



GLOBAL FOOD WASTE DASHBOARD: ANALISIS & PREDIKSI LIMBAH MAKANAN PER KAPITA

Study Group EDM Kelompok 22



OUR TEAM



Nazwa Yunia
Santosa



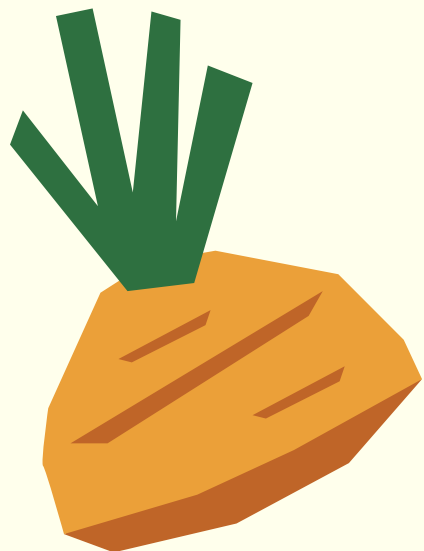
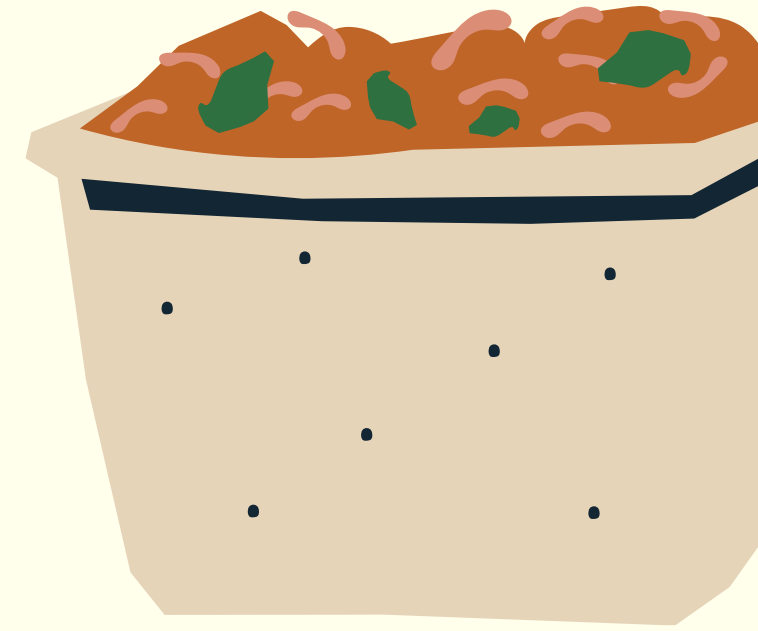
Prisa Ayudia
Ningtiyas

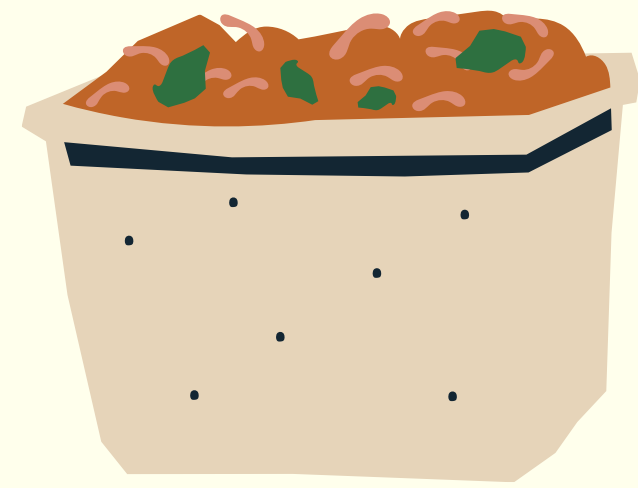


Sabrina Adinda
Willyasarie



PROBLEM STATEMENT





Dalam beberapa dekade terakhir, limbah makanan telah menjadi salah satu isu lingkungan dan sosial paling mendesak di dunia. Menurut data yang tersedia, negara-negara di seluruh dunia menghasilkan limbah makanan dalam jumlah besar, baik di tingkat rumah tangga, ritel, maupun layanan makanan. Meskipun data global telah dikumpulkan, pemahaman mendalam mengenai pola, perbedaan antar negara dan sektor, serta faktor-faktor yang mempengaruhinya masih terbatas.





