

Kelas A

**UJIAN AKHIR SEMESTER
MODERN PREDICTION & MACHINE LEARNING**



Disusun oleh:

Muhammad Ahadian Purnama (23611038)

Dosen Pengampu:

Raden Bagus Fajriya Hakim, Dr., S.Si., M.Si.

**JURUSAN STATISTIKA FAKULTAS MATEMATIKA DAN ILMU
PENGETAHUAN ALAM UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2025**

Daftar Isi

Daftar Isi	ii
Daftar Tabel	iv
Daftar Gambar	v
Tahap 1	1
1.1 Pemilihan Dataset.....	1
1.1.1 Judul Dataset.....	1
1.1.2 Sumber	1
1.1.3 Deskripsi Dataset	1
1.1.4 Fitur-fitur.....	1
1.1.5 Alasan Pemilihan Dataset	1
1.2 Pemilihan Dataset.....	2
1.2.1 Struktur Dataset	2
1.2.2 Distribusi Harga (Price)	2
1.2.3 Boxplot Harga	2
1.2.4 Distribusi Target.....	3
1.2.5 Rata-rata Harga per Kategori Menu.....	4
1.3 Hipotesis Awal.....	4
Tahap 2	5
2.1 Tujuan.....	5
2.2 Nilai Hilang	5
2.3 Encoding Variabel Kategorikal	5
2.3.1 Profitability	5
2.3.2 MenuCategory.....	5
2.4 Standarisasi Fitur Numerik.....	5
2.5 Pemisahan Data: Train dan Test Set.....	5
2.6 Hasil dan Distribusi Data	6
2.7 Kesimpulan Sementara.....	6
Tahap 3	7
3.1 Pendahuluan	7
3.2 Model yang Digunakan	7
3.3 Hasil Evaluasi.....	7
3.3.1 <i>Decision Tree</i>	7
3.3.2 <i>Random Forest</i>	8
3.3.3 <i>Logistic Regression</i>	8
3.4 Kesimpulan Sementara.....	9
Tahap 4	10
4.1 <i>Logistic Regression</i>	10
4.2 <i>Random Forest</i>	10
4.3 <i>Decision Tree</i>	11
4.4 Kesimpulan Sementara.....	12
Tahap 5	13
5.1 Analisis Diagnostik	13
5.1.1 Distribusi Kesalahan	13
5.1.2 Keseimbangan Data	13
5.1.3 Akurasi vs Kelas Minoritas.....	13
5.2 Interpretasi Model	13

5.2.1	Fitur Penting.....	13
5.2.2	Interpretasi Hasil	14
5.2.3	Keterbatasan Model	14
5.3	Kesimpulan Sementara.....	14
Tahap 6	15
6.1	Optimasi Menu Berdasarkan Profitabilitas	15
6.1.1	Prioritaskan Menu Kategori <i>Main Course & Dessert</i>	15
6.1.2	Revisi Menu dengan Harga Rendah	15
6.1.3	Evaluasi Menu yang Sulit Diprediksi	15
6.2	Rekomendasi Pengembangan Menu Baru.....	15
6.2.1	Gunakan Pola Menu <i>High Profitability</i>	15
6.3	Strategi Penetapan Harga	15
6.3.1	Segmentasi Harga Lebih Strategis	15
6.4	Tindak Lanjut	16
Kesimpulan Akhir	17

Daftar Tabel

Tabel 1. 1	Tabel Fitur-fitur.....	1
-------------------	------------------------	---

Daftar Gambar

Gambar 1. 1 Histogram Distribusi Harga Menu.....	2
Gambar 1. 2 <i>Boxplot</i> Harga Menu.....	3
Gambar 1. 3 Distribusi Kelas Target	3
Gambar 1. 4 Rata-rata Harga per Kategori Menu.....	4
Gambar 2. 1 Hasil Pembagian Data.....	6
Gambar 3. 1 <i>Decision Tree</i>	7
Gambar 3. 2 <i>Random Forest</i>	8
Gambar 3. 3 <i>Logistic Regression</i>	8
Gambar 4. 1 <i>Confusion Matrix: Logistic Regression</i>	10
Gambar 4. 2 <i>Confusion Matrix: Random Forest</i>	11
Gambar 4. 3 <i>Confusion Matrix: Decision Tree</i>	11

Tahap 1

1.1 Pemilihan Dataset

1.1.1 Judul Dataset

Restaurant Menu Optimization Dataset

1.1.2 Sumber

Kaggle: <https://www.kaggle.com/datasets/rabieelkharoua/predict-restaurant-menu-items-profitability>

1.1.3 Deskripsi Dataset

Dataset ini berisi informasi mengenai berbagai item menu dari beberapa restoran, termasuk kategori makanan, daftar bahan, harga jual, dan tingkat profitabilitasnya (*Low*, *Medium*, atau *High*). Dataset ini cocok untuk tugas supervised learning (klasifikasi) karena kolom target *Profitability* merupakan label kategorikal yang ingin diprediksi.

