SC1015 TEAM 3

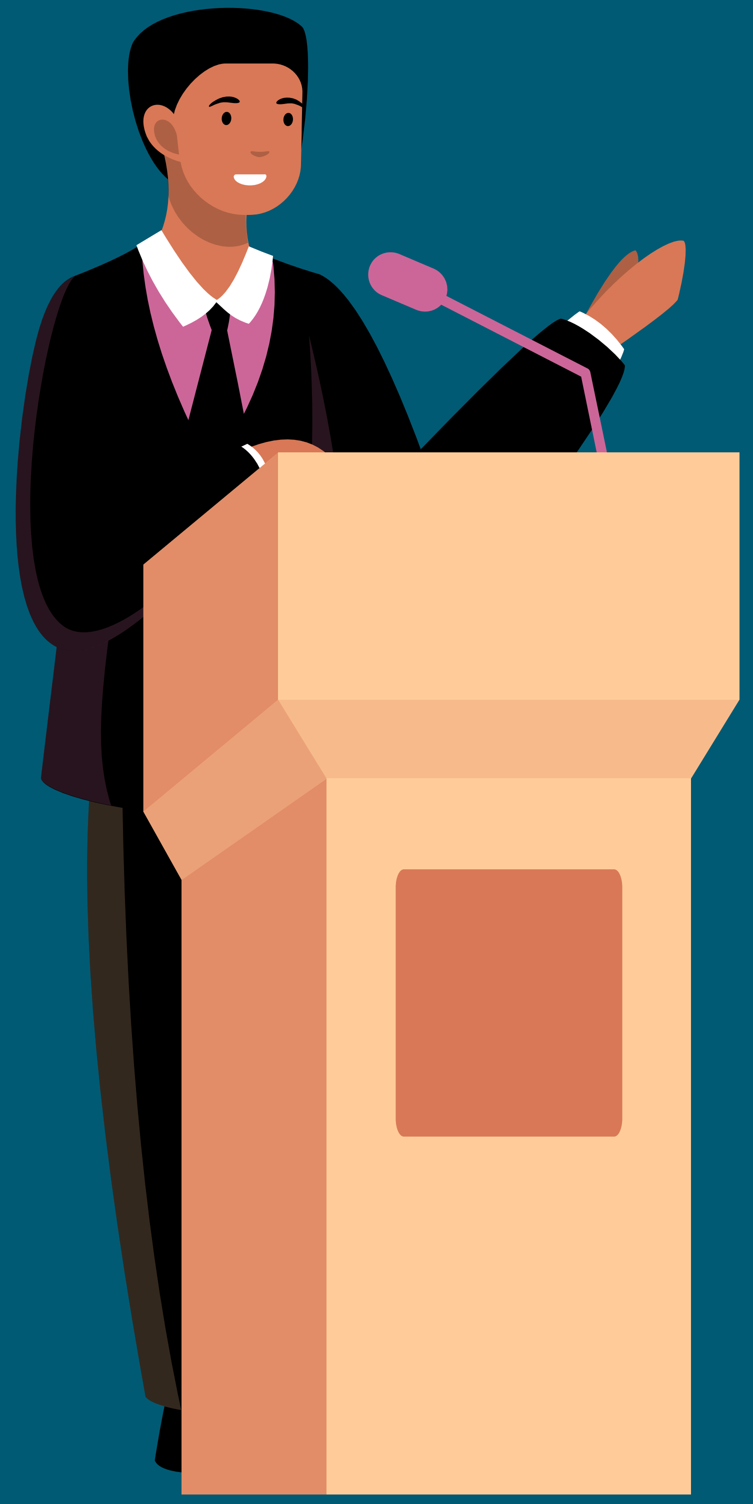
MINI PROJECT:
IMPACT OF THE GOVERNMENT ON HUMAN FREEDOM

Created by:
Syed Ridwan Kabir (U2321342C)
Evelyn Theresia Cuaca (U2320523L)

SAMPLE COLLECTION + PRACTICAL MOTIVATION

INTRODUCTION

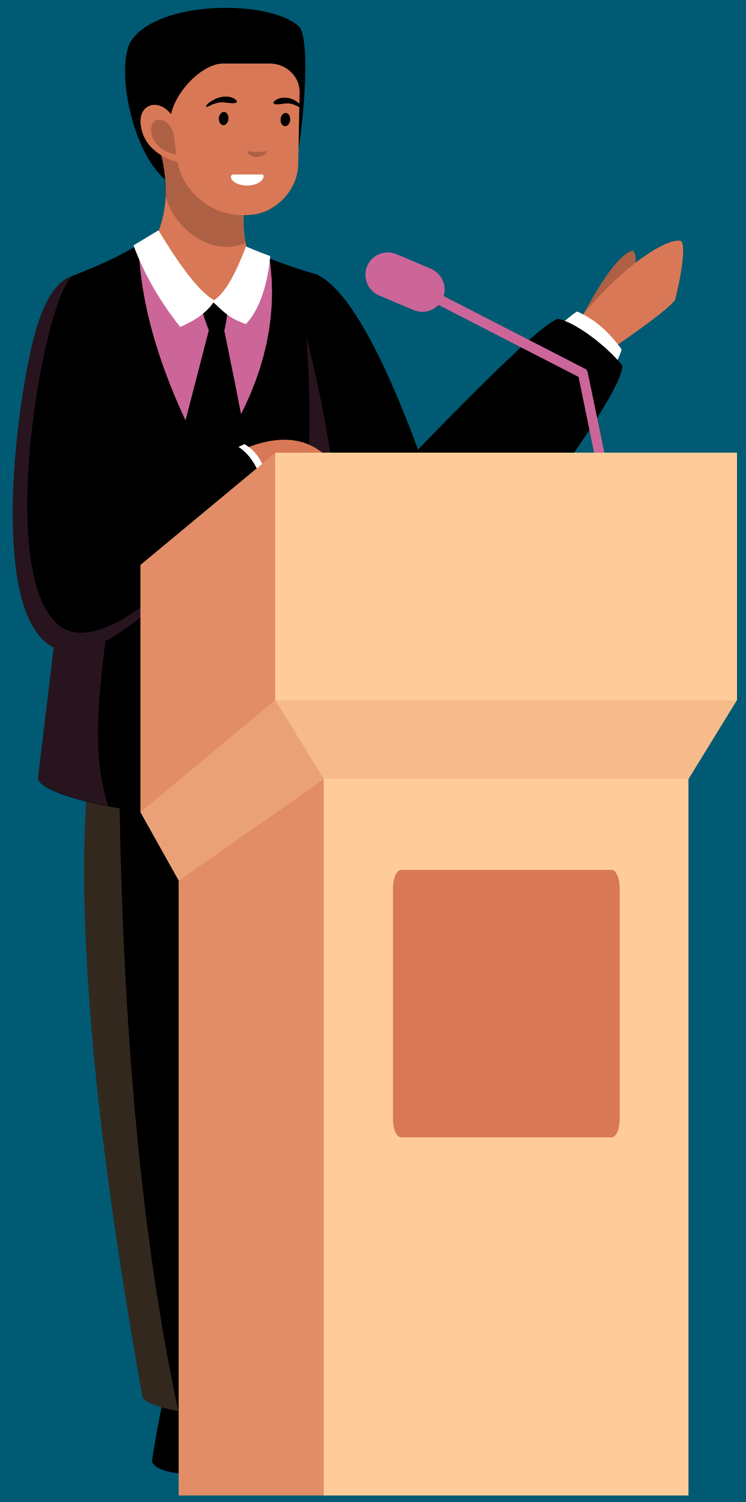
Governments play a pivotal role in shaping the socio-political landscape of their respective nations, influencing everything from economic policies to social rights and individual liberties.



INTRODUCTION

Governments play a pivotal role in shaping the socio-political landscape of their respective nations, influencing everything from economic policies to social rights and individual liberties.

As societies evolve, understanding the intricate dynamics between government efficacy and the maintenance of human freedoms becomes paramount.

