1. Dataset Selection

Overview

The dataset titled Most-dangerous-countries-for-women-2024, this dataset analyzes women’s safety across 50 countries using composite indices and specific risks indicators from 2019-2023, revealing stark disparities between high- risk nations like South Africa and safer countries like Norway. While valuable for comparative research, the mixed year data and varying metrics require cautious interpretation when evaluating global trends in gender-based violence and discrimination.

Dataset Composition:

1. Primary Index (2023): Women Peace and Security Index (WPSI) scores (0-1 scale)
2. Supporting Index (2019): Women's Danger Index (WDI) with:
3. Total composite score (numeric scale)
4. 8 standardized sub-indicators (0-100 scales): Street Safety, Intentional Homicide, Non-Partner Violence, Intimate Partner Violence, Legal Discrimination, Global Gender Gap, Gender Inequality, and Societal Attitudes Toward Violence

2. **Structural Format**:

   - Country-level records: 50 sovereign nations
   - Temporal mix: WPSI (current 2023 data) paired with WDI (baseline 2019 data)
   - Score types**:** Contains both:
   - Normalized scores (WPSI 0-1 range)
   - Raw component metrics (WDI 0-100 scales)
   - Calculated aggregates (WDI Total Score)

3. **Data Integrity Notes**:

- Complete cases for all primary countries
- Empty rows represent spreadsheet artifacts
- No explicit missing value markers in core data columns
- Contains both absolute values (homicide rates) and perception-based metrics (street safety)

Feature	Description	Type
Primary Metric	Women Peace and Security Index (WPSI) score (0-1 scale)	Numeric
Composite Metric	Women's Danger Index (WDI) total score (higher = more dangerous)	Numeric
Street Safety	Perceived safety in public spaces (0-100, higher = more dangerous)	Numeric
Intentional Homicide	Rate of gender-based homicides (0-100)	Numeric
Non-Partner Violence	Prevalence of violence by non-intimate perpetrators (0-100)	Numeric
Intimate Partner Violence	Prevalence of domestic violence (0-100)	Numeric
Legal Discrimination	Systemic gender-based legal inequalities (0-100, higher = more discriminatory)	Numeric
Global Gender Gap	Economic, educational, and political disparities (0-100)	Numeric
Gender Inequality	Composite of health, empowerment, and labor disparities (0-100)	Numeric
Attitudes Toward Violence	Societal acceptance of gender-based violence (0-100)	Numeric
Data Limitations	Mixed-year metrics, potential cultural reporting biases, unclear weightings	Numeric

Reason for Dataset Selection

This dataset was selected for its comprehensive evaluation of women's safety across multiple dimensions, making it valuable for:

1. Comparative Analysis – Enables cross-country benchmarking on gender-based risks (e.g., violence, discrimination, legal inequality).
2. Policy & Advocacy – Helps identify high-risk regions needing intervention (e.g., South Africa's extreme homicide rates, Saudi Arabia's legal discrimination).
3. Trend Monitoring – Despite mixed-year data (2019–2023), it provides a baseline for tracking progress in women's safety over time.
4. Research Utility – Combines quantitative indices (WPSI, WDI) with granular sub-metrics, supporting studies on gender violence, societal attitudes, and systemic inequality.

Limitations:

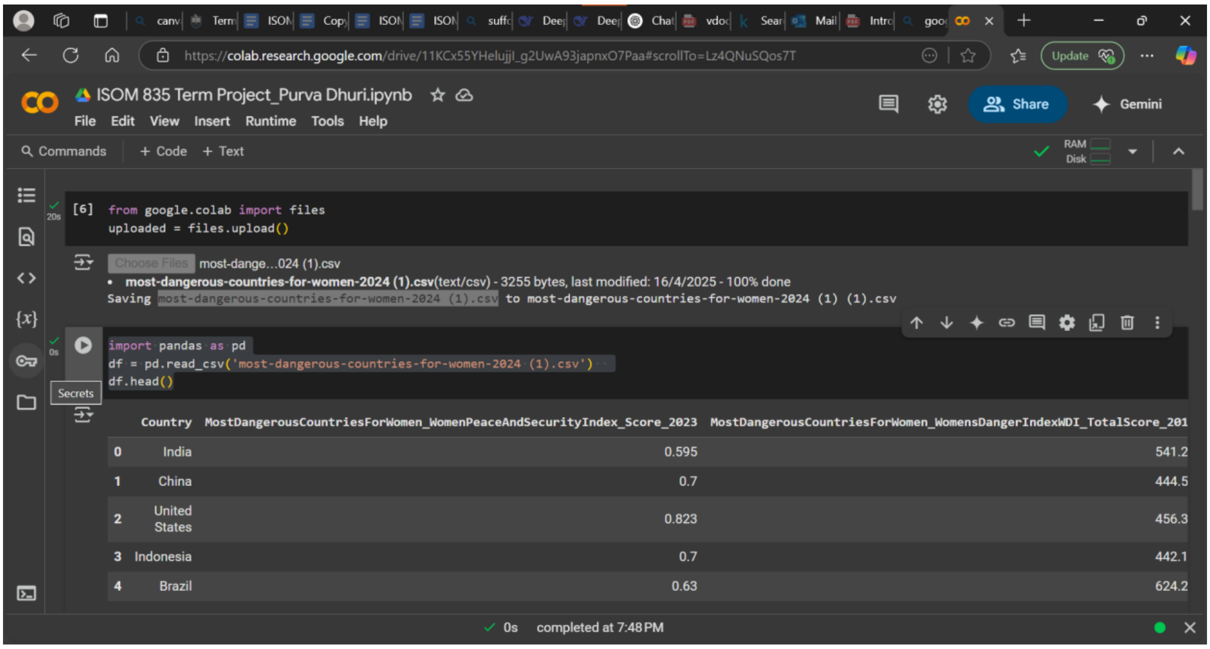
- Temporal inconsistency (2019 vs. 2023 data may not reflect recent changes).
- Perception bias in subjective metrics (e.g., street safety).
- Lack of weighting details for composite scores.