OVERVIEW DATASET






Sumber Dataset

dataset diperoleh dari kaggle dengan link <https://www.kaggle.com/code/joebeachcapital/food-waste-data-import-eda-starter/input>

Deskripsi dataset

- Country: Nama negara.,
- combined figures (kg/capita/year): Total limbah makanan per orang per tahun.,
- Household estimate (kg/capita/year): Limbah rumah tangga per orang per tahun.,
- Household estimate (tonnes/year): Total limbah rumah tangga per tahun.,
- Retail estimate (kg/capita/year): Limbah sektor retail per orang per tahun.,
- Retail estimate (tonnes/year): Total limbah retail per tahun.,
- Food service estimate (kg/capita/year): Limbah sektor jasa makanan per orang per tahun.,
- Food service estimate (tonnes/year): Total limbah jasa makanan per tahun.,
- Confidence in estimate: Tingkat kepercayaan data.,
- M49 code: Kode negara dari PBB.,
- Region: Wilayah geografis negara.,
- Source: Sumber data.



PEMBUATAN MODEL

IMPORT LIBRARY

No.3

```
✓import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import roc_curve, auc, roc_auc_score
```

Pada bagian ini, berbagai library penting diimpor. Library seperti pandas dan numpy digunakan untuk manipulasi dan analisis data. matplotlib dan seaborn digunakan untuk membuat visualisasi seperti boxplot dan heatmap. Dari scikit-learn, beberapa modul digunakan untuk preprocessing (seperti encoding dan scaling), pembagian data, pemodelan (Random Forest, Naive Bayes, Logistic Regression), dan evaluasi model (ROC, AUC).

LOAD DATASET

```
df = pd.read_csv("Food Waste data and research - by country.csv")
```

Dataset dimuat dari file CSV yang berisi data mengenai food waste berdasarkan negara. Ini adalah langkah awal untuk mempersiapkan data sebelum dianalisis lebih lanjut.

No.3

DATA UNDERSTANDING

```
print(df.info())
```

Menampilkan struktur data, termasuk jumlah baris, kolom, jenis data setiap kolom, serta jumlah nilai yang tidak null. Ini penting untuk memahami struktur dasar dataset.

```
df.duplicated().sum()
```

Digunakan untuk menghitung jumlah baris yang duplikat. Baris duplikat dapat menyebabkan distorsi dalam analisis dan pemodelan, sehingga perlu dihapus jika ada.