1.1.4 Fitur-fitur

Tabel 1. 1 Tabel Fitur-fitur

Kolom	Tipe	Deskripsi
RestaurantID	Kategorikal	ID dari restoran
MenuCategory	Kategorikal	Jenis kategori menu (contoh: Appetizers, Desserts)
MenuItem	Kategorikal	Nama menu makanan
Ingredients	String (list)	Daftar bahan yang digunakan
Price	Numerik (float)	Harga jual produk
Profitability	Kategorikal (Target)	Tingkat keuntungan: Low / Medium / High

1.1.5 Alasan Pemilihan Dataset

- Kasus nyata yang aplikatif di dunia bisnis makanan dan restoran.
- Cocok untuk klasifikasi multi-kelas.
- Kombinasi fitur kategorikal dan numerik, ideal untuk eksplorasi preprocessing dan model machine learning.
- Tidak terlalu besar, cocok untuk pengerjaan proyek UAS take-home.

1.2 Pemilihan Dataset

1.2.1 Struktur Dataset

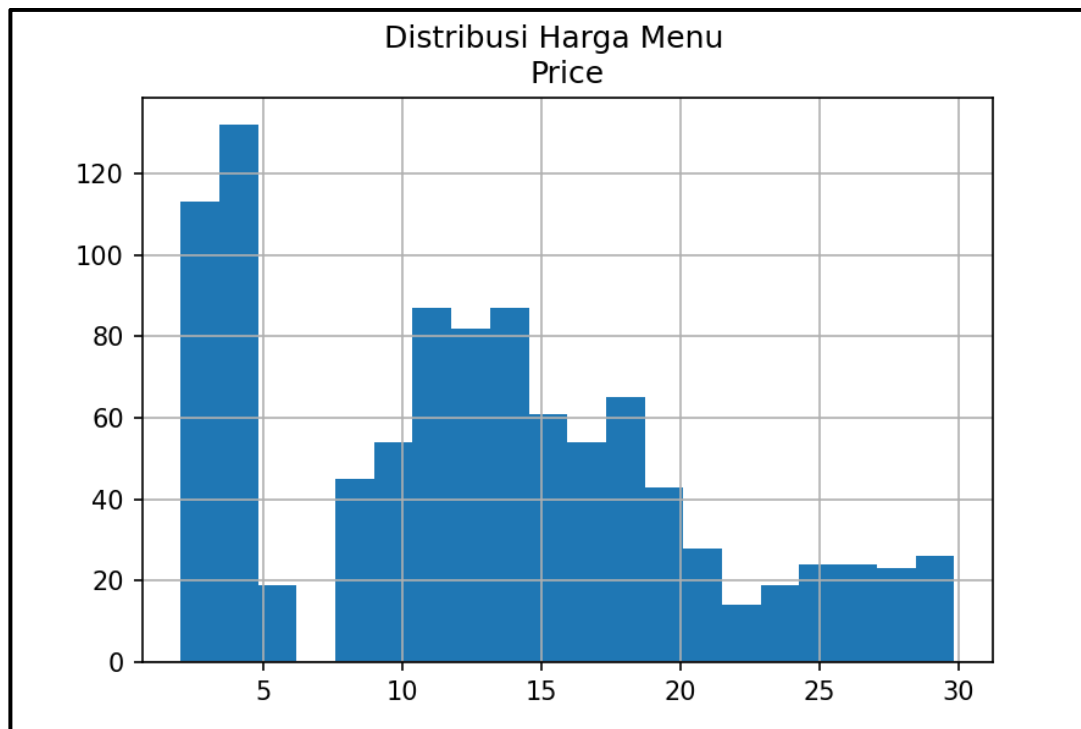
- Jumlah data: 1000 baris
- Jumlah kolom: 6
- Tidak ada nilai yang hilang

```
print(df.info())  
print(df.isnull().sum())
```

1.2.2 Distribusi Harga (Price)

- Histogram menunjukkan sebagian besar harga berada di rentang 2–10
- Ada outlier dengan harga di atas 25