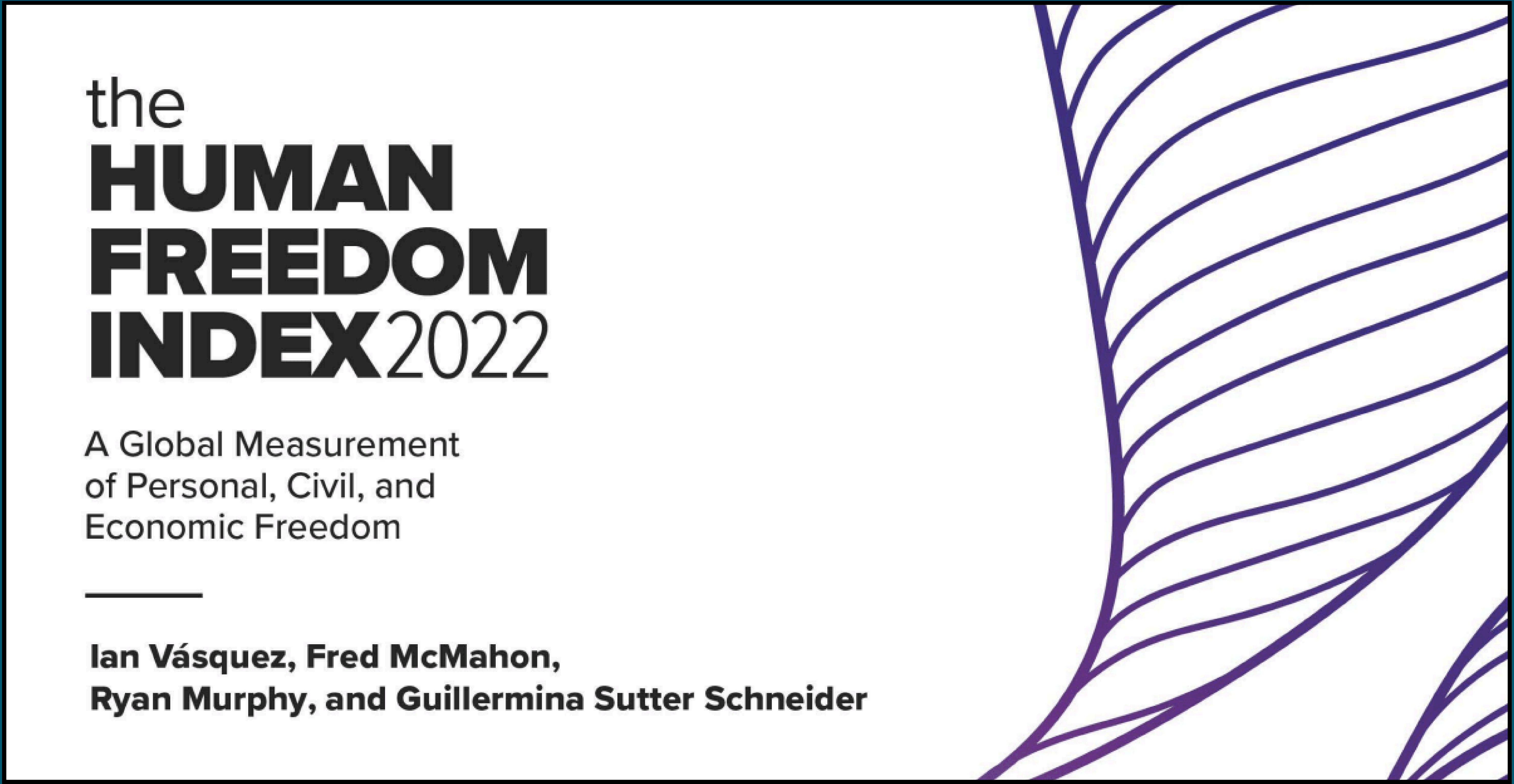


PURPOSE OF THIS PROJECT

Employing statistical analysis and machine learning techniques to forecast how the effectiveness of a government might impact future levels of human freedom within a given country.

PURPOSE OF THIS PROJECT

We seek not only to enhance our understanding of governance dynamics but also to offer insights that can inform policy-making and foster the advancement of individual liberties worldwide.



Human freedom index

Kaggle: https://www.kaggle.com/datasets/gsutters/the-human-freedom-index

the
HUMAN FREEDOM INDEX 2022

A Global Measurement of Personal, Civil, and Economic Freedom

Ian Vásquez, Fred McMahon, Ryan Murphy, and Guillermina Sutter Schneider

Rule of Law

Security and Safety

Movement

Religion

Association/ Assembly/ Civil Society

Expression & Information

Relationship

Size of Government

Legal System and Property Rights

Sound Money

Freedom to Trade Internationally

Regulation

In [43]:
```python
print("Number of rows:", freedom_data.shape[0])
print("Number of columns:", freedom_data.shape[1])
```

```
Number of rows: 3465
Number of columns: 141
```

```
year
countries
region
hf_score
hf_rank
hf_quartile
pf_rol_procedural
pf_rol_civil
pf_rol_criminal
pf_rol_vdem
pf_rol
pf_ss_homicide
pf_ss_homicide_data
pf_ss_disappearances_disap
pf_ss_disappearances_violent
pf_ss_disappearances_violent_data
pf_ss_disappearances_organized
pf_ss_disappearances_fatalities
pf_ss_disappearances_fatalities_data
pf_ss_disappearances_injuries
pf_ss_disappearances_injuries_data
pf_ss_disappearances_torture
pf_ss_killings
pf_ss_disappearances
pf_ss
pf_movement_vdem_foreign
pf_movement_vdem_men
pf_movement_vdem_women
pf_movement_vdem
pf_movement_cld
pf_movement
pf_religion_freedom_vdem
pf_religion_freedom_cld
pf_religion_freedom
pf_religion_suppression
pf_religion
pf_assembly_entry
pf_assembly_freedom_house
pf_assembly_freedom_bti
pf_assembly_freedom_cld
pf_assembly_freedom
pf_assembly_parties_barriers
pf_assembly_parties_bans
pf_assembly_parties_auton
pf_assembly_parties
pf_assembly_civil
pf_assembly
pf_expression_direct_killed
pf_expression_direct_killed_data
pf_expression_direct_jailed
pf_expression_direct_jailed_data
pf_expression_direct
pf_expression_vdem_cultural
pf_expression_vdem_harass
pf_expression_vdem_gov
pf_expression_vdem_internet
pf_expression_vdem_selfcens
pf_expression_vdem
pf_expression_house
pf_expression_bti
pf_expression_cld
pf_expression
pf_identity_same_m
pf_identity_same_f
pf_identity_same
pf_identity_divorce
pf_identity_inheritance_widows
pf_identity_inheritance_daughters
pf_identity_inheritance
pf_identity_fgm
pf_identity
pf_score
pf_rank
ef_government_consumption
ef_government_consumption_data
ef_government_transfers
ef_government_transfers_data
ef_government_investment
ef_government_investment_data
ef_government_tax_income
ef_government_tax_income_data
ef_government_tax_payroll
ef_government_tax_payroll_data
ef_government_tax
ef_government_soa
ef_government
ef_legal_judicial
ef_legal_courts
ef_legal_protection
ef_legal_military
ef_legal_integrity
ef_legal_enforcement
ef_legal_regulatory
ef_legal_police
ef_gender
ef_legal
ef_money_growth
ef_money_growth_data
ef_money_sd
ef_money_sd_data
ef_money_inflation
ef_money_inflation_data
ef_money_currency
ef_money
ef_trade_tariffs_revenue
ef_trade_tariffs_revenue_data
ef_trade_tariffs_mean
ef_trade_tariffs_mean_data
ef_trade_tariffs_sd
ef_trade_tariffs_sd_data
ef_trade_tariffs
ef_trade_regulatory_nontariff
ef_trade_regulatory_compliance
ef_trade_regulatory
ef_trade_black
ef_trade_movement_open
ef_trade_movement_capital
ef_trade_movement_visit
ef_trade_movement
ef_trade
ef_regulation_credit_ownership
ef_regulation_credit_private
ef_regulation_credit_interest
ef_regulation_credit
ef_regulation_labor_minwage
ef_regulation_labor_firing
ef_regulation_labor_bargain
ef_regulation_labor_hours
ef_regulation_labor_dismissal
ef_regulation_labor_conscription
ef_regulation_labor
ef_regulation_business_adm
ef_regulation_business_burden
ef_regulation_business_start
ef_regulation_business_impartial
ef_regulation_business_licensing
ef_regulation_business_compliance
ef_regulation_business
ef_regulation
ef_score
ef_rank
```

DATA PREPARATION + PROBLEM FORMULATION

PROBLEM STATEMENT

Forecasting the impact of the government on Freedom (How might the effectiveness of a government impact future levels of human freedom and what can be done to improve future levels of human freedom?)

CLEANING DATA (>40% NULL)

In [45]:
```
null_counts = freedom_data.isnull().sum()
pd.set_option('display.max_columns', None)
```

In [46]:
```
print("Number of null values for each column:")
for column, count in null_counts.items():
    print(f"Column '{column}': {count}")
```

In [47]:
```
null_percentage_before = freedom_data.isnull().mean() * 100
columns_to_drop = null_percentage_before[null_percentage_before > 40].index
freedom_dropped = freedom_data.drop(columns=columns_to_drop)
null_percentage_after = freedom_dropped.isnull().mean() * 100
```