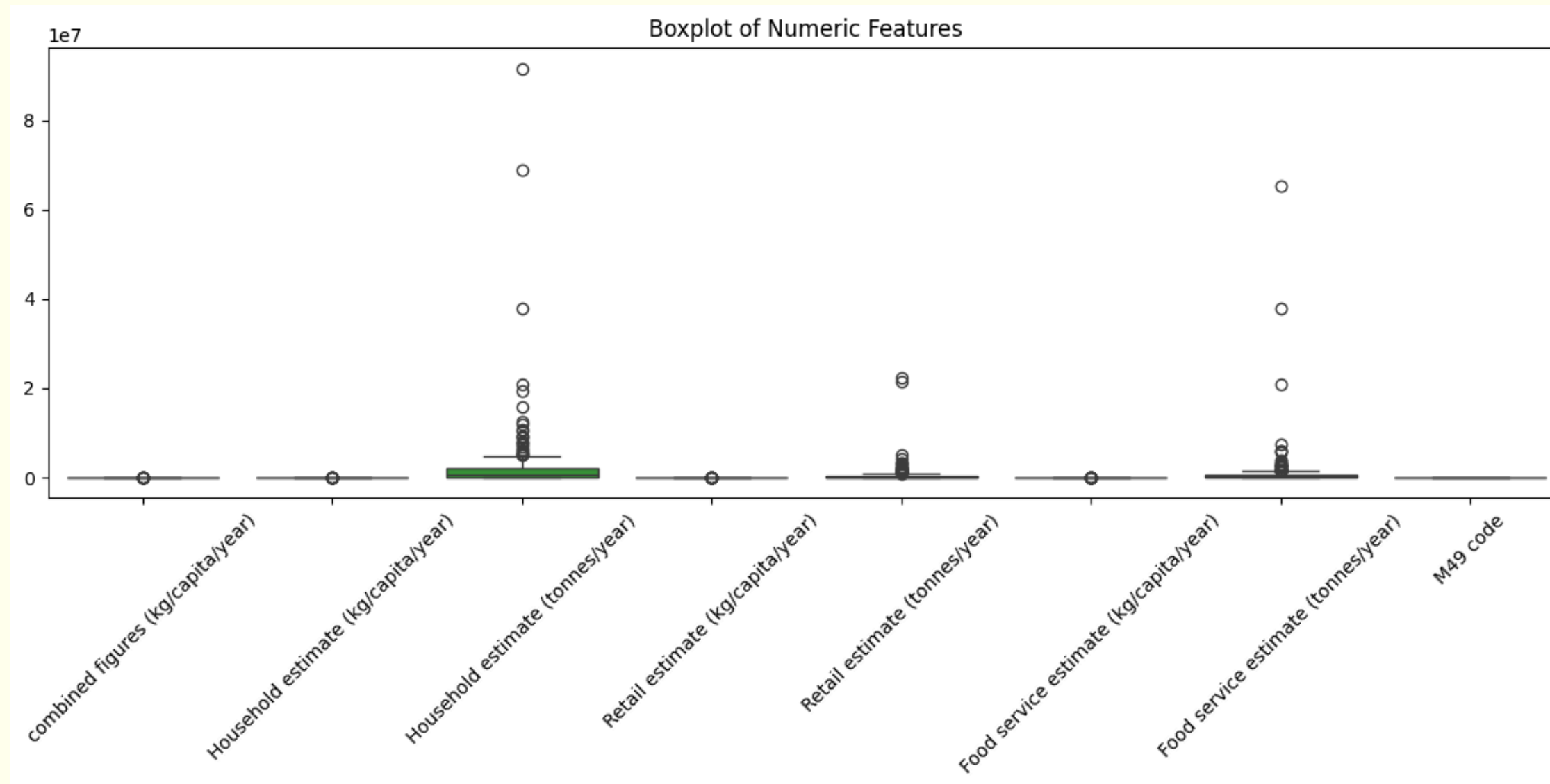
```
df.isnull().sum()
```

Digunakan untuk menghitung jumlah nilai kosong (null) di setiap kolom. Tahap ini penting untuk menentukan apakah perlu dilakukan penanganan data hilang seperti imputasi atau penghapusan.

```
df.describe()
```

Menampilkan ringkasan statistik deskriptif dari kolom numerik seperti rata-rata, standar deviasi, nilai maksimum, minimum, dll. Ini membantu memahami distribusi nilai.

VISUALISASI DATA



Boxplot digunakan untuk mendeteksi outlier pada fitur numerik. Dengan visualisasi ini, dapat dilihat nilai-nilai ekstrem di masing-masing fitur yang mungkin perlu penanganan lebih lanjut.

ENCODING DATA

```
df_encoded = df.copy()
le_conf = LabelEncoder()
le_region = LabelEncoder()
df_encoded["Confidence in estimate"] = le_conf.fit_transform(df["Confidence in estimate"])
df_encoded["Region"] = le_region.fit_transform(df["Region"])
```

Label encoding dilakukan pada kolom kategorikal: Confidence in estimate dan Region. Ini penting karena model ML hanya dapat menerima data numerik.