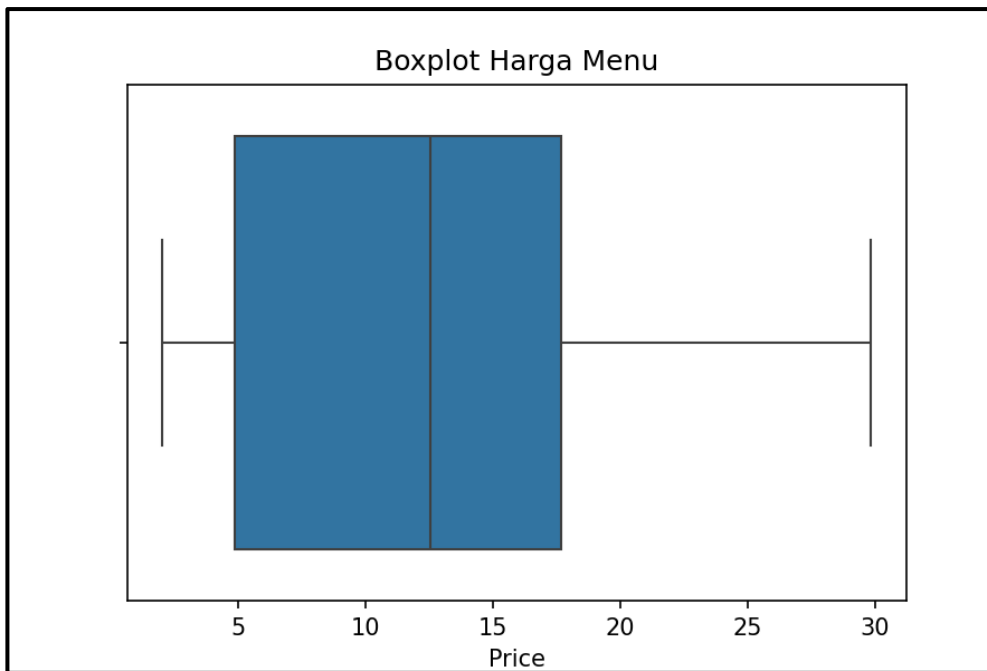
```
df['Price'].hist()
```



Gambar 1. 1 Histogram Distribusi Harga Menu

1.2.3 Boxplot Harga

Visualisasi memperlihatkan persebaran harga dan outlier dengan jelas.

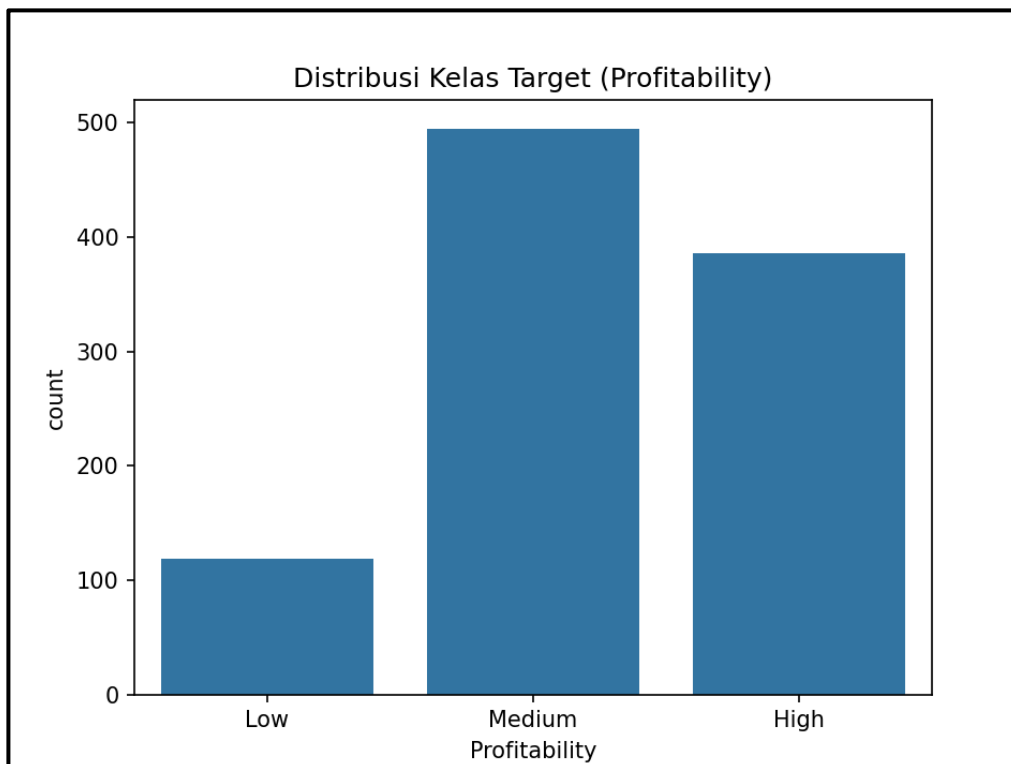


Gambar 1. 2 *Boxplot* Harga Menu

1.2.4 Distribusi Target

- Kelas Medium dan High mendominasi
- Distribusi target agak tidak seimbang, tapi masih bisa ditangani

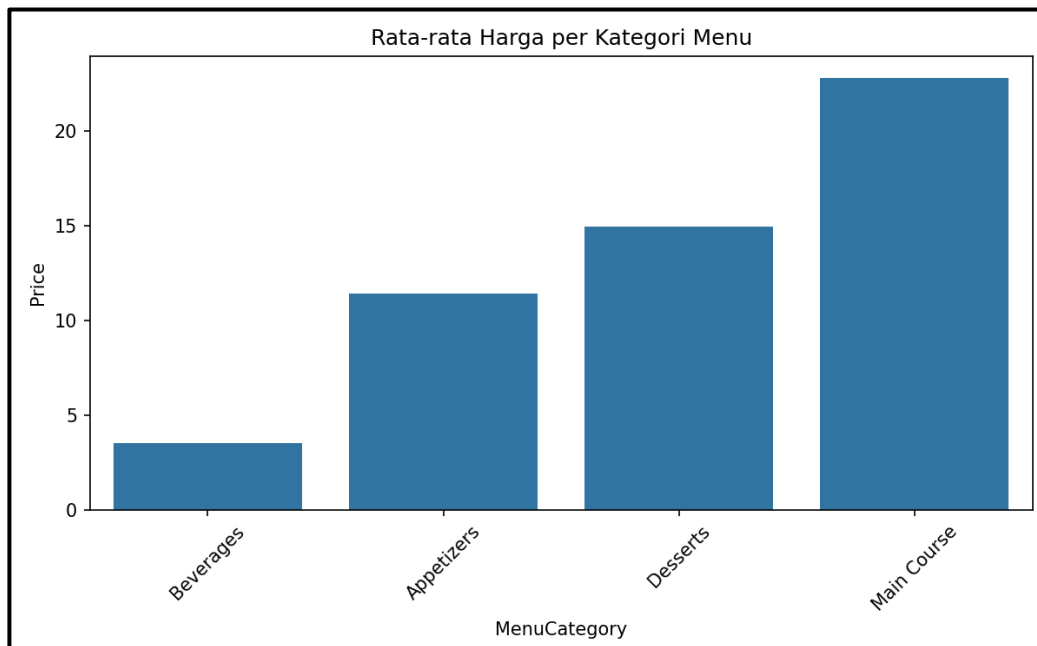
```
sns.countplot(x='Profitability', data=df)
```