CLEANING DATA (>40% NULL)

```
Column 'pf_ss_disappearances_violent': 0
Column 'pf_ss_disappearances_violent_data': 0
Column 'pf_ss_disappearances_organized': 1494
Column 'pf_ss_disappearances_fatalities': 0
Column 'pf_ss_disappearances_fatalities_data': 0
Column 'pf_ss_disappearances_injuries': 0
Column 'pf_ss_disappearances_injuries_data': 0
Column 'pf_ss_disappearances_torture': 63
Column 'pf_ss_killings': 63
Column 'pf_ss_disappearances': 0
Column 'pf_ss': 0
Column 'pf_movement_vdem_foreign': 63
Column 'pf_movement_vdem_men': 63
Column 'pf_movement_vdem_women': 63
Column 'pf_movement_vdem': 63
Column 'pf_movement_cld': 33
Column 'pf_movement': 0
Column 'pf_religion_freedom_vdem': 63
Column 'pf_religion_freedom_cld': 33
Column 'pf_religion_freedom': 0
Column 'pf_religion_suppression': 63
Column 'pf_religion': 0
Column 'pf_assembly_entry': 63
Column 'pf_assembly_freedom_house': 0
Column 'pf_assembly_freedom_bti': 970
Column 'pf_assembly_freedom_cld': 33
Column 'pf_assembly_freedom': 0
Column 'pf_assembly_parties_barriers': 63
Column 'pf_assembly_parties_bans': 63
Column 'pf_assembly_parties_auton': 146
Column 'pf_assembly_parties': 63
Column 'pf_assembly_civil': 63
Column 'pf_assembly': 0
Column 'pf_expression_direct_killed': 0
Column 'pf_expression_direct_killed_data': 0
Column 'pf_expression_direct_jailed': 0
Column 'pf_expression_direct_jailed_data': 0
Column 'pf_expression_direct': 0
Column 'pf_expression_vdem_cultural': 63
Column 'pf_expression_vdem_harass': 63
Column 'pf_expression_vdem_gov': 63
Column 'pf_expression_vdem_internet': 67
Column 'pf_expression_vdem_selfcens': 63
Column 'pf_expression_vdem': 63
```

```
plt.title('Null Values After Dropping Columns')
plt.ylabel('Columns')
plt.xlabel('Percentage of Null Values')
plt.tight_layout()
plt.show()
```

CLEANING DATA (GROUP BY COUNTRY)

```
In [55]:  numeric_columns = freedom_dropped.select_dtypes(include=['int64', 'float64'])
          group_by_countries = freedom_dropped.groupby('countries')

In [56]:  for column in numeric_columns:
              mean_by_country = group_by_countries[column].transform('mean')
              freedom_dropped[column].fillna(mean_by_country, inplace=True)
```

```
pf_assembly_entry                      63
pf_assembly_freedom_house               0
pf_assembly_freedom_bti               777
pf_assembly_freedom_cld                21
pf_assembly_freedom                     0
pf_assembly_parties_barriers           63
pf_assembly_parties_bans               63
pf_assembly_parties_auton             126
pf_assembly_parties                    63
pf_assembly_civil                      63
pf_assembly                             0
pf_expression_direct_killed             0
pf_expression_direct_killed_data        0
pf_expression_direct_jailed             0
pf_expression_direct_jailed_data        0
pf_expression_direct                    0
pf_expression_vdem_cultural            63
pf_expression_vdem_harass              63
pf_expression_vdem_gov                 63
pf_expression_vdem_internet            63
pf_expression_vdem_selfcens            63
pf_expression_vdem                     63
pf_expression_house                     0
pf_expression_bti                     777
pf_expression_cld                      21
pf_expression                           0
pf_identity_same_m                      0
pf_identity_same_f                      0
pf_identity_same                        0
pf_identity_divorce                    21
pf_identity_inheritance                 0
pf_identity_fgm                         0
pf_identity                             0
pf_score                                0
pf_rank                                 0
ef_government_consumption               0
ef_government_consumption_data          0
ef_government_transfers               168
ef_government_transfers_data          168
ef_government_investment              105
ef_government_investment_data         105
ef_government_tax_income                0
ef_government_tax_income_data         570
```

CLEANING DATA (GROUP BY REGION)

```
In [58]:
group_by_region = freedom_dropped.groupby('region')
numeric_columns = freedom_dropped.select_dtypes(include=[np.int64, np.float64])

In [59]:
for column in numeric_columns:
    mean_by_region = group_by_region[column].transform('mean')
    freedom_dropped[column].fillna(mean_by_region, inplace=True)
```

```
pf_religion_freedom                      0
pf_religion_suppression                  0
pf_religion                              0
pf_assembly_entry                        0
pf_assembly_freedom_house                0
pf_assembly_freedom_bti                420
pf_assembly_freedom_cld                  0
pf_assembly_freedom                      0
pf_assembly_parties_barriers             0
pf_assembly_parties_bans                 0
pf_assembly_parties_auton                0
pf_assembly_parties                      0
pf_assembly_civil                        0
pf_assembly                              0
pf_expression_direct_killed              0
pf_expression_direct_killed_data         0
pf_expression_direct_jailed              0
pf_expression_direct_jailed_data         0
pf_expression_direct                     0
pf_expression_vdem_cultural              0
pf_expression_vdem_harass                0
pf_expression_vdem_gov                   0
pf_expression_vdem_internet              0
pf_expression_vdem_selfcens              0
pf_expression_vdem                       0
pf_expression_house                      0
pf_expression_bti                      420
pf_expression_cld                        0
pf_expression                            0
pf_identity_same_m                       0
pf_identity_same_f                       0
pf_identity_same                         0
pf_identity_divorce                      0
pf_identity_inheritance                  0
pf_identity_fgm                          0
pf_identity                              0
pf_score                                 0
pf_rank                                  0
ef_government_consumption                0
ef_government_consumption_data           0
ef_government_transfers                  0
ef_government_transfers_data             0
ef_government_investment                 0
```

CLEANING DATA (LEFTOVER NULL VARIABLES)

```
In [61]: freedom_final = freedom_dropped.dropna(axis=1)

In [62]: freedom_final.isnull().sum()
```

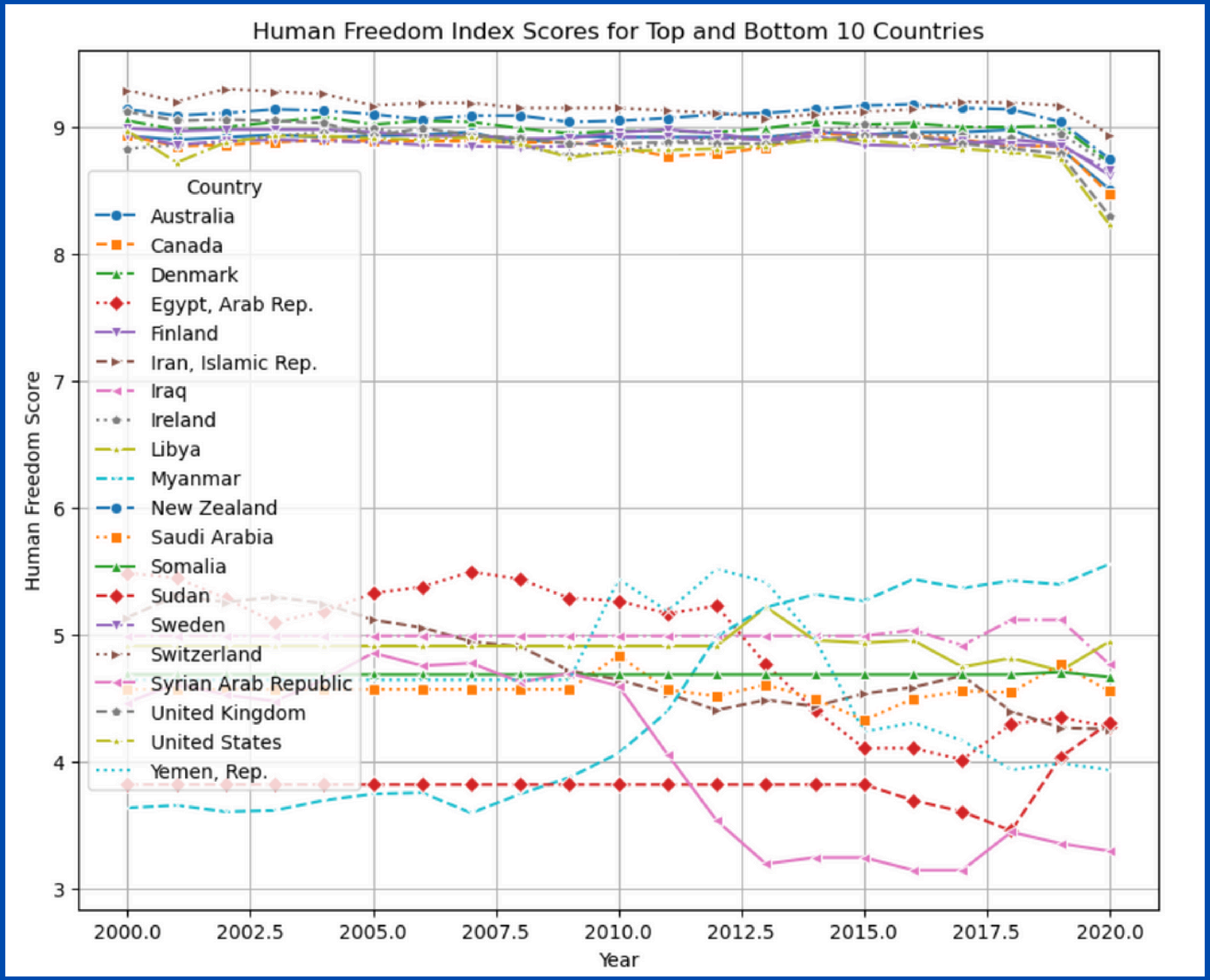
```
pf_ss_killings                      0
pf_ss_disappearances                0
pf_ss                               0
pf_movement_vdem_foreign            0
pf_movement_vdem_men                0
pf_movement_vdem_women              0
pf_movement_vdem                    0
pf_movement_cld                     0
pf_movement                         0
pf_religion_freedom_vdem            0
pf_religion_freedom_cld             0
pf_religion_freedom                 0
pf_religion_suppression             0
pf_religion                         0
pf_assembly_entry                   0
pf_assembly_freedom_house           0
pf_assembly_freedom_cld             0
pf_assembly_freedom                 0
pf_assembly_parties_barriers        0
pf_assembly_parties_bans            0
pf_assembly_parties_auton           0
pf_assembly_parties                 0
pf_assembly_civil                   0
pf_assembly                         0
pf_expression_direct_killed         0
pf_expression_direct_killed_data    0
pf_expression_direct_jailed         0
pf_expression_direct_jailed_data    0
pf_expression_direct                0
pf_expression_vdem_cultural         0
pf_expression_vdem_harass           0
pf_expression_vdem_gov              0
pf_expression_vdem_internet         0
pf_expression_vdem_selfcens         0
pf_expression_vdem                  0
pf_expression_house                 0
pf_expression_cld                   0
pf_expression                       0
pf_identity_same_m                  0
pf_identity_same_f                  0
pf_identity_same                    0
pf_identity_divorce                 0
pf_identity_inheritance             0
pf_identity_fgm                     0
```