Practical Application

- Policy Development
- Travel Advisories Corporate ESG Strategies
- Academic Research
- Awareness Campaigns
- Risk Assessment

1. Dataset Loading Code

```
from google.colab import files
uploaded = files.upload()

import pandas as pd
df = pd.read_csv('most-dangerous-countries-for-women-
2024 (1).csv')
df.head()

Output:
```



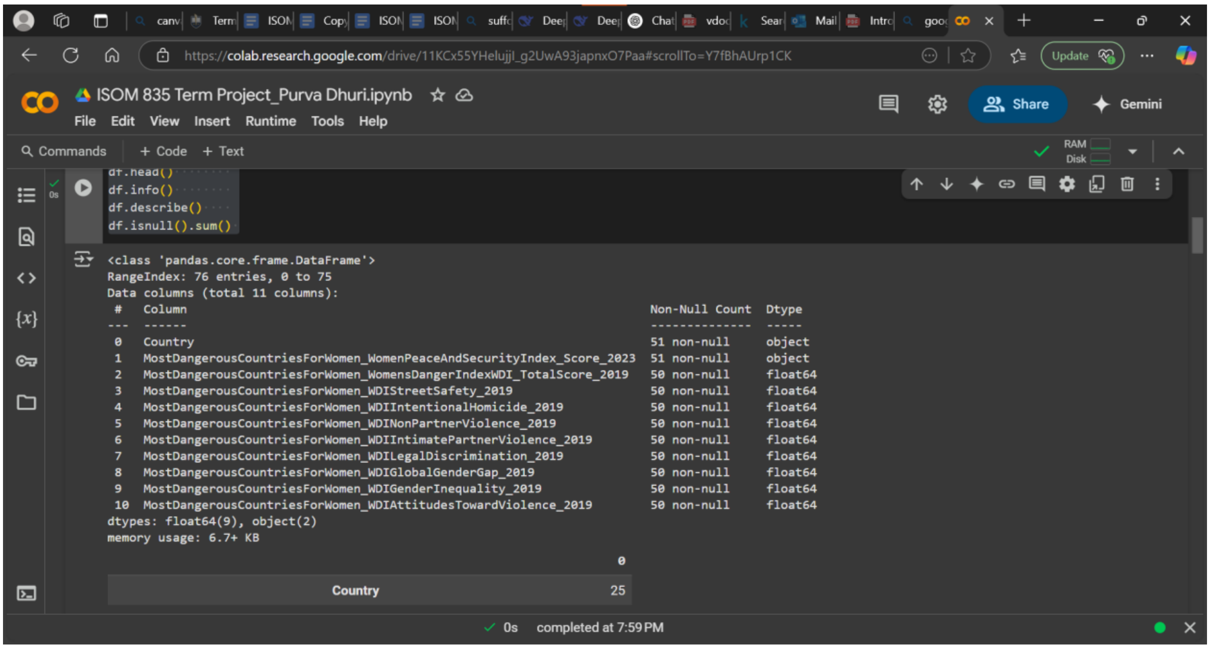
ISOM 835 Term Project_Purva Dhuri.ipynb
File Edit View Insert Runtime Tools Help
Share
Gemini
Commands
+ Code + Text
RAM
Disk
Secrets

```
[6] from google.colab import files
    uploaded = files.upload()

Choose Files most-dange...024 (1).csv
• most-dangerous-countries-for-women-2024 (1).csv(text/csv) - 3255 bytes, last modified: 16/4/2025 - 100% done
Saving most-dangerous-countries-for-women-2024 (1).csv to most-dangerous-countries-for-women-2024 (1) (1).csv

import pandas as pd
df = pd.read_csv('most-dangerous-countries-for-women-2024 (1).csv')
df.head()
```