No.3

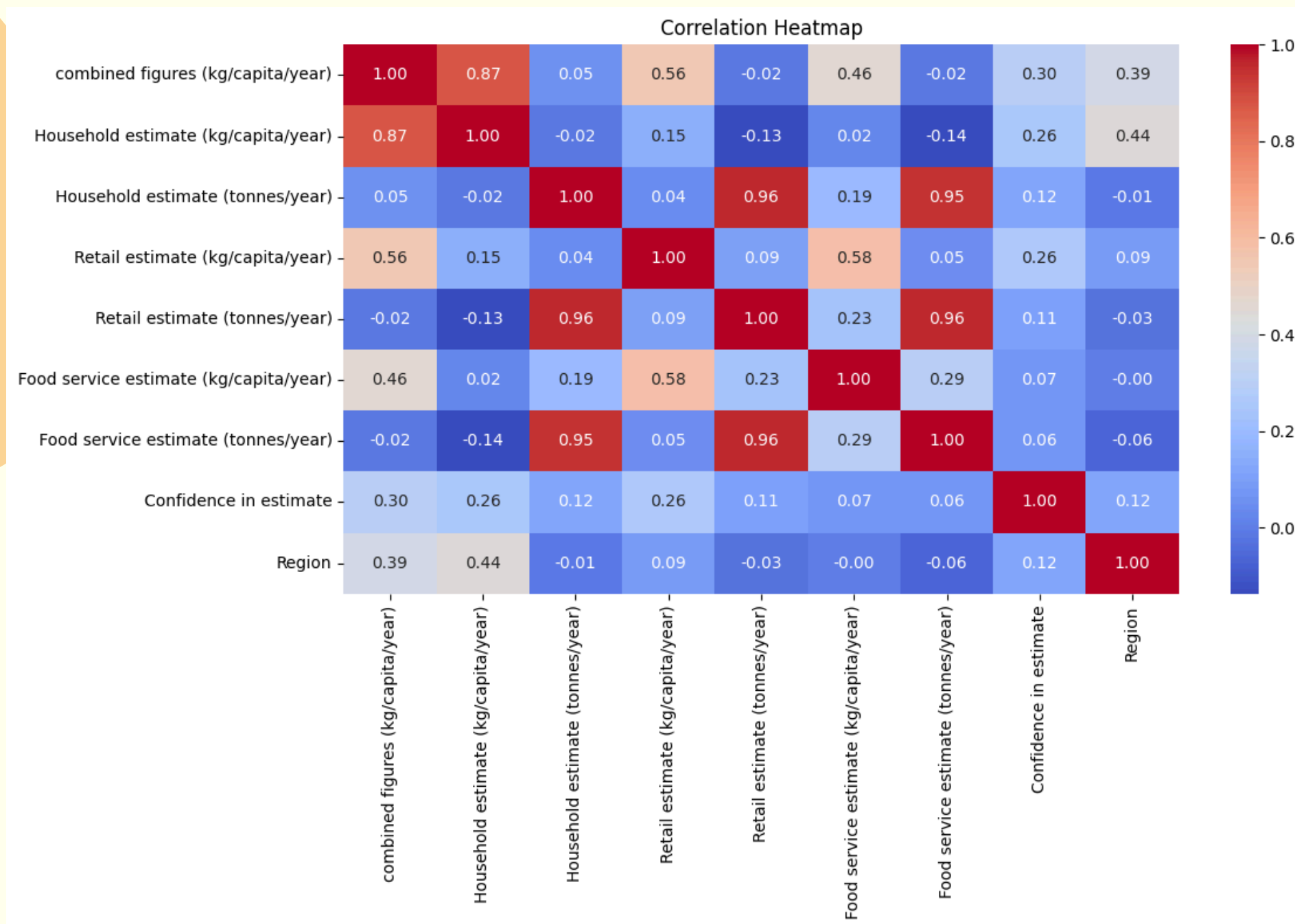
DROP KOLOM

```
df_encoded = df_encoded.drop(columns=["Country", "Source", "M49 code"])
```

Kolom-kolom yang tidak diperlukan dalam pelatihan model dihapus karena tidak memberikan kontribusi signifikan, atau memiliki informasi yang tidak dapat diproses oleh model.

No.3

KORELASI HEATMAP



Heatmap digunakan untuk melihat korelasi antar fitur. Ini dapat membantu dalam feature selection, yaitu memilih fitur yang relevan dan tidak saling berkorelasi tinggi.

No.3

VISUALISASI ROC CURVE

Fungsi ini digunakan untuk menggambarkan kurva ROC untuk masing-masing kelas pada prediksi multi-kelas. ROC digunakan untuk mengevaluasi kinerja model klasifikasi dalam hal trade-off antara True Positive Rate dan False Positive Rate.