EXPLORATORY ANALYSIS + STATISTICAL DESCRIPTION

TOP AND BOTTOM 10 COUNTRIES

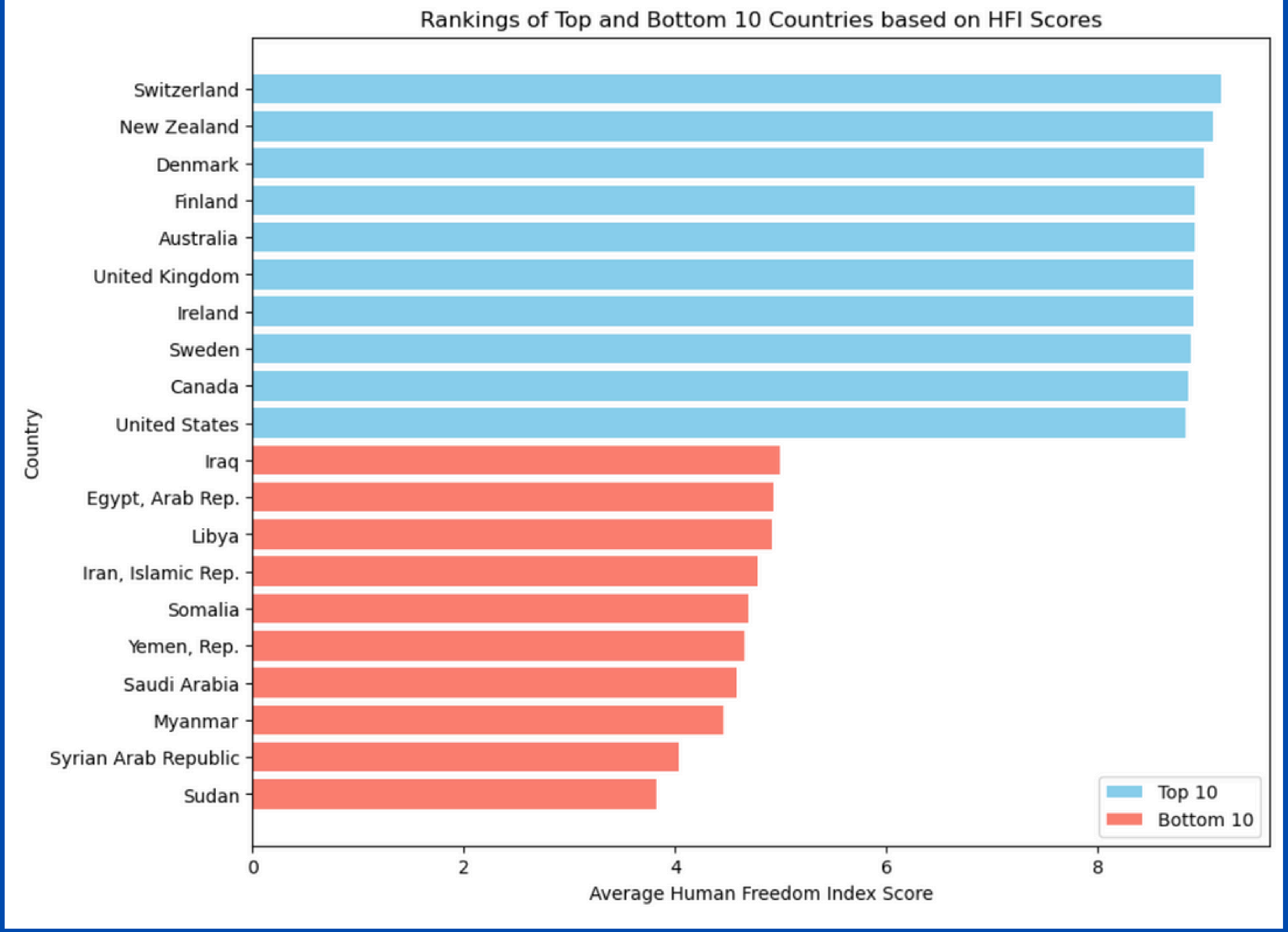
```
[21]: average_hf_scores = freedom_final.groupby('countries')['hf_score'].mean()
      sorted_countries = average_hf_scores.sort_values(ascending=False)
      top_10_countries = sorted_countries.head(10)
      bottom_10_countries = sorted_countries.tail(10)
      top_bottom_countries_data = freedom_final[freedom_final['countries'].isin(top_10_countries.index) |
                                   freedom_final['countries'].isin(bottom_10_countries.index)]
```

```
[22]: plt.figure(figsize=(20, 8))

      line_styles = ['-', '--', '-.', ':']
      markers = ['o', 's', '^', 'D', 'v', '>', '<', 'p', '*', 'x']
      for i, country in enumerate(top_bottom_countries_data['countries'].unique()):
          data = top_bottom_countries_data[top_bottom_countries_data['countries'] == country]
          linestyle = line_styles[i % len(line_styles)]
          marker = markers[i % len(markers)]
          sns.lineplot(data=data, x='year', y='hf_score', label=country, linestyle=linestyle, marker=marker)

      plt.title('Human Freedom Index Scores for Top and Bottom 10 Countries')
      plt.xlabel('Year')
      plt.ylabel('Human Freedom Score')
      plt.legend(title='Country')
      plt.grid(True)
      plt.show()
```



TOP AND BOTTOM 10 COUNTRIES

```
In [70]:
plt.figure(figsize=(10, 8))
plt.barh(top_10_countries.index, top_10_countries.values, color='skyblue', label='Top 10')
plt.barh(bottom_10_countries.index, bottom_10_countries.values, color='salmon', label='Bottom 10')

plt.xlabel('Average Human Freedom Index Score')
plt.ylabel('Country')
plt.title('Rankings of Top and Bottom 10 Countries based on HFI Scores')
plt.legend()
plt.gca().invert_yaxis()
plt.show()
```



FEATURE IMPORTANCE - TOP 10 COUNTRIES

[24]:
```
from sklearn.linear_model import LinearRegression
columns_to_exclude = ['year', 'region', 'countries', 'ef_score', 'ef_rank', 'pf_score', 'pf_rank', 'hf_rank', 'hf_quartile']
```

[25]:
```
top_10_countries_data = top_bottom_countries_data[top_bottom_countries_data['countries'].isin(top_10_countries.index)]
X_top_10 = top_10_countries_data.drop(columns=columns_to_exclude + ['hf_score'])
y_top_10 = top_10_countries_data['hf_score']
```

[26]:
```
model_top_10 = LinearRegression()
model_top_10.fit(X_top_10, y_top_10)
```

[26]:
```
▾ LinearRegression
LinearRegression()
```

FEATURE IMPORTANCE - TOP 10 COUNTRIES

```
[27]: feature_importances_top_10 = model_top_10.coef_
      absolute_importances_top_10 = np.abs(feature_importances_top_10)
      sorted_indices_top_10 = np.argsort(absolute_importances_top_10)[::-1]
      sorted_features_top_10 = X_top_10.columns[sorted_indices_top_10]
      sorted_importances_top_10 = absolute_importances_top_10[sorted_indices_top_10]
```

```
[28]: plt.figure(figsize=(10, 25))
      plt.barh(sorted_features_top_10, sorted_importances_top_10, color='skyblue')
      plt.xlabel('Absolute Coefficient Value')
      plt.ylabel('Feature')
      plt.title('Feature Importances for HF Scores - Top 10 Countries')
      plt.show()
```