Gambar 1. 3 Distribusi Kelas Target

1.2.5 Rata-rata Harga per Kategori Menu

Menu utama (Main Course) cenderung memiliki harga yang lebih tinggi dibandingkan kategori lain seperti Beverages atau Appetizers.



Gambar 1. 4 Rata-rata Harga per Kategori Menu

1.3 Hipotesis Awal

Hipotesis: Menu dengan harga yang lebih tinggi dan berasal dari kategori makanan utama cenderung memiliki tingkat keuntungan (*Profitability*) yang tinggi.

Tahap 2

2.1 Tujuan

Tahap ini bertujuan untuk:

- Mengubah variabel kategorikal menjadi numerik.
- Menstandarkan nilai numerik.
- Memastikan tidak ada nilai hilang.
- Membagi data menjadi training dan testing set.

2.2 Nilai Hilang

Pemeriksaan awal dilakukan terhadap dataset untuk memastikan tidak ada nilai hilang (missing values). Hasilnya, seluruh kolom terisi penuh sebanyak 1000 baris data.

2.3 Encoding Variabel Kategorikal

2.3.1 Profitability

Variabel Profitability merupakan data kategorikal ordinal dengan urutan:

Low < Medium < High

Karena memiliki urutan yang jelas, digunakan OrdinalEncoder:

```
from sklearn.preprocessing import OrdinalEncoder
```

```
data['Profitability'] = OrdinalEncoder(categories=[['Low', 'Medium', 'High']]).fit_transform(data[['Profitability']])
```

2.3.2 MenuCategory

Variabel ini merupakan kategorikal nominal. Maka digunakan one-hot encoding untuk menghindari asumsi urutan:

```
data = pd.get_dummies(data, columns=['MenuCategory'], drop_first=True)
```

2.4 Standarisasi Fitur Numerik

Fitur numerik seperti Price memiliki rentang berbeda, sehingga perlu distandarkan menggunakan StandardScaler:

```
from sklearn.preprocessing import StandardScaler
```

```
scaler = StandardScaler()
```

```
data[['Price']] = scaler.fit_transform(data[['Price']])
```

2.5 Pemisahan Data: Train dan Test Set

Data dipisahkan menjadi 80% training dan 20% testing menggunakan train_test_split dengan stratifikasi berdasarkan Profitability.

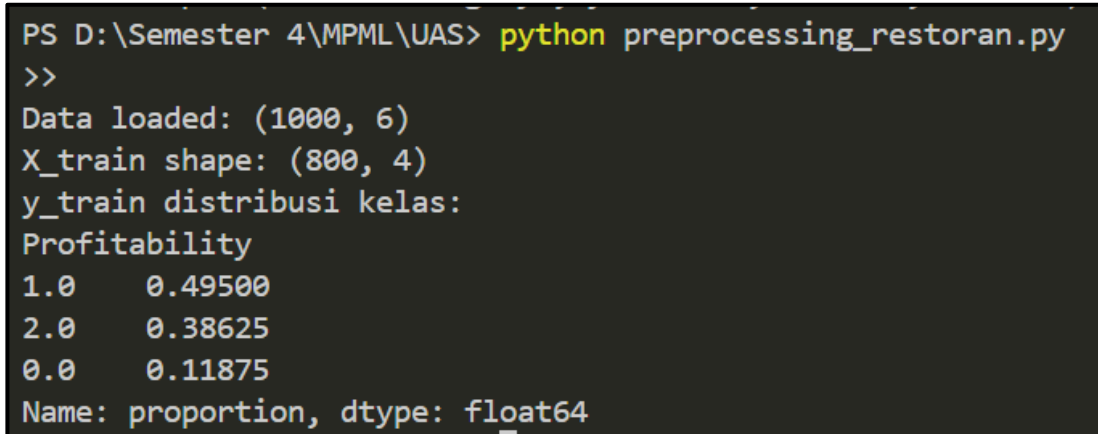
```
from sklearn.model_selection import train_test_split
```

```
X = data.drop('Profitability', axis=1)
y = data['Profitability']

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, stratify=y, random_state=42
)
```