No.3

SPLIT DATA MODEL

```
# ===== 4. Pisahkan Fitur dan Target =====
X = df_encoded.drop(columns=["Confidence in estimate"])
y = df_encoded["Confidence in estimate"]

# ===== 5. Split Data =====
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)

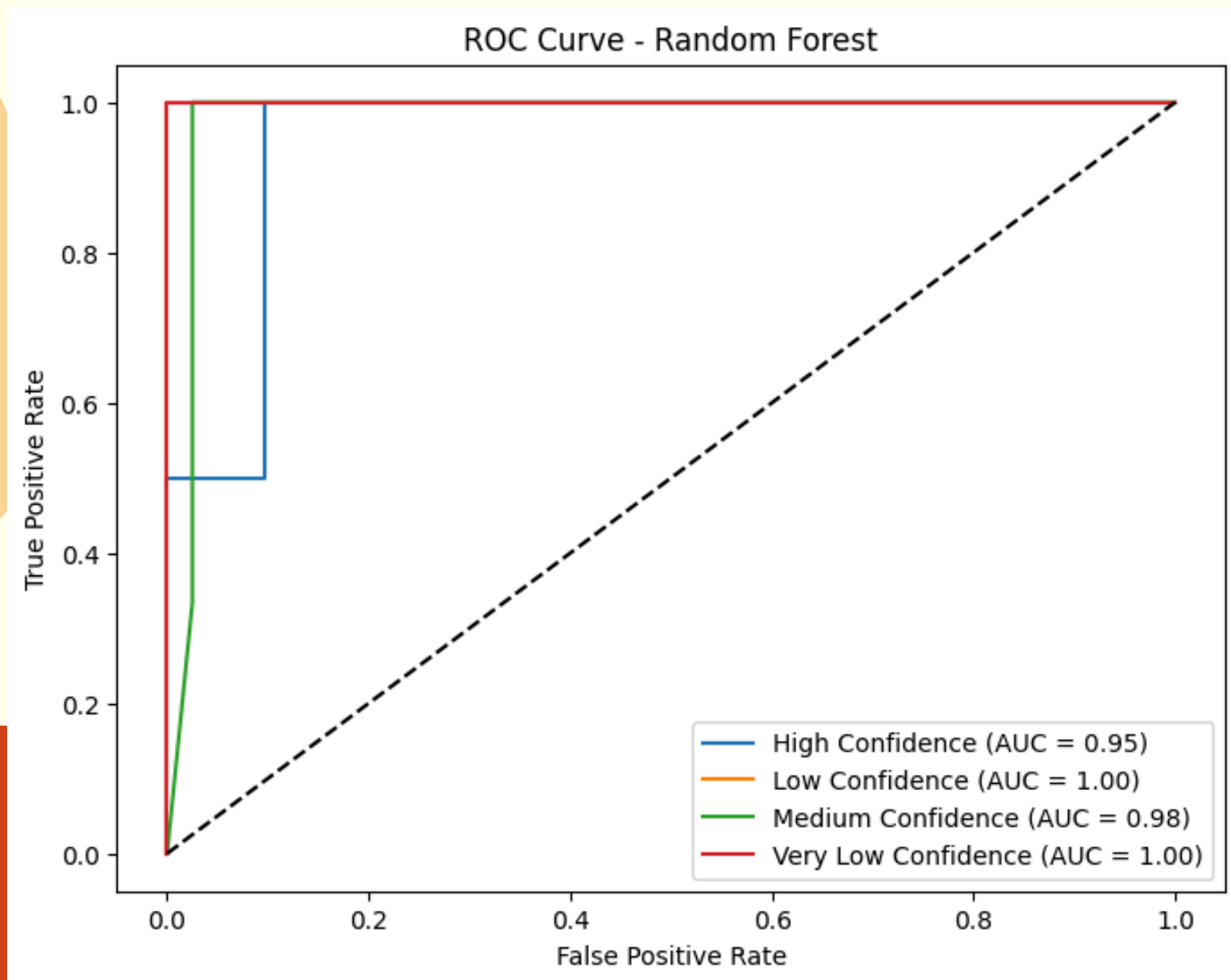
# ===== 6. Scaling =====
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# ===== 7. Random Forest =====
rf = RandomForestClassifier(random_state=42)
rf.fit(X_train, y_train)
rf_preds = rf.predict(X_test)

# ===== 8. Naive Bayes =====
nb = GaussianNB()
nb.fit(X_train_scaled, y_train)
nb_preds = nb.predict(X_test_scaled)

# ===== 9. Logistic Regression =====
lr = LogisticRegression(max_iter=1000)
lr.fit(X_train_scaled, y_train)
lr_preds = lr.predict(X_test_scaled)
```

