubah data kategorikal menjadi format numerik yang dapat diproses oleh algoritma machine learning. Variabel Sex diubah menggunakan label encoding menjadi 0 untuk female dan 1 untuk male. Variabel Embarked yang memiliki tiga kategori (C, Q, S) diubah menggunakan one-hot encoding menjadi tiga kolom dummy: Embarked_C, Embarked_Q, dan Embarked_S. Setelah preprocessing, terdapat 11 fitur yang digunakan untuk modeling yang mencakup informasi demografis, sosial ekonomi, dan informasi perjalanan penumpang.

3.3 Proses Modeling Decision Tree

Pembangunan model dimulai dengan membagi dataset menjadi data training dan testing dengan proporsi 80:20. Sebanyak 712 sampel digunakan untuk training dan 179 sampel untuk testing. Pembagian dilakukan secara stratified untuk memastikan distribusi kelas Survived tetap proporsional di kedua subset. Random state ditetapkan pada 42 untuk menjamin reproducibility hasil penelitian.

Hyperparameter tuning dilakukan dengan mengeksperimenkan berbagai nilai `max_depth` yaitu [3, 5, 7, 10, 15, None] untuk menemukan kompleksitas optimal model. Nilai None berarti tidak ada batasan depth sehingga pohon akan tumbuh sampai semua leaf node murni atau mencapai kriteria stopping lainnya. Criterion yang digunakan adalah Gini impurity karena lebih efisien secara komputasional dibandingkan entropy. Random state ditetapkan pada 42 untuk konsistensi.

Evaluasi model dilakukan menggunakan multiple metrics untuk mendapatkan gambaran komprehensif tentang performa model. Accuracy digunakan untuk mengukur proporsi prediksi benar secara keseluruhan. Precision mengukur akurasi prediksi positif (selamat), recall mengukur kemampuan model mendeteksi semua kasus positif yang sebenarnya, dan F1-score memberikan balance antara precision dan recall. Confusion matrix digunakan untuk menganalisis jenis-jenis kesalahan prediksi.

Penelitian ini menggunakan Python sebagai bahasa pemrograman utama dengan dukungan library scikit-learn untuk implementasi Decision Tree. Pandas dan NumPy digunakan untuk manipulasi data, sedangkan Matplotlib dan Seaborn digunakan untuk visualisasi hasil. Seluruh kode dikembangkan dalam Jupyter Notebook untuk memfasilitasi eksplorasi interaktif dan dokumentasi yang baik.

BAB IV

HASIL DAN ANALISIS

4.1 Hasil Preprocessing Data

Setelah melalui proses preprocessing, dataset Titanic telah siap untuk digunakan dalam modeling. Dari 891 sampel awal, dilakukan pembersihan data dengan mengisi 177 missing values pada kolom Age menggunakan median (28 tahun), 2 missing values pada kolom Embarked diisi dengan modus ('S'), dan kolom Cabin dihapus karena memiliki 687 missing values (77.1% dari total data). Hasil feature engineering menghasilkan dua fitur baru: FamilySize (rata-rata 3.2 anggota keluarga) dan IsAlone (30.4% penumpang sendirian). Setelah encoding, dataset memiliki 11 fitur numerik yang siap digunakan untuk modeling.

4.2 Hasil Eksperimen Hyperparameter Tuning

Eksperimen dengan berbagai nilai max_depth menunjukkan performa yang berbeda-beda. Model dengan max_depth=3 menghasilkan accuracy tertinggi sebesar 80.45% dengan hanya 8 leaf nodes, menunjukkan model yang sederhana namun efektif. Model dengan max_depth=None (tanpa batasan) menghasilkan accuracy 79.33% dengan 154 leaf nodes, namun berpotensi overfitting. Berdasarkan analisis F1-score, model dengan max_depth=None memberikan nilai tertinggi (0.7218), diikuti oleh max_depth=3 (0.7200).