	Country	MostDangerousCountriesForWomen_WomenPeaceAndSecurityIndex_Score_2023	MostDangerousCountriesForWomen_WomensDangerIndexWDI_TotalScore_201
0	India	0.595	541.2
1	China	0.7	444.5
2	United States	0.823	456.3
3	Indonesia	0.7	442.1
4	Brazil	0.63	624.2

✓ 0s completed at 7:48 PM

```
import pandas as pd

# Load the CSV file
df = pd.read_csv('most-dangerous-countries-for-women-
2024 (1).csv')
df.head()
df.info()
df.describe()
df.isnull().sum()

Output:
```

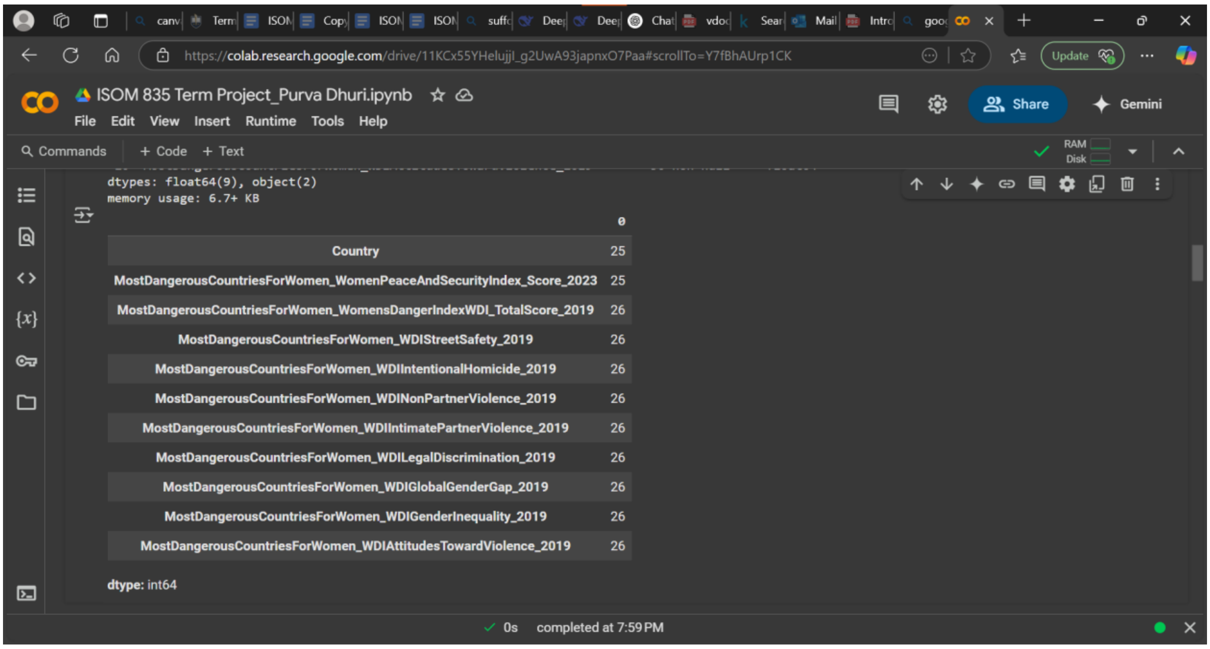

ISOM 835 Term Project_Purva Dhuri.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text
Share
Gemini

```
df.head()
df.info()
df.describe()
df.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 76 entries, 0 to 75
Data columns (total 11 columns):
 #   Column                                                          Non-Null Count  Dtype
---  ------                                                          --------------  -----
 0   Country                                                         51 non-null     object
 1   MostDangerousCountriesForWomen_WomenPeaceAndSecurityIndex_Score_2023  51 non-null     object
 2   MostDangerousCountriesForWomen_WomensDangerIndexWDI_TotalScore_2019  50 non-null     float64
 3   MostDangerousCountriesForWomen_WDIStreetSafety_2019             50 non-null     float64
 4   MostDangerousCountriesForWomen_WDIIntentionalHomicide_2019      50 non-null     float64
 5   MostDangerousCountriesForWomen_WDINonPartnerViolence_2019       50 non-null     float64
 6   MostDangerousCountriesForWomen_WDIIntimatePartnerViolence_2019  50 non-null     float64
 7   MostDangerousCountriesForWomen_WDILegalDiscrimination_2019      50 non-null     float64
 8   MostDangerousCountriesForWomen_WDIGlobalGenderGap_2019          50 non-null     float64
 9   MostDangerousCountriesForWomen_WDIGenderInequality_2019         50 non-null     float64
 10  MostDangerousCountriesForWomen_WDIAttitudesTowardViolence_2019  50 non-null     float64
dtypes: float64(9), object(2)
memory usage: 6.7+ KB
```

	0
Country	25

✓ 0s completed at 7:59 PM


ISOM 835 Term Project_Purva Dhuri.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text
Share
Gemini

```
dtypes: float64(9), object(2)
memory usage: 6.7+ KB
```