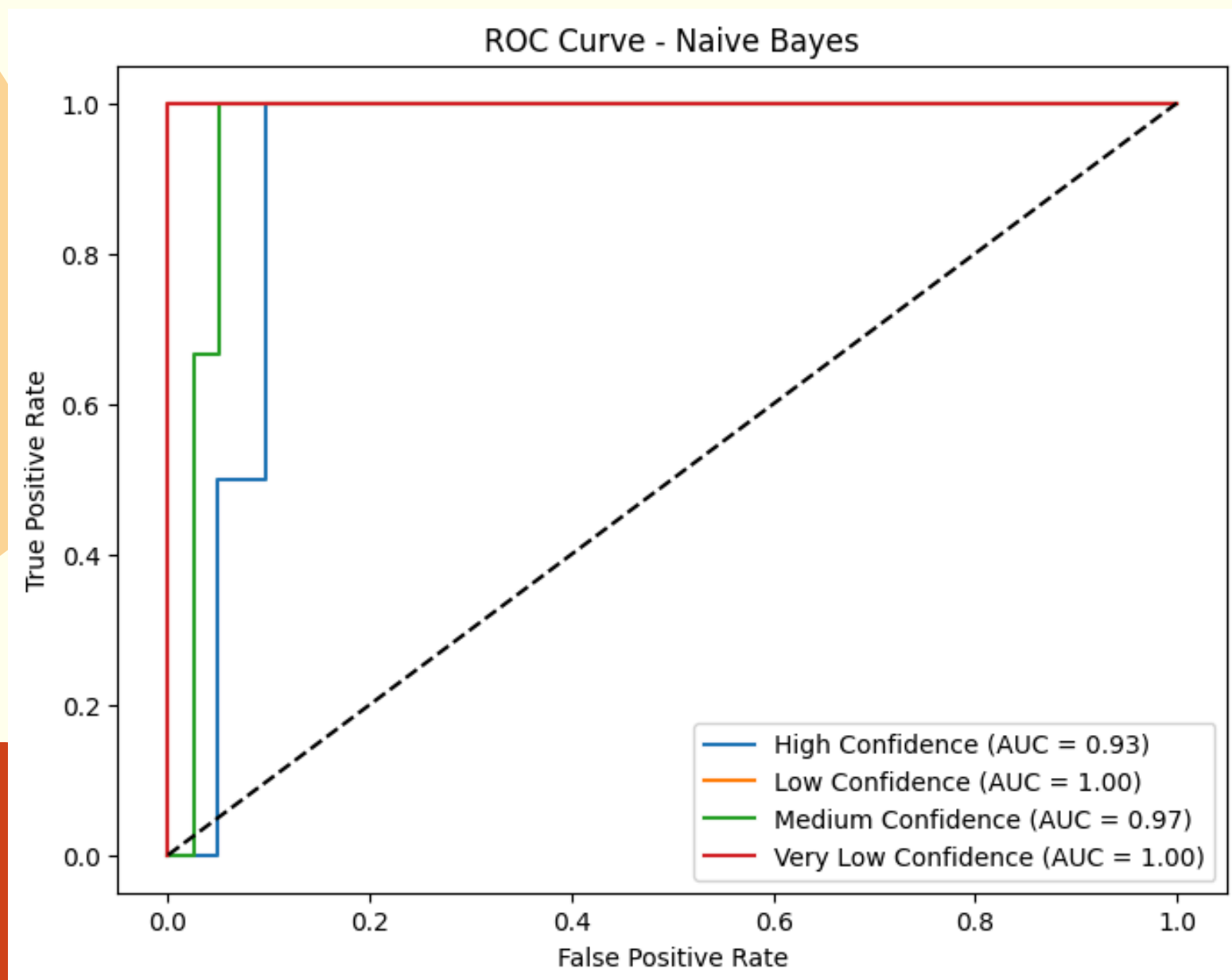

RANDOM FOREST



```
=== Classification Report: Random Forest ===
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	1.00	1.00	12
2	0.75	1.00	0.86	3
3	0.96	1.00	0.98	26
accuracy			0.95	43
macro avg	0.68	0.75	0.71	43
weighted avg	0.91	0.95	0.93	43

NAIVE BAYES