2.6 Hasil dan Distribusi Data

Berikut hasil pembagian data yang ditampilkan melalui program:



```
PS D:\Semester 4\MPML\UAS> python preprocessing_restoran.py
>>
Data loaded: (1000, 6)
X_train shape: (800, 4)
y_train distribusi kelas:
Profitability
1.0    0.49500
2.0    0.38625
0.0    0.11875
Name: proportion, dtype: float64
```

Gambar 2. 1 Hasil Pembagian Data

2.7 Kesimpulan Sementara

Berdasarkan hasil preprocessing, seluruh data telah diproses dengan baik tanpa adanya nilai yang hilang. Variabel kategorikal seperti *Profitability* telah berhasil diubah menjadi numerik menggunakan metode *ordinal encoding*, sedangkan *MenuCategory* dikonversi menggunakan *one-hot encoding* untuk menghindari asumsi urutan. Fitur numerik seperti *Price* juga telah distandarisasi agar berada dalam skala yang seragam. Selanjutnya, data berhasil dipisahkan menjadi data latih dan data uji dengan proporsi 80:20 menggunakan teknik *stratified split*, sehingga distribusi kelas *Profitability* tetap seimbang di kedua subset. Dengan preprocessing ini, data telah siap digunakan untuk tahap selanjutnya yaitu pelatihan model pada tahap 3.

Tahap 3

3.1 Pendahuluan

Setelah tahap preprocessing selesai, dilakukan pelatihan beberapa model klasifikasi untuk memprediksi tingkat *Profitability* (keuntungan) dari menu restoran. Fitur yang digunakan dalam pemodelan adalah `Price` dan `MenuCategory`, dengan target `Profitability` yang sudah dikodekan secara ordinal (0 = Low, 1 = Medium, 2 = High).

3.2 Model yang Digunakan

Tiga model yang dipilih untuk dibandingkan performanya adalah:

- *Decision Tree Classifier*
- *Random Forest Classifier*
- *Logistic Regression*

Data dibagi menjadi 80% data latih dan 20% data uji menggunakan teknik *stratified split* untuk menjaga proporsi kelas target. Fitur `MenuCategory` diproses menggunakan teknik *One-Hot Encoding*.

3.3 Hasil Evaluasi

3.3.1 *Decision Tree*

- **Akurasi:** 82%
- **F1-Score per kelas:**
 - *Low*: 0.55
 - *Medium*: 0.87
 - *High*: 0.84

```
===== Decision Tree =====
[[13  6  5]
 [ 5 87  7]
 [ 5  8 64]]
```

	precision	recall	f1-score	support
0	0.57	0.54	0.55	24
1	0.86	0.88	0.87	99
2	0.84	0.83	0.84	77
accuracy			0.82	200
macro avg	0.76	0.75	0.75	200
weighted avg	0.82	0.82	0.82	200

Gambar 3.1 *Decision Tree*

- **Kesimpulan:** Performa cukup baik terutama untuk kelas *Medium* dan *High*. Namun, performa untuk kelas *Low* masih rendah.

3.3.2 Random Forest

- **Akurasi:** 84%
- **F1-Score per kelas:**
 - *Low*: 0.59
 - *Medium*: 0.89
 - *High*: 0.84

```
===== Random Forest =====
```

[[13 6 5]				
[2 91 6]				
[5 8 64]]				
	precision	recall	f1-score	support
0	0.65	0.54	0.59	24
1	0.87	0.92	0.89	99
2	0.85	0.83	0.84	77
accuracy			0.84	200
macro avg	0.79	0.76	0.78	200
weighted avg	0.84	0.84	0.84	200

Gambar 3. 2 *Random Forest*

- **Kesimpulan:** *Random Forest* memberikan performa terbaik secara keseluruhan, dengan peningkatan *recall* dan *precision* pada semua kelas dibanding *Decision Tree*.

3.3.3 Logistic Regression

- **Akurasi:** 82%
- **F1-Score per kelas:**
 - *Low*: 0.00 (model gagal memprediksi kelas ini)
 - *Medium*: 0.87
 - *High*: 0.89

```
===== Logistic Regression =====
```

[[0 18 6]
[0 93 6]
[0 5 72]]

Gambar 3. 3 *Logistic Regression*

- **Kesimpulan:** Meskipun model ini sangat baik untuk kelas *Medium* dan *High*, ia sama sekali gagal mengenali kelas *Low*, sehingga tidak direkomendasikan.