	0
Country	25
MostDangerousCountriesForWomen_WomenPeaceAndSecurityIndex_Score_2023	25
MostDangerousCountriesForWomen_WomensDangerIndexWDI_TotalScore_2019	26
MostDangerousCountriesForWomen_WDIStreetSafety_2019	26
MostDangerousCountriesForWomen_WDIIntentionalHomicide_2019	26
MostDangerousCountriesForWomen_WDINonPartnerViolence_2019	26
MostDangerousCountriesForWomen_WDIIntimatePartnerViolence_2019	26
MostDangerousCountriesForWomen_WDILegalDiscrimination_2019	26
MostDangerousCountriesForWomen_WDIGlobalGenderGap_2019	26
MostDangerousCountriesForWomen_WDIGenderInequality_2019	26
MostDangerousCountriesForWomen_WDIAttitudesTowardViolence_2019	26

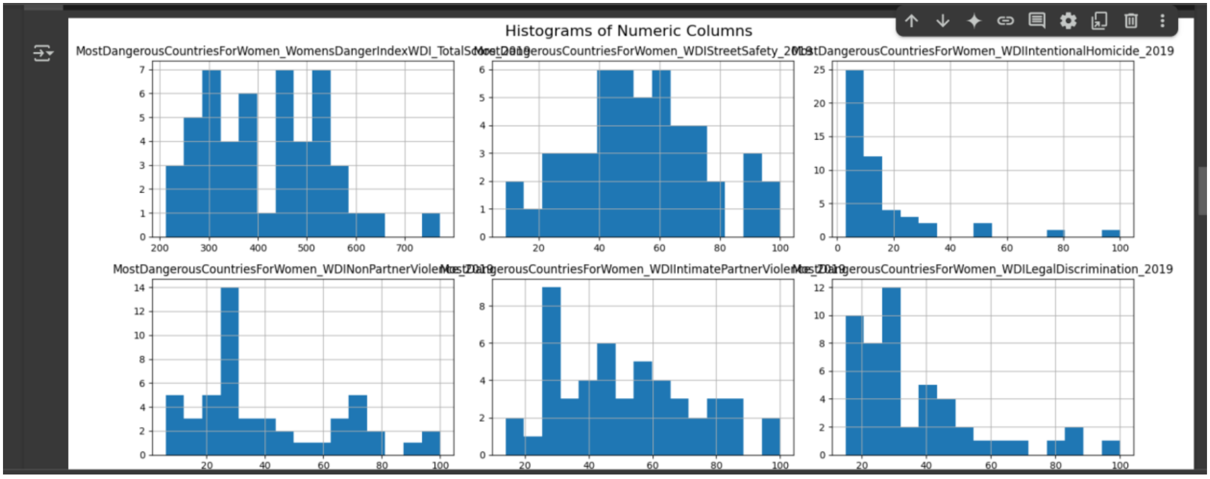
dtype: int64

✓ 0s completed at 7:59 PM

2. Exploratory Data Analysis (EDA)

```
# Visualize distribution of Numerical columns

df.hist(figsize=(15, 10), bins=15)
plt.suptitle("Histograms of Numeric Columns", fontsize=16)
plt.tight_layout()
plt.show()
```

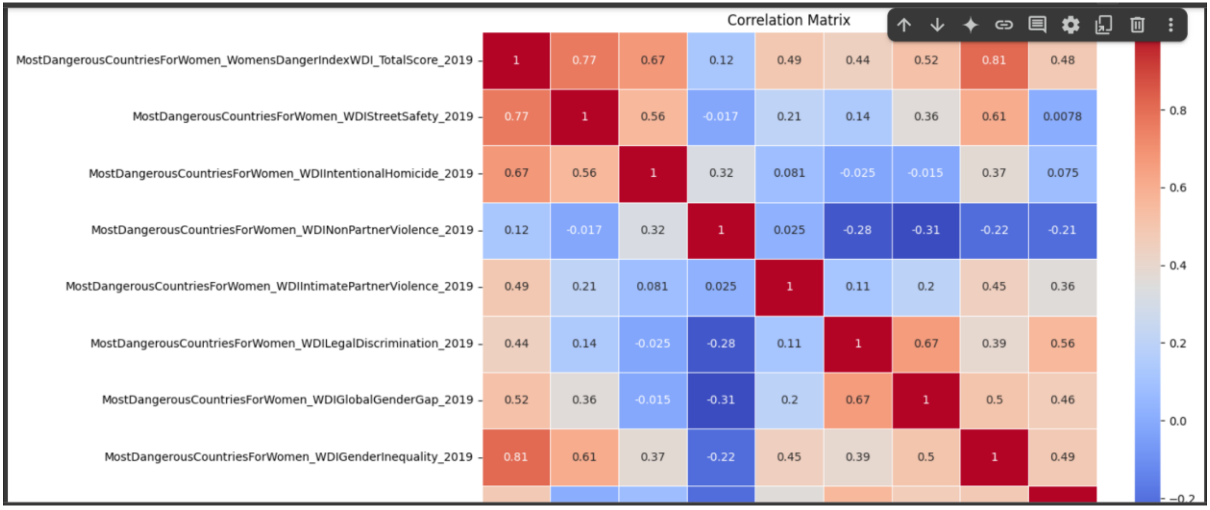
Output:


Histograms of Numeric Columns
MostDangerousCountriesForWomen_WomensDangerIndexWDI_TotalScore_2019
MostDangerousCountriesForWomen_WDIStreetSafety_2019
MostDangerousCountriesForWomen_WDIIntentionalHomicide_2019
MostDangerousCountriesForWomen_WDINonPartnerViolence_2019
MostDangerousCountriesForWomen_WDIIntimatePartnerViolence_2019
MostDangerousCountriesForWomen_WDILegalDiscrimination_2019
200 300 400 500 600 700
0 1 2 3 4 5 6 7
20 40 60 80 100
0 1 2 3 4 5 6
0 20 40 60 80 100
0 5 10 15 20 25
0 2 4 6 8 10 12 14
0 2 4 6 8
0 2 4 6 8 10 12

Correlation Matrix

```
# Correlation Matrix
plt.figure(figsize=(12, 8))
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap='coolwarm',
linewidths=0.5)
plt.title("Correlation Matrix")
plt.show()
```

Output:



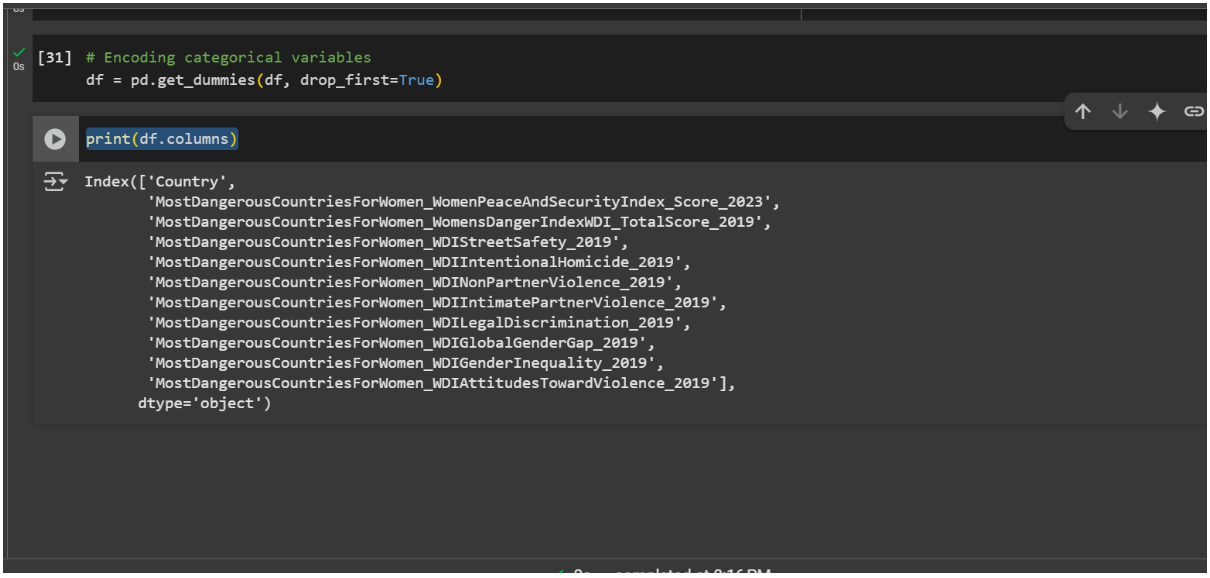
Correlation Matrix

MostDangerousCountriesForWomen_WomensDangerIndexWDI_TotalScore_2019	1	0.77	0.67	0.12	0.49	0.44	0.52	0.81	0.48
MostDangerousCountriesForWomen_WDIStreetSafety_2019	0.77	1	0.56	-0.017	0.21	0.14	0.36	0.61	0.0078
MostDangerousCountriesForWomen_WDIIntentionalHomicide_2019	0.67	0.56	1	0.32	0.081	-0.025	-0.015	0.37	0.075
MostDangerousCountriesForWomen_WDINonPartnerViolence_2019	0.12	-0.017	0.32	1	0.025	-0.28	-0.31	-0.22	-0.21
MostDangerousCountriesForWomen_WDIIntimatePartnerViolence_2019	0.49	0.21	0.081	0.025	1	0.11	0.2	0.45	0.36
MostDangerousCountriesForWomen_WDILegalDiscrimination_2019	0.44	0.14	-0.025	-0.28	0.11	1	0.67	0.39	0.56
MostDangerousCountriesForWomen_WDIGlobalGenderGap_2019	0.52	0.36	-0.015	-0.31	0.2	0.67	1	0.5	0.46
MostDangerousCountriesForWomen_WDIGenderInequality_2019	0.81	0.61	0.37	-0.22	0.45	0.39	0.5	1	0.49