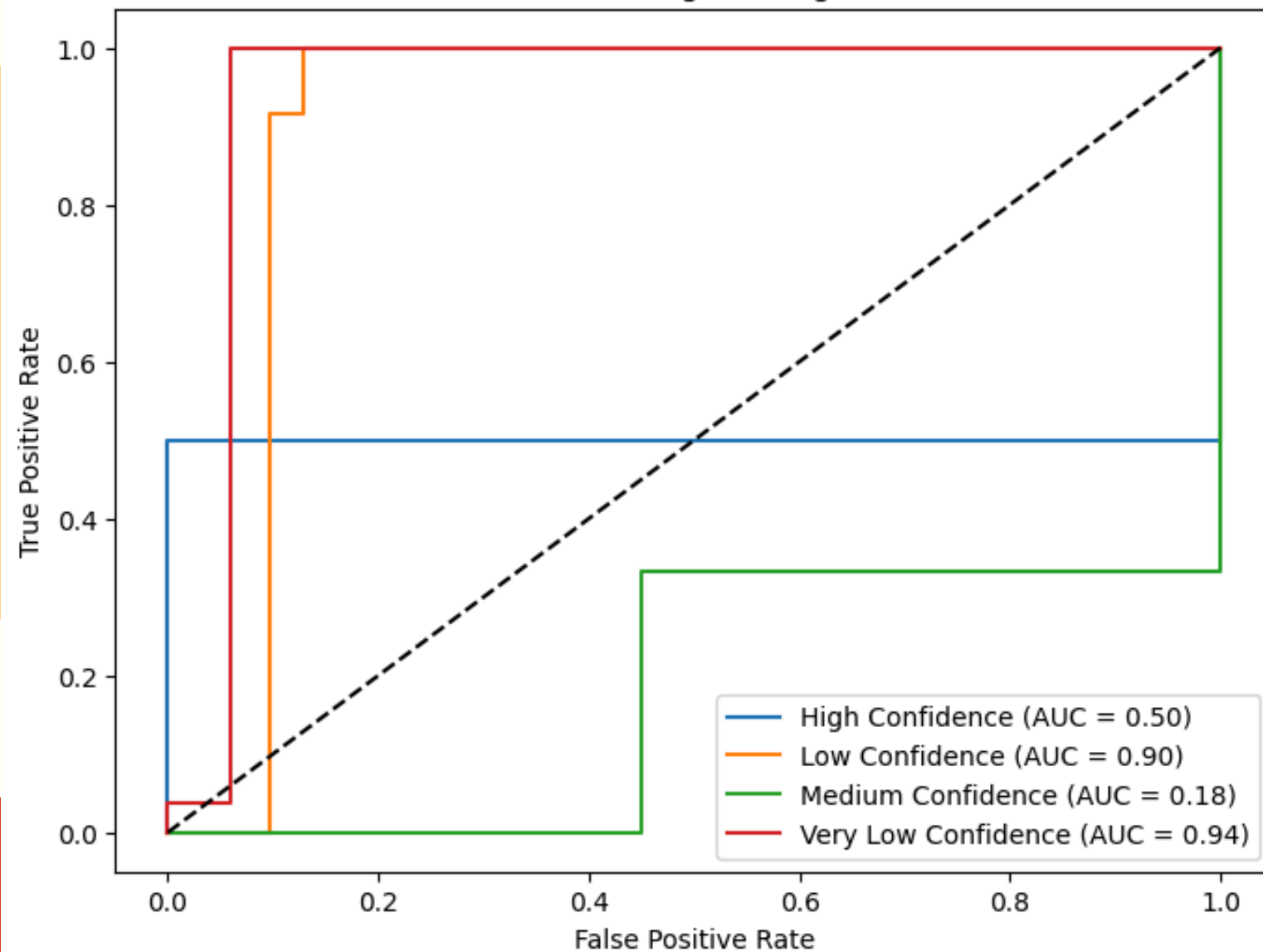


=== Classification Report: Naive Bayes ===

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	1.00	1.00	12
2	0.50	0.67	0.57	3
3	1.00	0.96	0.98	26
accuracy			0.91	43
macro avg	0.62	0.66	0.64	43
weighted avg	0.92	0.91	0.91	43

LOGISTIC REGRESSION

ROC Curve - Logistic Regression