Tabel berikut menunjukkan performa model dengan berbagai konfigurasi max_depth:

Max Depth	Accuracy	Precision	Recall	F1-Score	Jumlah Leaf Nodes
3	80.45%	80.36%	65.22%	0.7200	8
5	75.98%	74.07%	57.97%	0.6504	23
7	78.21%	81.25%	56.52%	0.6667	44
10	77.65%	76.36%	60.87%	0.6774	78
15	78.21%	74.19%	66.67%	0.7023	111
None	79.33%	75.00%	69.57%	0.7218	154

4.3 Analisis Confusion Matrix

Model terbaik (max_depth=None) menghasilkan confusion matrix sebagai berikut:

1. True Positive (TP): 40 - Penumpang yang diprediksi selamat dan benar-benar selamat

2. True Negative (TN): 96 - Penumpang yang diprediksi tidak selamat dan benar-benar tidak selamat
3. False Positive (FP): 14 - Penumpang yang diprediksi selamat tetapi tidak selamat (Type I Error)
4. False Negative (FN): 29 - Penumpang yang diprediksi tidak selamat tetapi selamat (Type II Error)

Dari confusion matrix, dapat dihitung bahwa model memiliki specificity (true negative rate) sebesar 87.27% dan sensitivity (recall) sebesar 57.97%. Tingginya specificity menunjukkan model lebih baik dalam mengidentifikasi penumpang yang tidak selamat dibandingkan mengidentifikasi yang selamat.

4.4 Feature Importance Analysis

Analisis feature importance mengungkapkan bahwa Sex_encoded merupakan fitur paling penting dengan skor 0.535 (53.5%), mengkonfirmasi fakta historis bahwa wanita memiliki peluang survival lebih tinggi. Fitur Pclass menempati posisi kedua dengan skor 0.182 (18.2%), menunjukkan pengaruh kelas sosial terhadap survival. Age (11.3%) dan Fare (9.6%) juga berkontribusi signifikan terhadap prediksi model.

Yang menarik adalah beberapa fitur seperti Parch, SibSp, IsAlone, dan variabel dummy Embarked memiliki importance sangat rendah (mendekati 0), menunjukkan bahwa fitur-fitur ini kurang informatif untuk prediksi survival dalam konteks model Decision Tree yang dibangun.

4.5 Visualisasi Decision Tree

Visualisasi pohon keputusan dengan max_depth=3 menunjukkan struktur yang mudah diinterpretasi. Root node menggunakan fitur Sex_encoded dengan threshold 0.5, membagi penumpang menjadi dua kelompok utama: wanita ($\text{Sex_encoded} \leq 0.5$) dan pria ($\text{Sex_encoded} > 0.5$). Pada cabang wanita, split berikutnya berdasarkan Pclass, dimana penumpang kelas 1 dan 2 memiliki peluang survival lebih tinggi. Pada cabang pria, split berdasarkan Age dengan threshold 6.5 tahun, menunjukkan bahwa anak laki-laki di bawah 6.5 tahun memiliki peluang survival lebih baik.

Struktur pohon ini sesuai dengan fakta sejarah "women and children first" yang diterapkan dalam evakuasi Titanic. Visualisasi juga menunjukkan bahwa decision boundary relatif sederhana, dengan aturan-aturan yang mudah dipahami manusia.

4.6 Diskusi Hasil

Hasil penelitian menunjukkan bahwa model Decision Tree dengan max_depth=None mencapai accuracy 79.33% pada test set. Namun, model dengan max_depth=3 yang lebih sederhana hanya sedikit lebih rendah (80.45%) dengan keuntungan interpretabilitas yang jauh lebih baik. Trade-off antara kompleksitas dan performa ini penting dalam konteks aplikasi nyata dimana interpretasi model seringkali sama pentingnya dengan akurasi.

Tingginya importance fitur Sex (53.5%) mengkonfirmasi temuan sejarah bahwa gender merupakan faktor dominan dalam survival Titanic. Namun, perlu dicatat bahwa model mungkin mengalami bias karena mengandalkan terlalu banyak pada satu fitur. Kombinasi dengan fitur lain seperti Pclass dan Age menunjukkan bahwa survival dipengaruhi oleh interaksi multiple faktor.