Feature Importances for HF Scores - Top 10 Countries
ef_trade_black
ef_money_currency
ef_gender
pf_identity_divorce
pf_identity_same_f
pf_identity_same_m
pf_religion_freedom_cld
pf_assembly_freedom_cld
pf_ss_disappearances_injuries_data
pf_ss_disappearances_violent_data
pf_ss_disappearances_fatalities_data
ef_trade_tariffs_sd_data
ef_regulation_labor_hours
ef_regulation_labor_bargain
ef_government_investment_data
ef_regulation_labor_conscription
ef_regulation_labor_minwage
pf_ss_disappearances_fatalities
ef_money_growth_data
ef_trade_tariffs_mean
ef_trade_movement_open
pf_rol_vdem
ef_regulation_labor_firing
pf_expression_direct_killed_data
ef_trade_movement_visit
ef_money_inflation_data
pf_ss_disappearances_injuries
pf_expression_vdem_gov
pf_assembly_parties_auton
ef_regulation_business_compliance
pf_ss_disappearances_torture
ef_trade_movement_capital
ef_government_transfers
ef_government_consumption
pf_expression_vdem_harass
ef_regulation_business_licensing
ef_regulation_business_burden
ef_trade_tariffs_mean_data
ef_regulation_business_impartial
pf_ss_disappearances_disap
ef_regulation_business_adm
pf_assembly_parties_barriers
ef_trade_tariffs_revenue
pf_assembly_parties
pf_ss_killings
ef_money_sd

FEATURE IMPORTANCE - BTM 10 COUNTRIES

[29]:
```
bottom_10_countries_data = top_bottom_countries_data[top_bottom_countries_data['countries'].isin(bottom_10_countries.index)]
X_bottom_10 = bottom_10_countries_data.drop(columns=columns_to_exclude + ['hf_score'])
y_bottom_10 = bottom_10_countries_data['hf_score']
```

[30]:
```
model_bottom_10 = LinearRegression()
model_bottom_10.fit(X_bottom_10, y_bottom_10)
```

[30]:
```
▾ LinearRegression
LinearRegression()
```

FEATURE IMPORTANCE - BTM 10 COUNTRIES

```
[31]: feature_importances_bottom_10 = model_bottom_10.coef_
      absolute_importances_bottom_10 = np.abs(feature_importances_bottom_10)
      sorted_indices_bottom_10 = np.argsort(absolute_importances_bottom_10)[::-1]
      sorted_features_bottom_10 = X_bottom_10.columns[sorted_indices_bottom_10]
      sorted_importances_bottom_10 = absolute_importances_bottom_10[sorted_indices_bottom_10]
```

```
[32]: plt.figure(figsize=(10, 25))
      plt.barh(sorted_features_bottom_10, sorted_importances_bottom_10, color='salmon')
      plt.xlabel('Absolute Coefficient Value')
      plt.ylabel('Feature')
      plt.title('Feature Importances for HF Scores - Bottom 10 Countries')
      plt.show()
```

```
plt.show()
```


Feature Importances for HF Scores - Bottom 10 Countries
pf_identity_same_m
pf_ss_disappearances_violent_data
pf_ss_disappearances_fatalities_data
pf_ss_disappearances_injuries_data
pf_expression_vdem
pf_expression_direct_killed_data
pf_ss_homicide
ef_money_growth_data
ef_government_investment_data
ef_regulation_business_licensing
pf_expression_direct_jailed_data
pf_ss_disappearances_violent
ef_legal_police
ef_trade_tariffs_sd_data
pf_ss_homicide_data
ef_money_inflation_data
pf_expression_vdem_selfcens
ef_legal_military
pf_ss_disappearances_injuries
ef_trade_tariffs_revenue_data
ef_money_growth
pf_expression_direct_jailed
ef_legal_protection
pf_movement_vdem
pf_movement
pf_identity
ef_money
ef_legal_judicial
pf_expression_direct
pf_assembly_freedom_house
ef_legal_regulatory
pf_assembly_parties_auton
ef_regulation_labor_conscription
pf_ss_disappearances_fatalities
pf_expression_direct_killed
ef_regulation_business_start
ef_trade_tariffs_mean_data
pf_ss_killings
pf_religion_freedom
ef_regulation_business_adm
ef_regulation_credit_ownership
ef_trade_tariffs_sd
ef_government_tax
ef_trade_regulatory_nontariff
pf_movement_cld
ef_money_inflation
ef_legal_courts

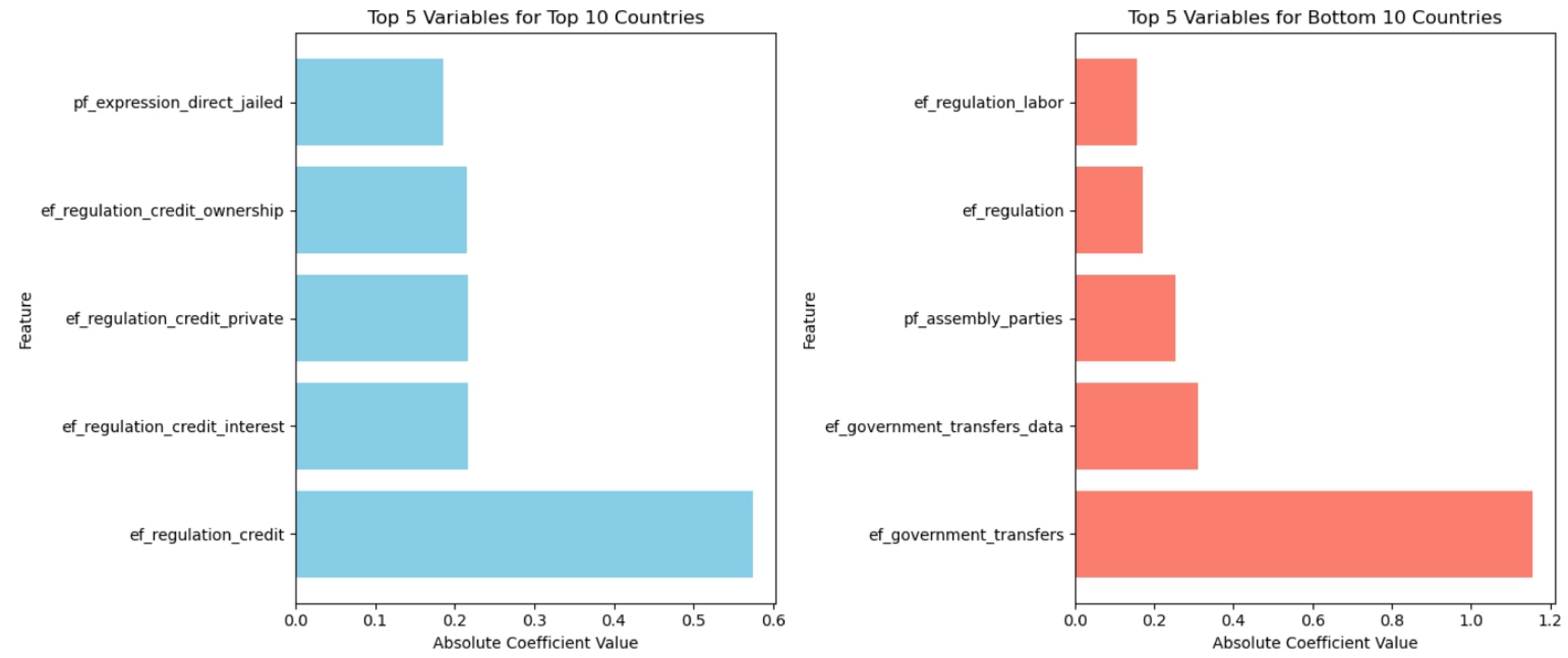
ANALYTICAL VISUALIZATION + PATTERN RECOGNITION

DETERMINING MOST SIGNIFICANT VARIABLES

```
[33]: top_5_top_countries = sorted_features_top_10[:5]
      importance_top_countries = sorted_importances_top_10[:5]
      top_5_bottom_countries = sorted_features_bottom_10[:5]
      importance_bottom_countries = sorted_importances_bottom_10[:5]
```

```
[34]: fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(14, 6))

      axes[0].barh(top_5_top_countries, importance_top_countries, color='skyblue')
      axes[0].set_xlabel('Absolute Coefficient Value')
      axes[0].set_ylabel('Feature')
      axes[0].set_title('Top 5 Variables for Top 10 Countries')

      axes[1].barh(top_5_bottom_countries, importance_bottom_countries, color='salmon')
      axes[1].set_xlabel('Absolute Coefficient Value')
      axes[1].set_ylabel('Feature')
      axes[1].set_title('Top 5 Variables for Bottom 10 Countries')

      plt.tight_layout()
      plt.show()
```

TOP 5 VARIABLES



TOP 5 VARIABLES (TOP 10 COUNTRIES)

1. ef_regulation_credit: This variable represents the level of regulation on credit markets in the top 10 countries. A higher value suggests less regulation, which may indicate greater ease of access to credit for individuals and businesses. Less restrictive credit regulations can facilitate economic growth, entrepreneurship, and investment, which are essential factors contributing to human freedom.

TOP 5 VARIABLES (TOP 10 COUNTRIES)

2. ef_regulation_credit_interest: This variable refers to the regulation of interest rates in credit markets. Lower regulation on interest rates allows market forces to determine rates, fostering competition and potentially leading to more favorable borrowing conditions for consumers and businesses. It can stimulate economic activity and promote financial inclusion, contributing to higher levels of human freedom.

TOP 5 VARIABLES (TOP 10 COUNTRIES)

3. ef_regulation_credit_private: This variable relates to regulations governing private credit provision. Less stringent regulations on private credit may encourage a vibrant credit market with diverse lending options, enabling individuals and businesses to access financing more easily. A conducive environment for private credit can foster economic growth and empower individuals to pursue their aspirations, enhancing human freedom.