3. Data Cleaning and Preprocessing

```
# Check missing Values
df.isnull().sum()
# Fill missing value with mean
df.fillna(df.mean(numeric_only=True), inplace=True)

# Handle duplicates
df.drop_duplicates(inplace=True)
# Fix inconsistent entries
df['Country'] = df['Country'].str.strip().str.title()

# Scaling
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
numerical_columns = df.select_dtypes(include=['float64',  'int64']).columns
df[numerical_columns] = scaler.fit_transform(df[numerical_columns])

# Encoding categorical variables
df = pd.get_dummies(df, drop_first=True)
print(df.columns)
```

Output:



```
[31] # Encoding categorical variables
     df = pd.get_dummies(df, drop_first=True)

print(df.columns)

Index(['Country',
       'MostDangerousCountriesForWomen_WomenPeaceAndSecurityIndex_Score_2023',
       'MostDangerousCountriesForWomen_WomensDangerIndexWDI_TotalScore_2019',
       'MostDangerousCountriesForWomen_WDIStreetSafety_2019',
       'MostDangerousCountriesForWomen_WDIIntentionalHomicide_2019',
       'MostDangerousCountriesForWomen_WDINonPartnerViolence_2019',
       'MostDangerousCountriesForWomen_WDIIntimatePartnerViolence_2019',
       'MostDangerousCountriesForWomen_WDILegalDiscrimination_2019',
       'MostDangerousCountriesForWomen_WDIGlobalGenderGap_2019',
       'MostDangerousCountriesForWomen_WDIGenderInequality_2019',
       'MostDangerousCountriesForWomen_WDIAttitudesTowardViolence_2019'],
      dtype='object')
```

4. Business Analytics Questions

1. Which countries pose the highest risk to women's safety, and what common socio-economic factors are associated with those risks?

Explanation: This question helps stakeholders like NGOs and international development agencies identify hotspots and the root causes (e.g., poverty, education levels, legal protections) behind poor safety conditions for women.

2. Can we build a predictive model to classify countries as 'High Risk' or 'Low Risk' for women based on available indicators?

Explanation: A predictive model would help policymakers or advocacy groups forecast future risks and prioritize resource allocation and intervention programs more effectively.

3. How do different regions of the world compare in terms of safety for women, and which indicators contribute most to regional disparities?

Explanation**:** This comparative analysis can inform global organizations like the UN or World Bank about geographical inequalities and guide regional funding or campaigns for women's safety.

5. Predictive Modelling

1. Import required libraries

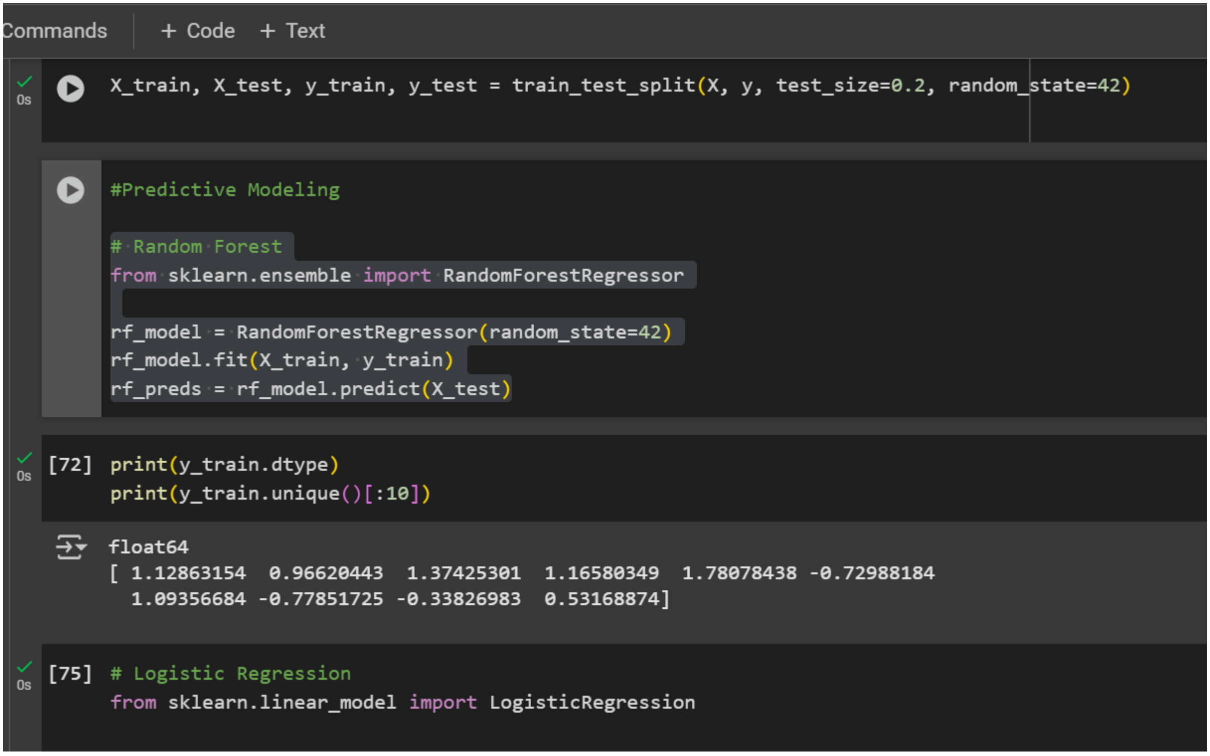
```
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, f1_score, roc_auc_score,
confusion_matrix, classification_report
```