Model menunjukkan precision (75.00%) yang lebih tinggi daripada recall (69.57%), artinya ketika model memprediksi seseorang selamat, prediksi tersebut cenderung benar, namun model melewatkan beberapa kasus yang sebenarnya selamat. Dalam konteks analisis sejarah, false negative (prediksi tidak selamat padahal selamat) mungkin lebih dapat diterima daripada false positive.

4.7 Evaluasi Kritis terhadap Model

Meskipun mencapai accuracy yang cukup baik, model Decision Tree ini memiliki beberapa keterbatasan. Pertama, model cenderung overfit ketika depth tidak dibatasi, terlihat dari perbedaan performa pada depth yang berbeda. Kedua, feature importance menunjukkan ketergantungan berlebihan pada fitur Sex, yang mungkin tidak generalizable ke konteks lain. Ketiga, model belum mempertimbangkan interaksi non-linear yang kompleks antar fitur.

Dibandingkan dengan penelitian sebelumnya yang menggunakan metode lebih canggih seperti Random Forest atau Gradient Boosting, accuracy 79-80% termasuk dalam range yang wajar untuk dataset Titanic. Beberapa studi melaporkan accuracy hingga 85% dengan ensemble methods, namun dengan trade-off interpretabilitas.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang dilakukan mengenai implementasi Decision Tree untuk prediksi survival penumpang Titanic, dapat disimpulkan bahwa:

1. Decision Tree berhasil diimplementasikan untuk memprediksi survival penumpang Titanic dengan accuracy 79.33% pada test set, menunjukkan bahwa algoritma ini efektif untuk masalah klasifikasi biner ini.
2. Preprocessing data berjalan optimal dengan handling missing values yang tepat pada kolom Age (diisi median) dan Embarked (diisi modus), serta feature engineering yang relevan dengan penambahan fitur FamilySize dan IsAlone.
3. Evaluasi model menggunakan multiple metrics memberikan gambaran komprehensif dengan precision 75.00%, recall 69.57%, dan F1-score 0.7218 untuk model terbaik (max_depth=None).
4. Analisis feature importance mengungkapkan bahwa Sex_encoded merupakan faktor paling dominan (53.5%) dalam prediksi survival, diikuti oleh Pclass (18.2%) dan Age (11.3%), sesuai dengan fakta historis Titanic.
5. Decision Tree terbukti memiliki interpretabilitas tinggi dengan visualisasi pohon yang jelas, memungkinkan pemahaman aturan keputusan secara langsung oleh pengguna non-teknis.