TOP 5 VARIABLES (TOP 10 COUNTRIES)

4. ef_regulation_credit_ownership: This variable reflects regulations concerning the ownership structure of credit institutions. Policies that promote diverse ownership models and competition in the credit sector can lead to more efficient allocation of capital and better services for consumers. A competitive credit market with varied ownership structures can support economic dynamism and resilience, thereby positively influencing human freedom.

TOP 5 VARIABLES (TOP 10 COUNTRIES)

5. pf_expression_direct_jailed: This variable pertains to the freedom of individuals from being directly jailed for expression-related offenses. A higher score indicates greater protection of freedom of expression, which is a fundamental aspect of human freedom. Robust legal protections for freedom of expression foster open discourse, political participation, and the exchange of ideas, contributing to a more vibrant and inclusive society.

TOP 5 VARIABLES (BOTTOM 10 COUNTRIES)

1. ef_government_transfers: This variable represents government transfers as a percentage of GDP in the bottom 10 countries. High levels of government transfers may indicate extensive welfare programs or social safety nets aimed at addressing poverty and inequality. However, heavy reliance on transfers could signal economic inefficiencies or insufficient investment in productive sectors, potentially hindering long-term economic growth and human freedom.

TOP 5 VARIABLES (BOTTOM 10 COUNTRIES)

2. ef_government_transfers_data: This variable likely reflects the availability and reliability of data on government transfers. Adequate data on government spending is crucial for transparency, accountability, and effective policymaking. Weaknesses in data collection and reporting may undermine governance effectiveness and limit the ability to address socio-economic challenges, impacting human freedom.

TOP 5 VARIABLES (BOTTOM 10 COUNTRIES)

3. pf_assembly_parties: This variable relates to restrictions on political party formation and operation. Stringent regulations on political parties can impede political pluralism, democratic participation, and representation. Limited political competition and inclusivity may lead to governance deficits and marginalized voices, constraining human freedom and democratic development.

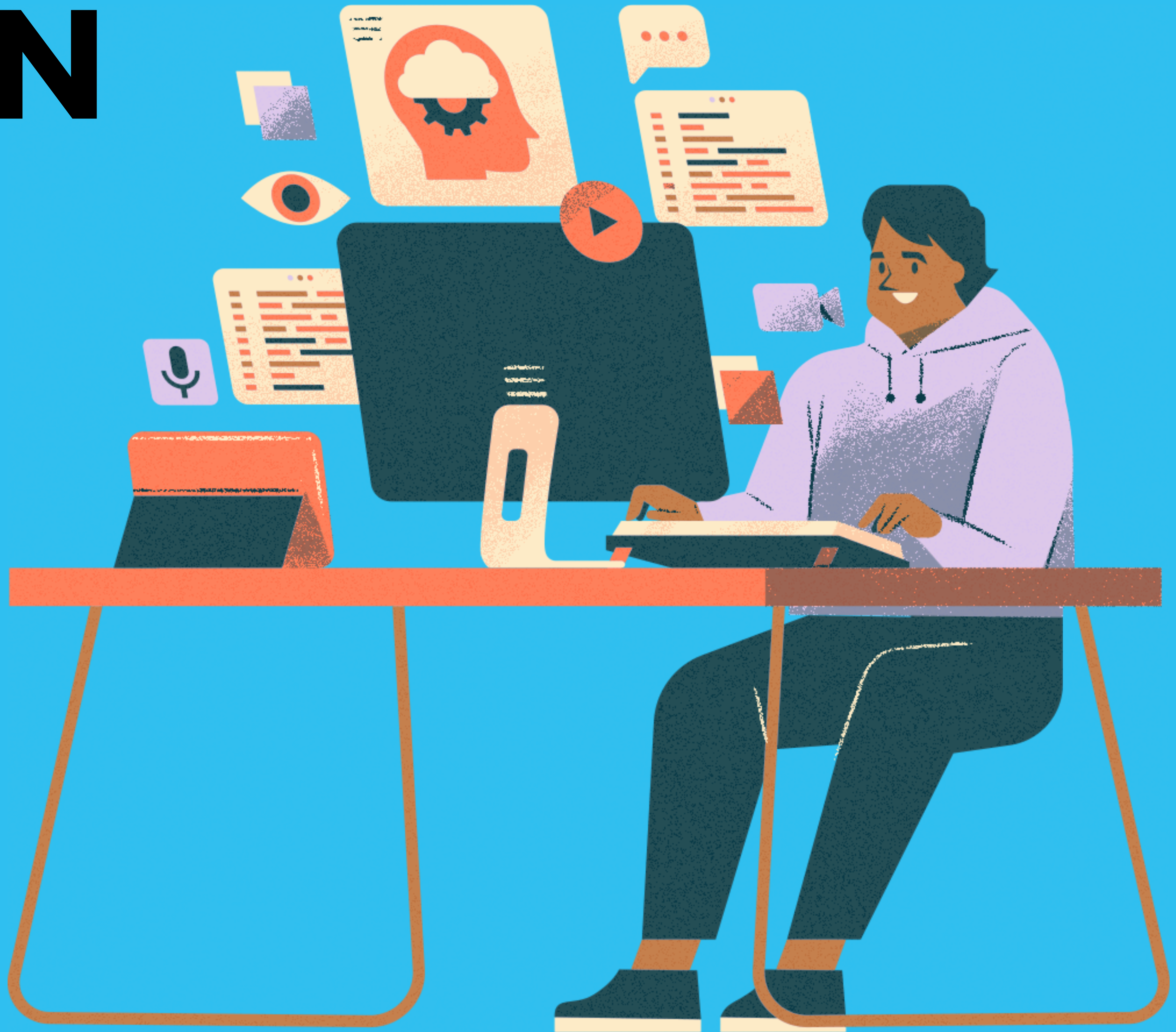
TOP 5 VARIABLES (BOTTOM 10 COUNTRIES)

4. ef_regulation: This variable encompasses the overall regulatory environment in the bottom 10 countries. A high score indicates a burdensome regulatory framework that may stifle entrepreneurship, investment, and economic growth. Excessive regulations can create barriers to entry, hamper innovation, and exacerbate inequalities, limiting opportunities and human freedom.

TOP 5 VARIABLES (BOTTOM 10 COUNTRIES)

5. ef_regulation_labor: This variable represents labor market regulations, including aspects such as minimum wage laws, hiring and firing regulations, and collective bargaining rights. Overly restrictive labor regulations can hinder job creation, labor mobility, and productivity growth. Flexible labor market policies that balance worker protections with market efficiency are essential for fostering inclusive economic growth and enhancing human freedom.

ALGORITHM OPTIMIZATION + MACHINE LEARNING



GRADIENT BOOSTING MACHINE (GBM)

GBM is a powerful boosting algorithm that **combines several weak learners into strong learners**, in which each new model is **trained to minimize the loss function** such as mean squared error or cross-entropy of the previous model using gradient descent.

In each iteration, the algorithm computes the **gradient of the loss function with respect to the predictions of the current ensemble** and then **trains a new weak model to minimize this gradient**. The predictions of the new model are then **added to the ensemble**, and the process is repeated until a **stopping criterion is met**.

SETTING UP GBM TO MINIMIZE ERROR

In [48]:
```python
years_until_2040 = np.arange(2021, 2041)
```

First, we initialize Gradient Boosting models with monitoring training performance.

In [58]:
```python
gbm_top = GradientBoostingRegressor(n_estimators=100, validation_fraction=0.2, n_iter_no_change=5, random_state=4
gbm_bottom = GradientBoostingRegressor(n_estimators=100, validation_fraction=0.2, n_iter_no_change=5, random_stat
```

Then, we shall split the data for top and bottom countries into train and test splits.

In [59]:
```python
X_train_top, X_test_top, y_train_top, y_test_top = train_test_split(X_top_10, y_top_10, test_size=0.2, random_sta
X_train_bottom, X_test_bottom, y_train_bottom, y_test_bottom = train_test_split(X_bottom_10, y_bottom_10, test_si
```

In [60]:
```python
gbm_top.fit(X_train_top, y_train_top)
gbm_bottom.fit(X_train_bottom, y_train_bottom)
```

Out[60]:
```
GradientBoostingRegressor(n_iter_no_change=5, random_state=42,
                          validation_fraction=0.2)
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

After training the model, we get the training errors for top and bottom countries.

In [62]:
```python
train_errors_top = np.zeros((gbm_top.n_estimators,), dtype=np.float64)
for i, y_pred in enumerate(gbm_top.staged_predict(X_train_top)):
    train_errors_top[i] = gbm_top.loss_(y_train_top, y_pred)

train_errors_bottom = np.zeros((gbm_bottom.n_estimators,), dtype=np.float64)
for i, y_pred in enumerate(gbm_bottom.staged_predict(X_train_bottom)):
    train_errors_bottom[i] = gbm_bottom.loss_(y_train_bottom, y_pred)
```

```
C:\Users\syedr\anaconda3\Lib\site-packages\sklearn\utils\deprecation.py:101: FutureWarning: Attribute `loss_` was d
eprecated in version 1.1 and will be removed in 1.3.
  warnings.warn(msg, category=FutureWarning)
C:\Users\syedr\anaconda3\Lib\site-packages\sklearn\utils\deprecation.py:101: FutureWarning: Attribute `loss_` was d
```

PREDICTING FUTURE VALUES