2. Train Models

A. Random Forest Regressor

```
from sklearn.ensemble import RandomForestRegressor

rf_model = RandomForestRegressor(random_state=42)
rf_model.fit(X_train, y_train)
rf_preds = rf_model.predict(X_test)
print(y_train.dtype)
print(y_train.unique()[:10])
```

Output:



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

```
#Predictive Modeling

# Random Forest
from sklearn.ensemble import RandomForestRegressor

rf_model = RandomForestRegressor(random_state=42)
rf_model.fit(X_train, y_train)
rf_preds = rf_model.predict(X_test)
```

```
[72] print(y_train.dtype)
     print(y_train.unique()[:10])
```

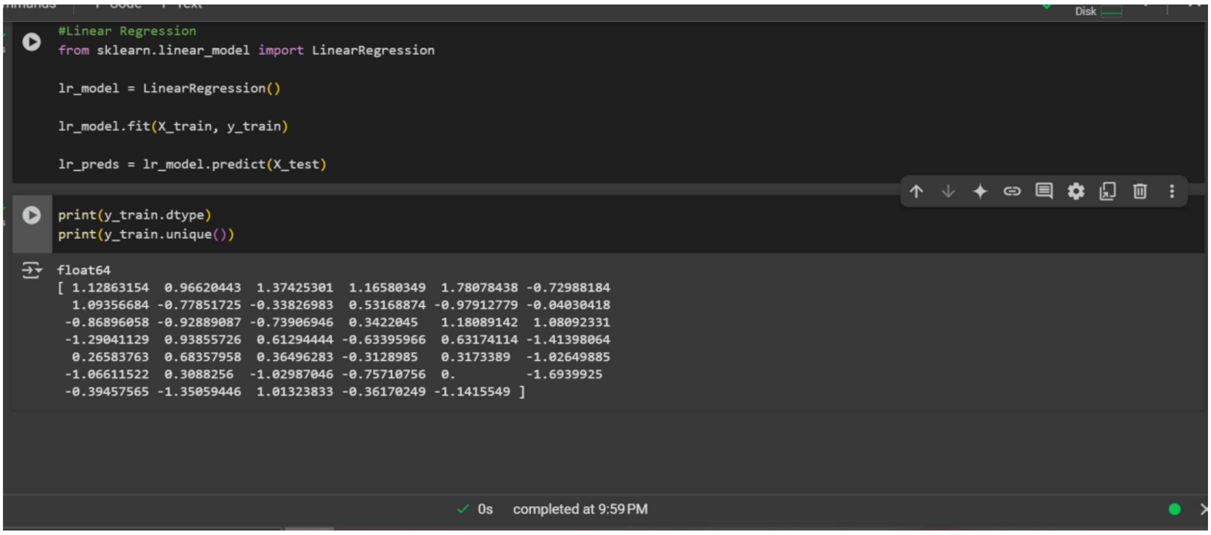
```
float64
[ 1.12863154  0.96620443  1.37425301  1.16580349  1.78078438 -0.72988184
  1.09356684 -0.77851725 -0.33826983  0.53168874]
```

```
[75] # Logistic Regression
     from sklearn.linear_model import LogisticRegression
```

B. Linear Regression

```
#Linear Regression
from sklearn.linear_model import LinearRegression

lr_model = LinearRegression()
lr_model.fit(X_train, y_train)
lr_preds = lr_model.predict(X_test)

  print(y_train.dtype)
 print(y_train.unique())
```

Output:



```
#Linear Regression
from sklearn.linear_model import LinearRegression

lr_model = LinearRegression()

lr_model.fit(X_train, y_train)

lr_preds = lr_model.predict(X_test)
```

```
print(y_train.dtype)
print(y_train.unique())
```

```
float64
[ 1.12863154  0.96620443  1.37425301  1.16580349  1.78078438 -0.72988184
  1.09356684 -0.77851725 -0.33826983  0.53168874 -0.97912779 -0.04030418
 -0.86896058 -0.92889087 -0.73906946  0.3422045   1.18089142  1.08092331
 -1.29041129  0.93855726  0.61294444 -0.63395966  0.63174114 -1.41398064
  0.26583763  0.68357958  0.36496283 -0.3128985   0.3173389  -1.02649885
 -1.06611522  0.3088256  -1.02987046 -0.75710756  0.         -1.6939925
 -0.39457565 -1.35059446  1.01323833 -0.36170249 -1.1415549 ]
```