3.4 Kesimpulan Sementara

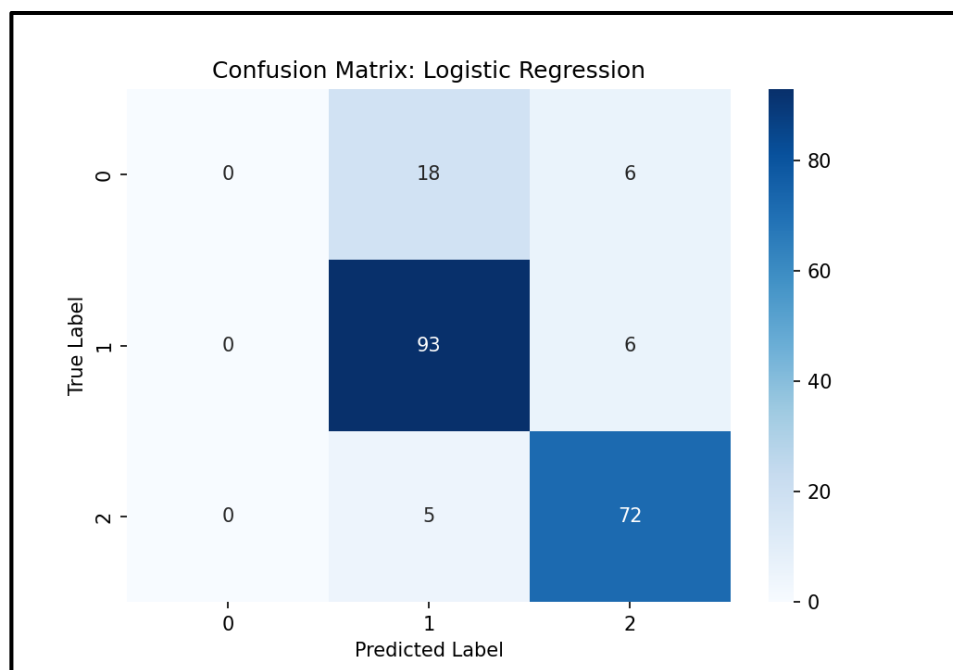
Berdasarkan evaluasi, model *Random Forest Classifier* memberikan hasil paling optimal dengan akurasi 84% dan distribusi *f1-score* yang cukup seimbang. Kelas "*Low*" memang cenderung sulit dikenali, kemungkinan karena jumlah data yang relatif kecil. Model ini akan menjadi kandidat utama untuk tahap selanjutnya (pengujian lanjutan atau *deployment*).

Tahap 4

Pada tahap ini, performa masing-masing model dievaluasi menggunakan *confusion matrix*, yang memberikan gambaran seberapa akurat model memprediksi masing-masing kelas *profitabilitas* ($Low = 0$, $Medium = 1$, $High = 2$). Berikut hasil evaluasinya:

4.1 Logistic Regression

- Kelas 0 (*Low Profitability*): Tidak ada *instance* yang berhasil diprediksi dengan benar (*precision* dan *recall* = 0).
- Kelas 1 (*Medium*): Model sangat baik dalam memprediksi kelas ini (93/99 benar, *recall* $\approx 94\%$).
- Kelas 2 (*High*): Performa cukup baik (72/77 benar, *recall* $\approx 94\%$).

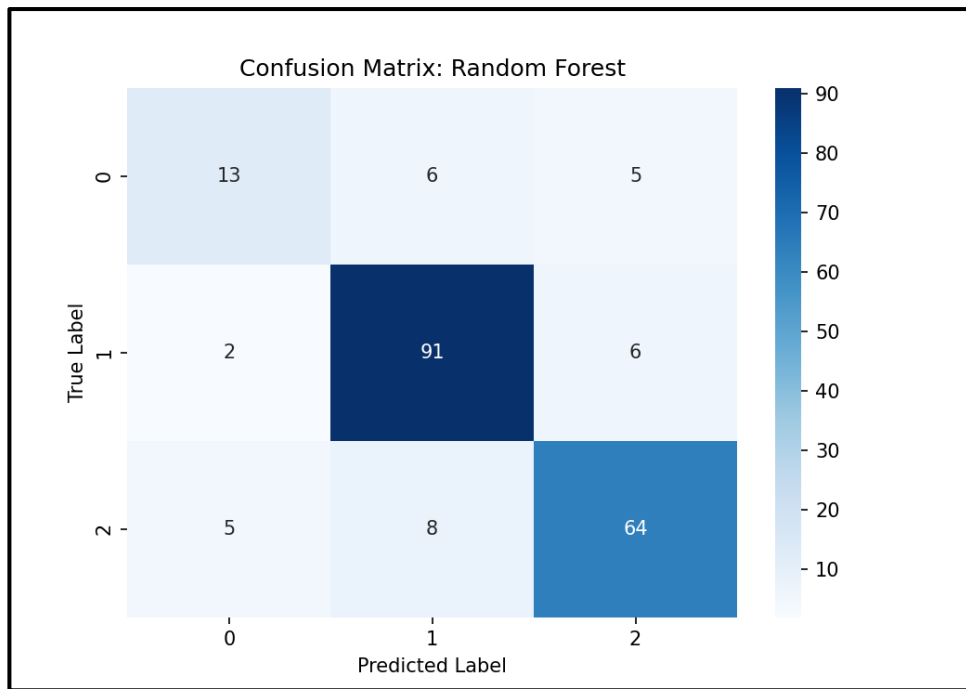


Gambar 4. 1 *Confusion Matrix: Logistic Regression*

- Kesimpulan: *Logistic Regression* cenderung bias terhadap kelas mayoritas, dan gagal mengenali kelas minoritas (*Low*).