```
=== Classification Report: Logistic Regression ===
```

	precision	recall	f1-score	support
0	1.00	0.50	0.67	2
1	0.79	0.92	0.85	12
2	0.00	0.00	0.00	3
3	0.93	1.00	0.96	26
accuracy			0.88	43
macro avg	0.68	0.60	0.62	43
weighted avg	0.83	0.88	0.85	43

HASIL PREDIKSI CONFIDENCE

Prediksi Confidence in Estimate

Masukkan estimasi limbah makanan dari berbagai sektor:

Household (kg/capita)

200.00

- +

Retail (kg/capita)

200.00

- +

Food Service (kg/capita)

50.00

- +

Prediksi Confidence

(40.00%) → Very Low Confidence

HASIL PREDIKSI KATEGORI LIMBAH MAKANAN



Prediksi Kategori Limbah Makanan (High/Low)

GDP per capita

100.00

- +

Populasi

10.00

- +

Prediksi High/Low Food Waste



Prediksi: Low Food Waste

HASIL ANALISIS FOOD WASTE PERBANDINGAN ANTAR NEGARA



Analisis Perbandingan Antar Negara

Pilih Negara:

Andorra ×

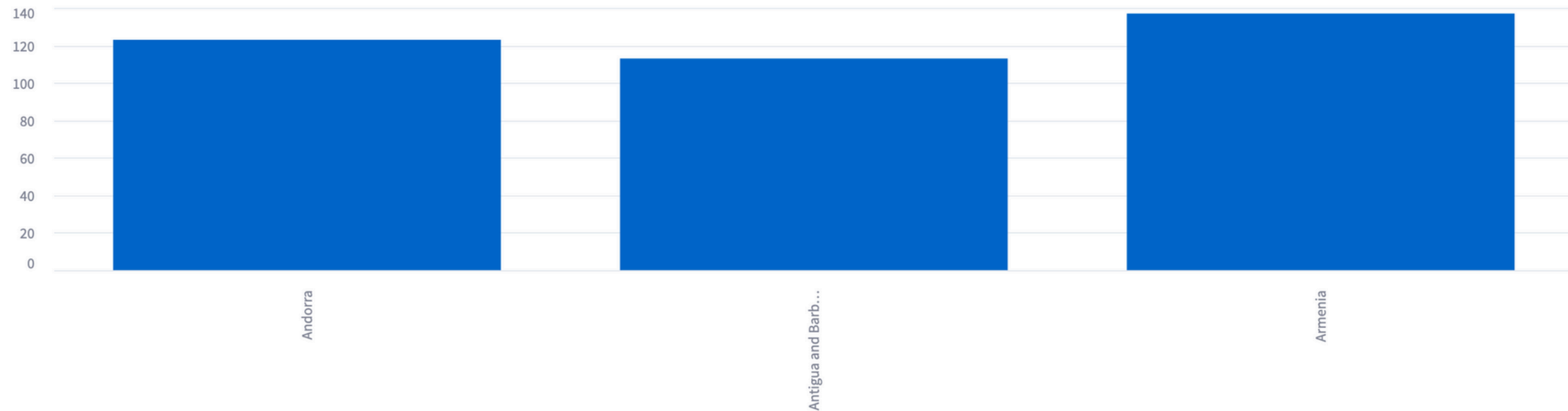
Antigua and Bar... ×

Armenia ×



Pilih Metrik:

combined figures (kg/capita/year)|



THANK
YOU