```
[66]: gbm_top = GradientBoostingRegressor(n_estimators=70, random_state=42)
      gbm_bottom = GradientBoostingRegressor(n_estimators=70, random_state=42)

      gbm_top.fit(X_top_10, y_top_10)
      gbm_bottom.fit(X_bottom_10, y_bottom_10)
```

```
[66]: ▾ GradientBoostingRegressor
      GradientBoostingRegressor(n_estimators=70, random_state=42)
```

```
[67]: predicted_scores_top = gbm_top.predict(X_top_10)
      predicted_scores_bottom = gbm_bottom.predict(X_bottom_10)

      predicted_scores_top = np.clip(predicted_scores_top, 0, 10)
      predicted_scores_bottom = np.clip(predicted_scores_bottom, 0, 10)
```

```
[68]: print("Predicted HF Scores until 2040 for Top 10 Countries:")
      for year, score in zip(years_until_2040.flatten(), predicted_scores_top):
          print(f"Year {year}: Predicted HF Score = {score}")

      print("\nPredicted HF Scores until 2040 for Bottom 10 Countries:")
      for year, score in zip(years_until_2040.flatten(), predicted_scores_bottom):
          print(f"Year {year}: Predicted HF Score = {score}")
```

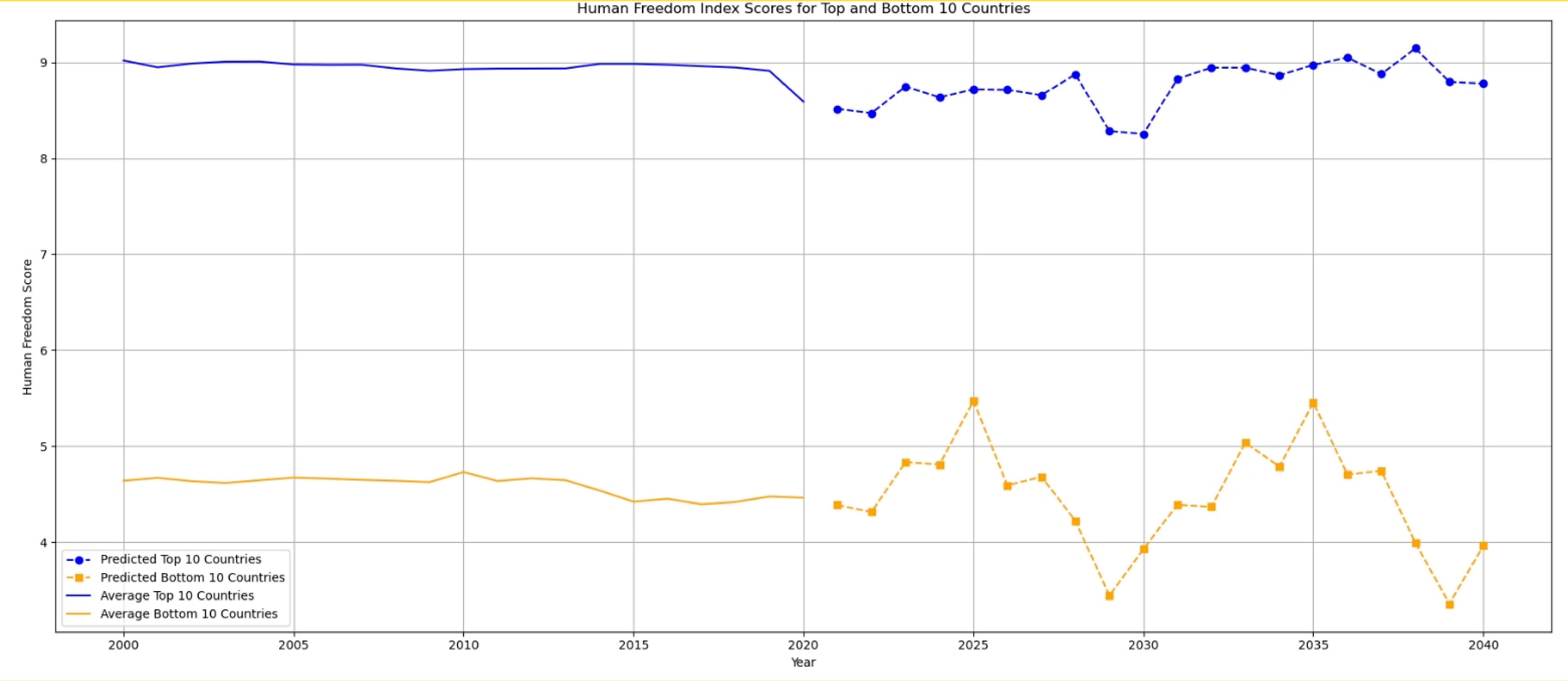
PREDICTING FUTURE VALUES

```
Predicted HF Scores until 2040 for Top 10 Countries:
Year 2021: Predicted HF Score = 8.5115278908145
Year 2022: Predicted HF Score = 8.470515960739213
Year 2023: Predicted HF Score = 8.734598990653959
Year 2024: Predicted HF Score = 8.628302251561117
Year 2025: Predicted HF Score = 8.705728981947612
Year 2026: Predicted HF Score = 8.747530992673047
Year 2027: Predicted HF Score = 8.661359413722465
Year 2028: Predicted HF Score = 8.95086677390937
Year 2029: Predicted HF Score = 8.297880059213849
Year 2030: Predicted HF Score = 8.231629845458594
Year 2031: Predicted HF Score = 8.842008442162394
Year 2032: Predicted HF Score = 8.862274074746788
Year 2033: Predicted HF Score = 9.011277152766427
Year 2034: Predicted HF Score = 8.87353130321527
Year 2035: Predicted HF Score = 8.942282259332346
Year 2036: Predicted HF Score = 9.050123161478313
Year 2037: Predicted HF Score = 8.853408608767952
Year 2038: Predicted HF Score = 9.168760027717378
Year 2039: Predicted HF Score = 8.803191049634117
Year 2040: Predicted HF Score = 8.753914270800516
```