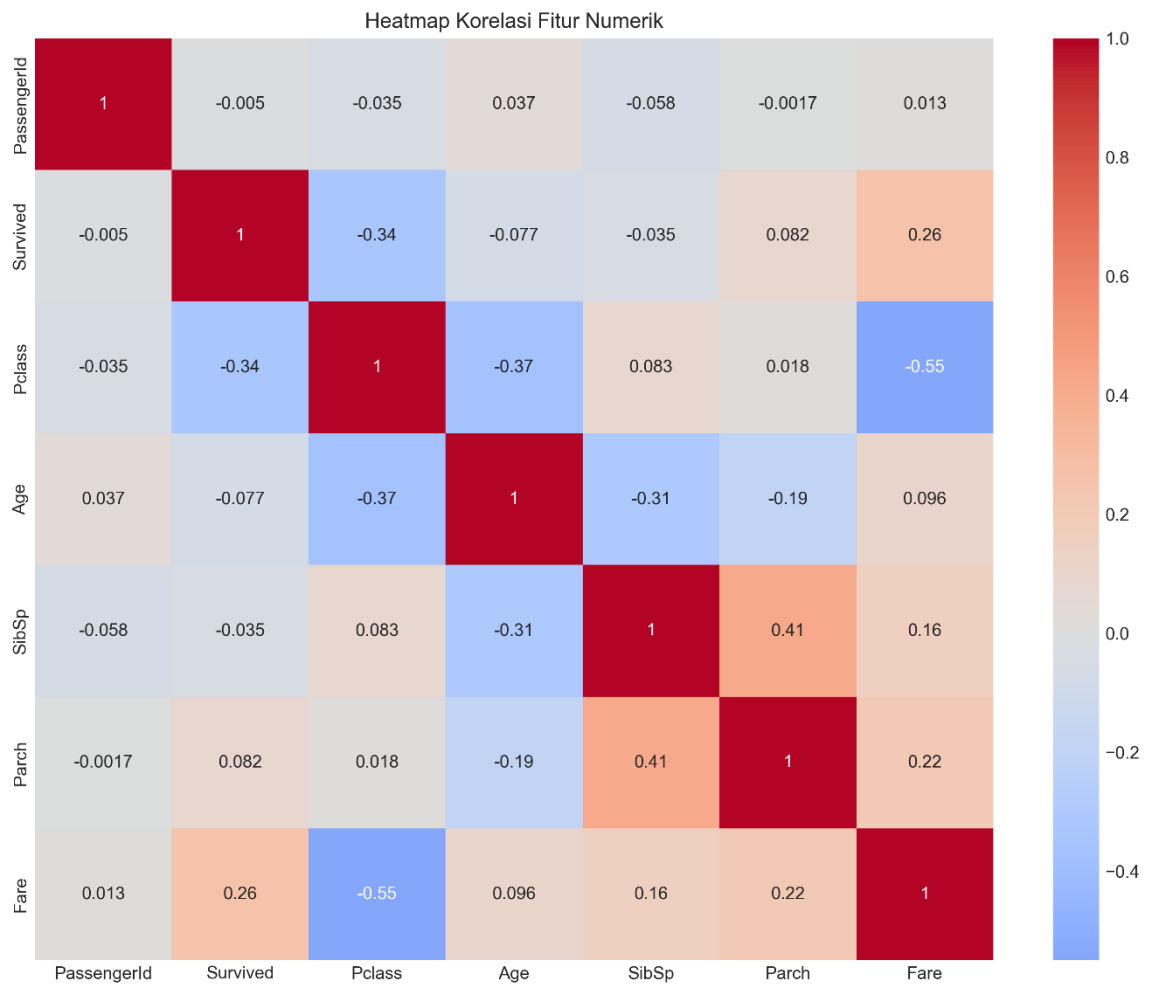
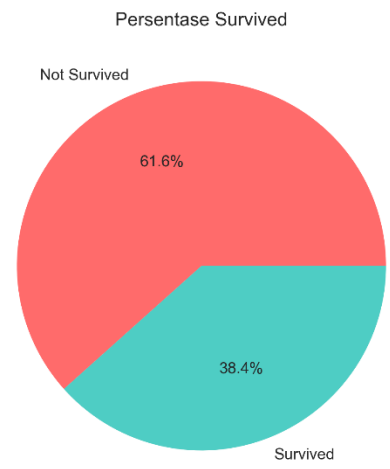
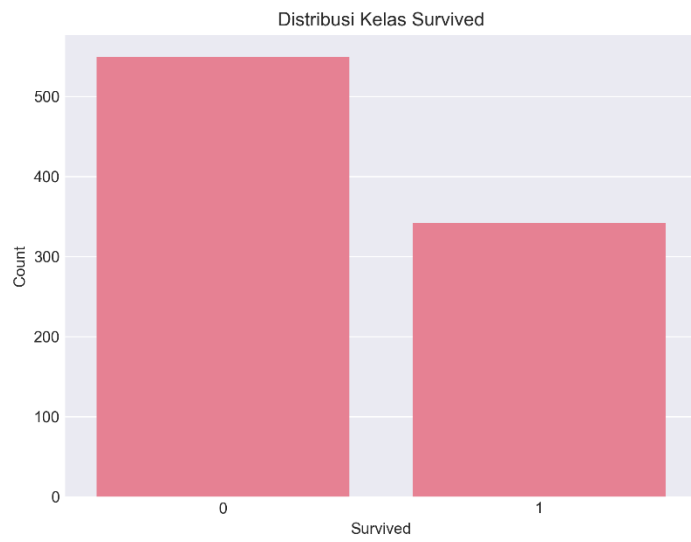
5.2 Saran