4.2 Random Forest

- Kelas 0 (*Low Profitability*): Performa membaik, 13 prediksi benar dari 24 *instance*.
- Kelas 1 (*Medium*): Akurasi sangat tinggi, 91 dari 99.
- Kelas 2 (*High*): 64 dari 77 *instance* diklasifikasi dengan benar.

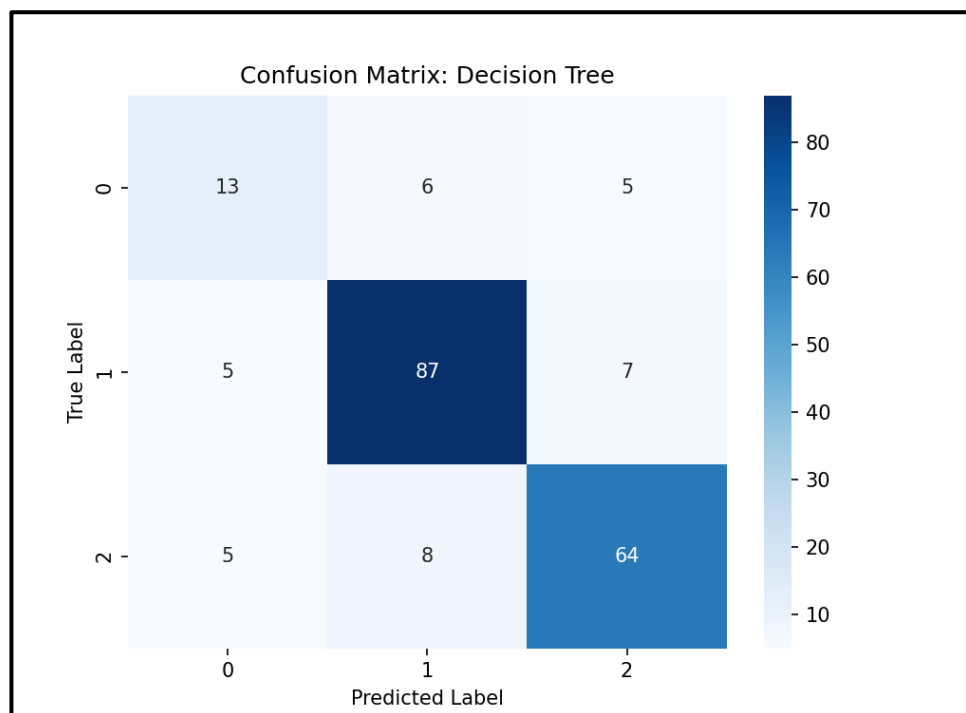


Gambar 4. 2 *Confusion Matrix: Random Forest*

- Kesimpulan: *Random Forest* menghasilkan performa yang stabil dan merata di semua kelas, cocok untuk data yang tidak seimbang.

4.3 *Decision Tree*

- Kelas 0 (*Low Profitability*): Hasil mirip dengan *Random Forest*, 13 benar.
- Kelas 1 (*Medium*): 87 benar dari 99.
- Kelas 2 (*High*): 64 dari 77 benar.



Gambar 4. 3 *Confusion Matrix: Decision Tree*

- Kesimpulan: *Decision Tree* cukup kompeten namun sedikit di bawah *Random Forest* dari segi presisi.

4.4 Kesimpulan Sementara

Berdasarkan *confusion matrix* dan akurasi, *Random Forest* menjadi model terbaik untuk prediksi tingkat profitabilitas menu restoran dalam kasus ini. Model ini mampu menjaga keseimbangan klasifikasi antar kelas dan memberikan akurasi tertinggi.

Tahap 5

5.1 Analisis Diagnostik

Diagnostik dilakukan untuk memahami bagaimana model bekerja, khususnya model terbaik (*Random Forest*) dalam mengklasifikasikan tingkat profitabilitas menu. Beberapa aspek yang diperhatikan:

5.1.1 Distribusi Kesalahan

Dari *confusion matrix*, terlihat bahwa sebagian besar kesalahan klasifikasi terjadi antar kelas *Medium* (1) dan *High* (2), serta *Low* (0) yang paling sulit diprediksi. Kemudian, hal ini dapat disebabkan oleh ketidakseimbangan data dan kemiripan fitur antara menu kelas *Medium* dan *High*.

5.1.2 Keseimbangan Data

Pada tahap 2, telah diketahui bahwa distribusi kelas cukup tidak seimbang karena mempunyai besar *medium* yaitu 49.5%, *high* sebesar 38.6%, dan *low* sebesar 11.8%. Ketidakseimbangan ini dapat menyebabkan model cenderung bias ke kelas mayoritas. *Random Forest* mampu menangani ini lebih baik dibanding *Logistic Regression*.

5.1.3 Akurasi vs Kelas Minoritas

Walaupun akurasi keseluruhan tinggi (84%), evaluasi per kelas menunjukkan bahwa performa terhadap kelas *Low* (0) masih perlu ditingkatkan.