```
Predicted HF Scores until 2040 for Bottom 10 Countries:
Year 2021: Predicted HF Score = 4.3253388197924805
Year 2022: Predicted HF Score = 4.2802875880512055
Year 2023: Predicted HF Score = 4.82776302789081
Year 2024: Predicted HF Score = 4.899911382528115
Year 2025: Predicted HF Score = 5.5276716753241
Year 2026: Predicted HF Score = 4.564711400630575
Year 2027: Predicted HF Score = 4.679176376203272
Year 2028: Predicted HF Score = 4.336038576981505
Year 2029: Predicted HF Score = 3.3309297959113118
Year 2030: Predicted HF Score = 3.9622105285539937
Year 2031: Predicted HF Score = 4.350944073291549
Year 2032: Predicted HF Score = 4.272308033954935
Year 2033: Predicted HF Score = 5.0875242324540295
Year 2034: Predicted HF Score = 4.771779782000183
Year 2035: Predicted HF Score = 5.403170208353439
Year 2036: Predicted HF Score = 4.722452937614061
Year 2037: Predicted HF Score = 4.701454618932462
Year 2038: Predicted HF Score = 4.058694134161795
Year 2039: Predicted HF Score = 3.388098056522922
Year 2040: Predicted HF Score = 3.9667208010261095
```

INFORMATION PRESENTATION + STATISTICAL INFERENCE

PREDICTING FUTURE VALUES

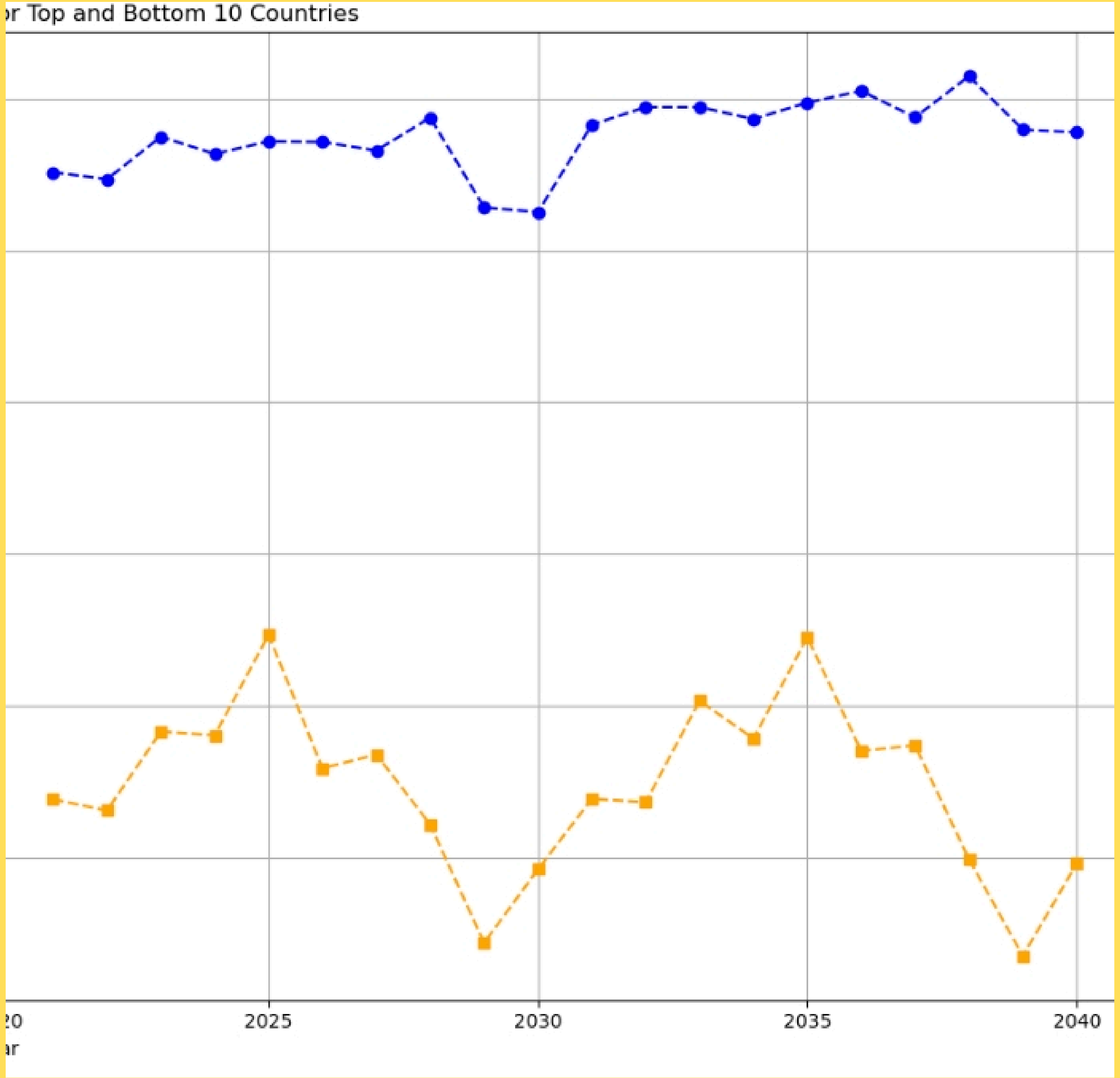


INFERENCE

Countries at the bottom can **learn from the those at the top**. These variables underscore the **critical role of governance, economic policies, and legal frameworks in shaping human freedom**.

In the top 10 countries, policies that prioritize **economic freedom, regulatory efficiency, and civil liberties** contribute to robust socio-economic outcomes and high levels of human freedom.

Conversely, the bottom 10 countries **face challenges related to governance effectiveness, regulatory burdens, and political rights**, which may impede progress toward greater human freedom.



GRADIENT BOOSTING MACHINE

PROS:

- High Predictive Accuracy
- Handles Different Types of Data
- Feature Importance
- Robust to Overfitting
- Handles Missing Data
- Flexibility

CONS:

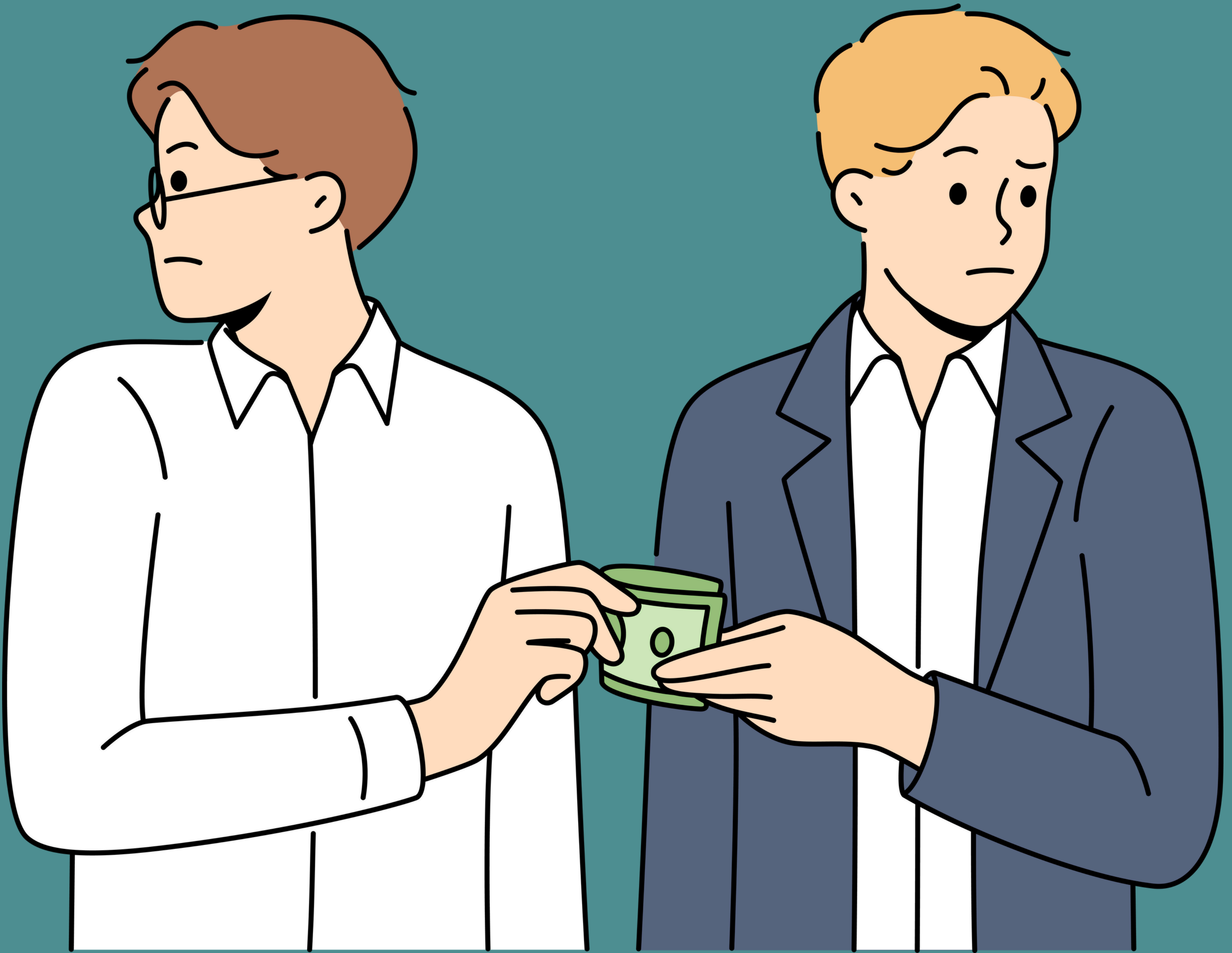
- Computationally Intensive
- Sensitive to Hyperparameters
- Potential for Overfitting
- Black Box Model

GRADIENT BOOSTING MACHINE

Why GBM for our Model:

- **High Predictive Accuracy:** Given the complexity of the relationship between government effectiveness and human freedom, we need a model capable of capturing nonlinear and intricate patterns in the data. GBM's ability to build complex models makes it well-suited for this task.
- **Feature Importance Analysis:** Understanding which factors contribute most to human freedom is crucial for our project.
- **Handles Missing Data:** Since real-world datasets often contain missing values, GBM's ability to handle missing data without preprocessing simplifies the data preparation process, allowing us to focus more on analyzing the relationship between government effectiveness and human freedom.
- **Flexibility:** Whether it's regression or classification analysis, GBM can accommodate both types of tasks. This flexibility allows us to explore different aspects of the relationship between government effectiveness and human freedom using the same modeling framework.

ETHICAL CONSIDERATION + INTELLIGENT DECISION



WHAT CAN BE DONE TO IMPROVE LEVELS OF HUMAN FREEDOM ?

Promoting Economic Freedom: Enhancing economic freedom by reducing regulatory barriers, fostering competition, and safeguarding property rights can stimulate innovation, entrepreneurship, and prosperity, thereby expanding opportunities and human freedom.

WHAT CAN BE DONE TO IMPROVE LEVELS OF HUMAN FREEDOM ?

Strengthening Democratic Institutions: Investing in transparent and accountable governance structures, electoral processes, and rule of law mechanisms can bolster political freedoms, civic engagement, and democratic resilience, fostering inclusive and participatory societies.

WHAT CAN BE DONE TO IMPROVE LEVELS OF HUMAN FREEDOM ?

Advancing Social Justice: Implementing equitable policies that address poverty, inequality, and social exclusion can enhance human dignity, social cohesion, and equal access to opportunities, promoting a more just and inclusive society.

WHAT CAN BE DONE TO IMPROVE LEVELS OF HUMAN FREEDOM ?

Protecting Civil Liberties: Safeguarding fundamental rights such as freedom of expression, assembly, and association is crucial for nurturing democratic culture, pluralism, and individual autonomy, safeguarding human freedom against encroachments by authoritarian regimes or societal prejudices.

SC1015 TEAM 3

WITH THAT, WE HAVE COME TO THE END.
THANK YOU!

Created by:
Syed Ridwan Kabir (U2321342C)
Evelyn Theresia Cuaca (U2320523L)