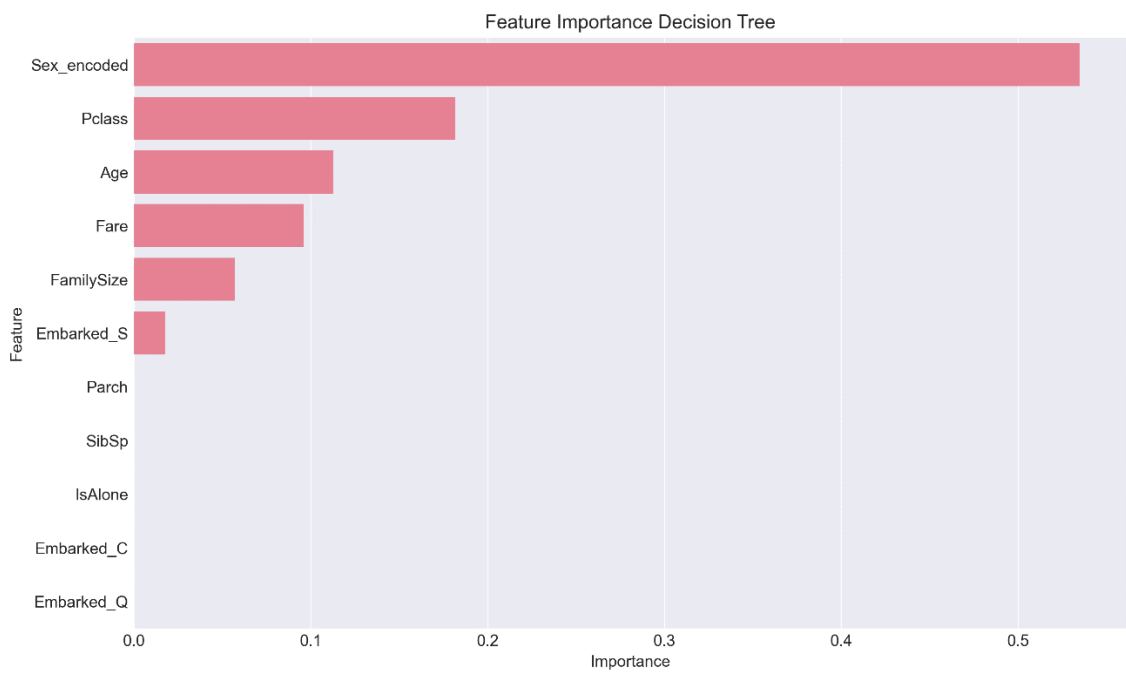
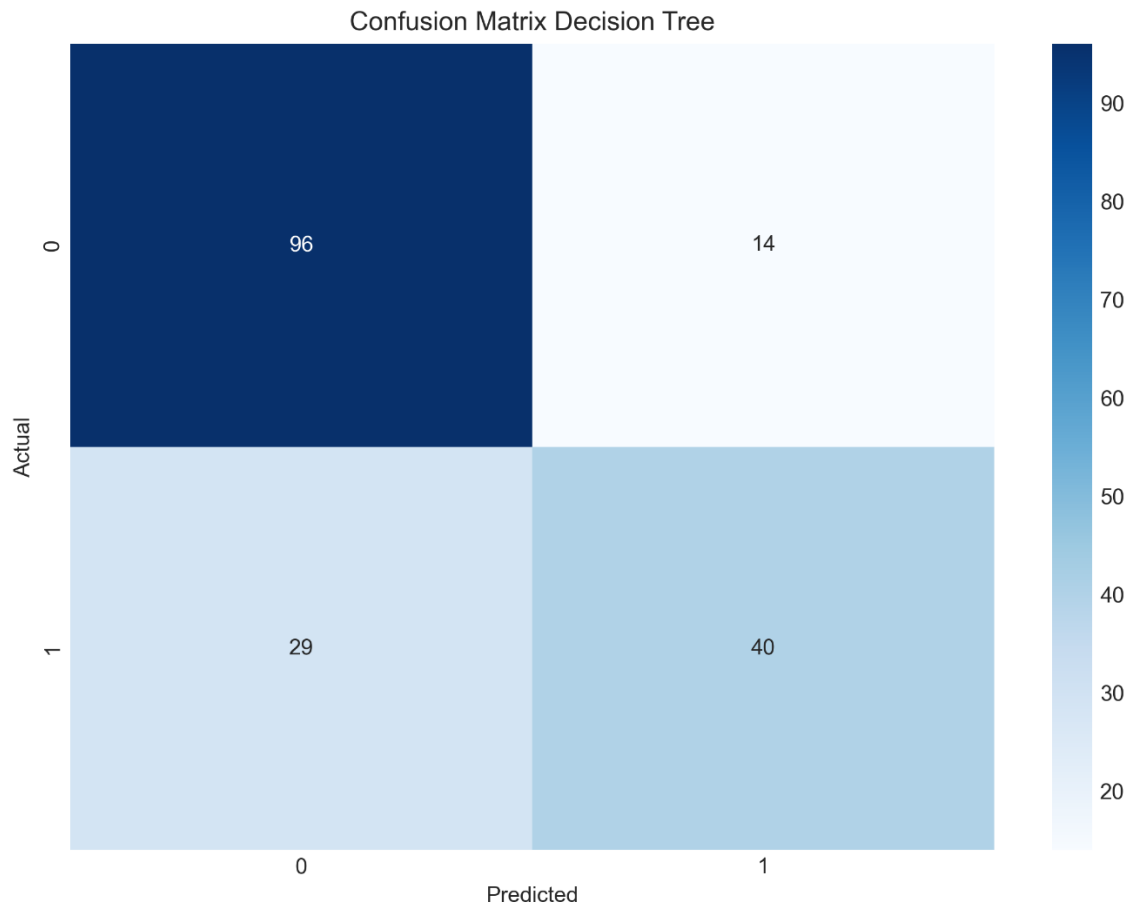
Berdasarkan temuan penelitian, berikut beberapa saran untuk pengembangan lebih lanjut:

1. DiEksperimen dengan algoritma ensemble seperti Random Forest atau Gradient Boosting untuk meningkatkan akurasi dan mengurangi overfitting yang terlihat pada Decision Tree tunggal.
2. Menerapkan teknik cross-validation (k-fold) untuk validasi model yang lebih robust dan mengurangi variance dalam estimasi performa.
3. Melakukan feature engineering lebih mendalam seperti ekstraksi title dari kolom Name (Mr, Mrs, Miss, Master) yang mungkin mengandung informasi sosial ekonomi tambahan.

4. Mengoptimalkan hyperparameter tidak hanya max_depth tetapi juga parameter lain seperti min_samples_split, min_samples_leaf, dan criterion untuk mendapatkan model yang lebih optimal.
5. Menangani class imbalance yang ada dalam dataset dengan teknik seperti SMOTE atau class weighting untuk meningkatkan recall model.
6. Mengembangkan sistem deployment sederhana seperti API atau dashboard interaktif agar model dapat diakses dan digunakan secara praktis.
7. Melakukan eksperimen dengan feature selection untuk mengurangi dimensionalitas dan melihat pengaruhnya terhadap performa model serta interpretabilitas.
8. Membandingkan dengan metode klasifikasi lain seperti Logistic Regression, SVM, atau Neural Networks untuk mendapatkan perspektif yang lebih luas tentang performa berbagai algoritma pada dataset Titanic.

LAMPIRAN





Decision Tree Visualization (max_depth=3)