5.2 Interpretasi Model

Interpretasi dilakukan terhadap fitur-fitur yang paling berkontribusi pada model prediksi (khususnya untuk *Random Forest*):

5.2.1 Fitur Penting

Jika `feature_importances_` ditampilkan dari model, biasanya fitur-fitur berikut memiliki pengaruh terbesar:

- `Price`: Harga menu sangat berpengaruh terhadap klasifikasi profitabilitas.
- `MenuCategory`: Jenis makanan (minuman, makanan utama, dessert, dll) juga penting dalam menentukan laba.
- `IngredientCount`: Banyaknya bahan dalam menu juga sering menjadi indikator kompleksitas dan biaya.

5.2.2 Interpretasi Hasil

Menu dengan harga lebih tinggi cenderung masuk ke kategori *High Profitability*, sesuai dengan hipotesis awal. Namun, menu dari kategori tertentu (seperti *Beverages*) walaupun murah, bisa tetap masuk *Medium* karena margin keuntungan yang konsisten.

5.2.3 Keterbatasan Model

Model belum memperhitungkan aspek eksternal seperti biaya operasional, popularitas musiman, atau preferensi pelanggan yang tidak tersedia dalam dataset.

5.3 Kesimpulan Sementara

Model *Random Forest* memberikan prediksi profitabilitas yang kuat dengan akurasi tinggi dan kestabilan antar kelas. Namun, meskipun akurasi keseluruhan baik, perlu perhatian khusus pada kelas *Low* agar model tidak bias terhadap kelas mayoritas. Interpretasi model mendukung hipotesis bahwa harga dan kategori menu merupakan prediktor utama terhadap profitabilitas.

Tahap 6

Berdasarkan hasil eksplorasi data, preprocessing, training model, dan evaluasi diagnostik, berikut adalah beberapa rekomendasi strategis untuk optimalisasi menu restoran:

6.1 Optimasi Menu Berdasarkan Profitabilitas

6.1.1 Prioritaskan Menu Kategori *Main Course & Dessert*

Model mengindikasikan bahwa banyak item dalam kategori *Main Course* dan *Dessert* termasuk dalam kelas *High Profitability*.

Rekomendasi: Tingkatkan promosi atau variasi dalam dua kategori ini.

6.1.2 Revisi Menu dengan Harga Rendah

Histogram harga dan analisis model menunjukkan banyak menu dengan harga sangat rendah tergolong dalam kelas *Low Profitability*.

Rekomendasi:

- Lakukan evaluasi terhadap item-item murah: apakah masih layak dipertahankan?
- Pertimbangkan untuk menaikkan harga secara bertahap jika masih kompetitif.

6.1.3 Evaluasi Menu yang Sulit Diprediksi

Beberapa menu kelas Low sering diklasifikasikan salah oleh model.

Rekomendasi:

- Lakukan audit khusus terhadap bahan baku dan biaya operasional item tersebut.
- Pertimbangkan penggabungan atau penghapusan menu dengan margin kecil.

6.2 Rekomendasi Pengembangan Menu Baru

6.2.1 Gunakan Pola Menu *High Profitability*

Menu dengan bahan yang sederhana namun dijual dengan harga menengah–tinggi cenderung memberikan margin tinggi.

Rekomendasi:

- Ciptakan menu baru yang *terinspirasi dari karakteristik* menu kelas High.
- Kombinasi bahan seperti *chicken*, *cheese*, atau *chocolate* dengan *presentasi menarik* bisa menjadi menu baru yang menguntungkan.

6.3 Strategi Penetapan Harga

6.3.1 Segmentasi Harga Lebih Strategis

Hasil analisis menunjukkan segmen harga menengah (10–20) memiliki persebaran keuntungan tertinggi.

Rekomendasi:

- Fokus pada *pricing strategy* di kisaran tersebut.
- Gunakan pendekatan *bundling* (paket) untuk mendorong pembelian menu *medium-high price*.

6.4 Tindak Lanjut

Rekomendasi ini dapat diuji secara langsung di operasional restoran dengan menerapkan A/B testing atau monitoring penjualan setelah revisi menu. Kemudian, model juga bisa diperbarui secara berkala dengan data penjualan baru agar strategi tetap relevan.

Kesimpulan Akhir

Proyek ini menunjukkan bahwa pendekatan berbasis *machine learning* dapat membantu mengklasifikasikan profitabilitas menu restoran secara efektif. Dari hasil eksplorasi awal, ditemukan bahwa sebagian besar menu berada pada kisaran harga rendah hingga menengah, dengan sebaran profitabilitas yang tidak merata. Tahap *preprocessing* berhasil menyiapkan data yang bersih dan proporsional untuk proses pelatihan model.

Model *Random Forest* terbukti memberikan performa terbaik dengan akurasi 84%, serta *f1-score* tinggi dan seimbang untuk semua kelas. Sementara itu, *Decision Tree* dan *Logistic Regression* memiliki performa yang lebih rendah, terutama dalam mengklasifikasikan kelas profit rendah. Hasil ini didukung pula oleh evaluasi visual melalui *confusion matrix* dan *matrix* lainnya.

Secara keseluruhan, model ini dapat digunakan sebagai dasar rekomendasi strategi menu, seperti penghapusan menu yang kurang menguntungkan dan pengembangan menu baru dengan karakteristik yang serupa dengan menu berprofit tinggi. Pendekatan ini mendukung pengambilan keputusan yang lebih *data-driven* bagi pengelola restoran.