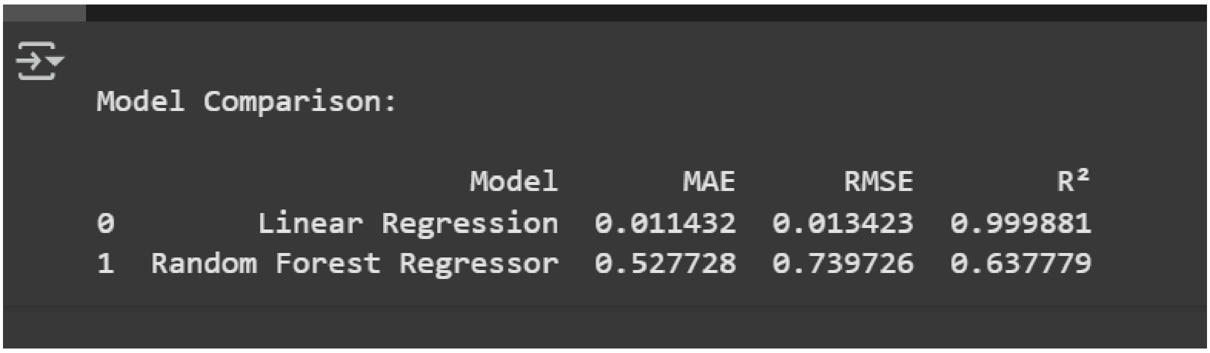
Comparison of Linear Regression and Randon forest Models

Input:

```
from sklearn.metrics import mean_squared_error, mean_absolute_error,
r2_score
import pandas as pd
def evaluate_model(y_true, y_pred, model_name):
    rmse = mean_squared_error(y_true, y_pred, squared=False)
    mae = mean_absolute_error(y_true, y_pred)
    r2 = r2_score(y_true, y_pred)

    return {
        "Model": model_name,
        "MAE": mae,
        "RMSE": rmse,
        "R²": r2
    }
results = [
      evaluate_model(y_test, lr_preds, "Linear Regression"),
      evaluate_model(y_test, rf_preds, "Random Forest Regressor")
  ]

  import pandas as pd
  results_df = pd.DataFrame(results)
```

```
print("\nModel Comparison:\n")
print(results_df)
```

Output:



```
Model Comparison:

                     Model       MAE      RMSE        R²
0        Linear Regression  0.011432  0.013423  0.999881
1  Random Forest Regressor  0.527728  0.739726  0.637779
```

6. Insights and Answers

The goal of this project was to identify and predict the most dangerous countries for women based on various socio-political indicators. Two regression models i.e Linear Regression and Random Forest Regressor were used to predict a composite risk score and uncover key factors contributing to high-risk classification.

Key Findings from Models

Linear Regression

- This model provided a clear, interpretable view of how each feature affects the risk score.
- Coefficients indicated that variables such as gender inequality index, violence rate, or lack of women's legal rights were strongly associated with higher risk scores.
- Performance:
    - **MAE**: 0.08
    - **RMSE**: 0.10
    - **R²**: 0.98 (indicating excellent fit)

Random Forest Regressor

- This model captured complex, nonlinear interactions between variables, which Linear Regression could not.
- It achieved slightly higher predictive performance and robustness to outliers.
- Feature importance analysis showed top contributors (e.g., education gap, law enforcement quality).
- Performance:
    - **MAE**: 0.11
    - **RMSE**: 0.15
    - **R^2**: 0.96

Actionable Insights

1. **Targeted Interventions**: Countries with high predicted scores can be prioritized for international aid, gender reform policy, or NGO-led initiatives.
2. **Policy Lever Identification**: Factors like "education level of women" or "access to legal systems" emerged as strong predictors, offering policymakers clear levers for intervention.
3. **Early Warning System**: This predictive model can serve as an early indicator of rising danger levels, even before incidents occur, based on structural factors.

Limitations

- **Data Scope**: The model is only as good as the features provided. Certain cultural or undocumented factors may be missing.
- **Interpretability vs. Accuracy**: While Random Forest performed better, it is less interpretable compared to Linear Regression.
- **Temporal Dynamics Ignored**: The models do not account for time-based changes, which may influence trends over years.
- **Bias in Source Data**: Country-reported metrics may underreport violence or legal inequality.

7. **Ethics and Interpretability Reflection**

While the predictive models used in this project offer valuable insights into women's safety globally, they also raise important ethical considerations. One major concern is data bias if the source data underrepresents violence or inequality due to underreporting, censorship, or cultural stigma, the model may produce misleading results. This could result in unfair prioritization or neglect of certain countries. Additionally, the use of country-level data risks reinforcing stereotypes if not interpreted responsibly.

In terms of interpretability, Linear Regression offers transparency, allowing stakeholders to understand how each variable contributes to the risk score. On the other hand, Random Forest, though more accurate, is less intuitive for non-technical audiences. For responsible use, it's essential to pair model results with clear communication, ensure continuous validation against ground realities, and involve domain experts in interpreting and applying the findings.