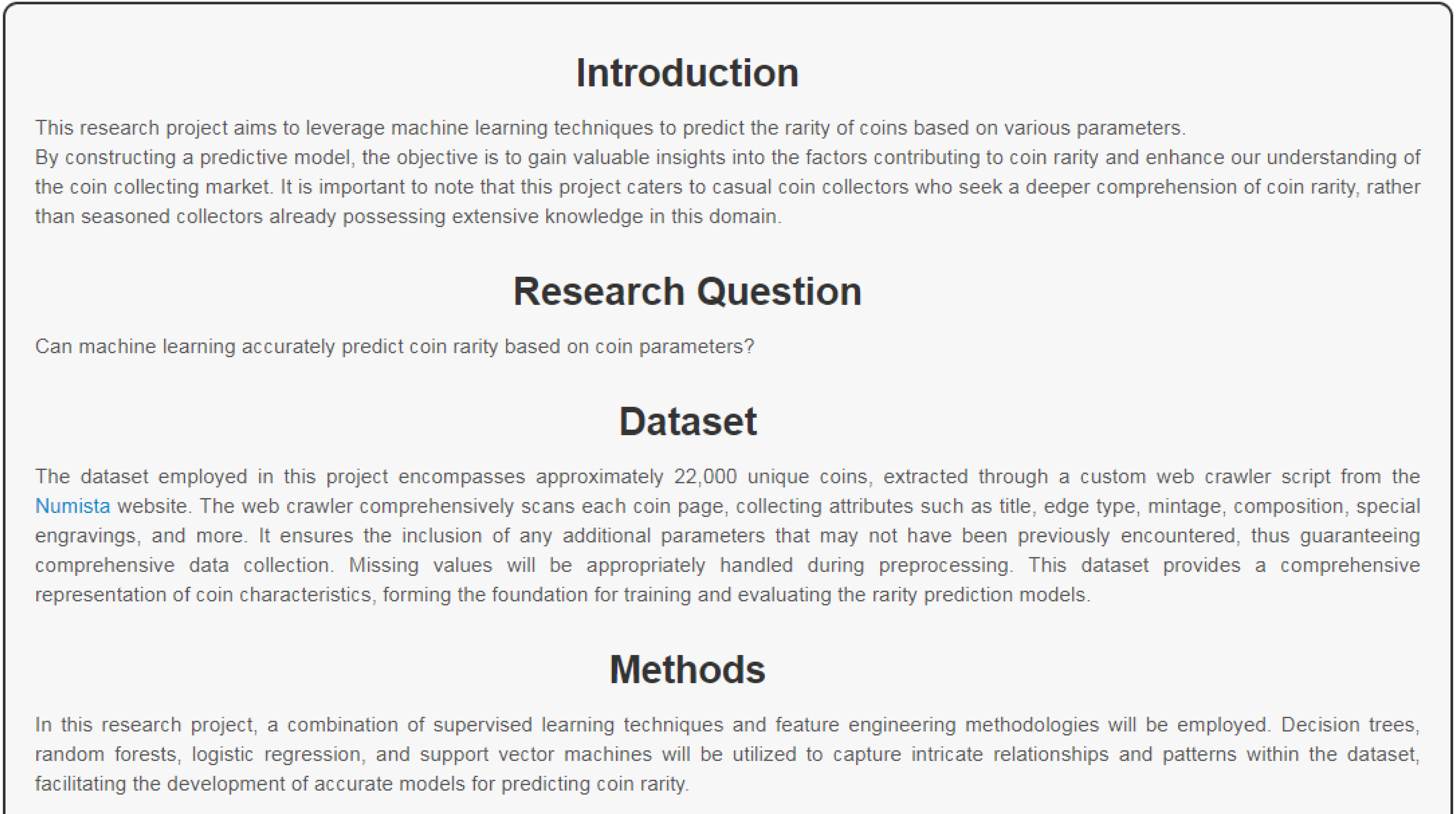
# PREDICTING COIN RARITY THROUGH DATA-DRIVEN MACHINE LEARNING

**By Alexey Kovalev**
Holon Institute of Technology | HIT
2023

Data Source: numista.com

HIT 50 YEARS OF EXCELLENCE
מכון טכנולוגי חולון
Holon Institute of Technology



# Introduction

This research project aims to leverage machine learning techniques to predict the rarity of coins based on various parameters.
By constructing a predictive model, the objective is to gain valuable insights into the factors contributing to coin rarity and enhance our understanding of the coin collecting market. It is important to note that this project caters to casual coin collectors who seek a deeper comprehension of coin rarity, rather than seasoned collectors already possessing extensive knowledge in this domain.

# Research Question

Can machine learning accurately predict coin rarity based on coin parameters?

# Dataset

The dataset employed in this project encompasses approximately 22,000 unique coins, extracted through a custom web crawler script from the Numista website. The web crawler comprehensively scans each coin page, collecting attributes such as title, edge type, mintage, composition, special engravings, and more. It ensures the inclusion of any additional parameters that may not have been previously encountered, thus guaranteeing comprehensive data collection. Missing values will be appropriately handled during preprocessing. This dataset provides a comprehensive representation of coin characteristics, forming the foundation for training and evaluating the rarity prediction models.

# Methods

In this research project, a combination of supervised learning techniques and feature engineering methodologies will be employed. Decision trees, random forests, logistic regression, and support vector machines will be utilized to capture intricate relationships and patterns within the dataset, facilitating the development of accurate models for predicting coin rarity.

# DATA COLLECTING

## Gameplan

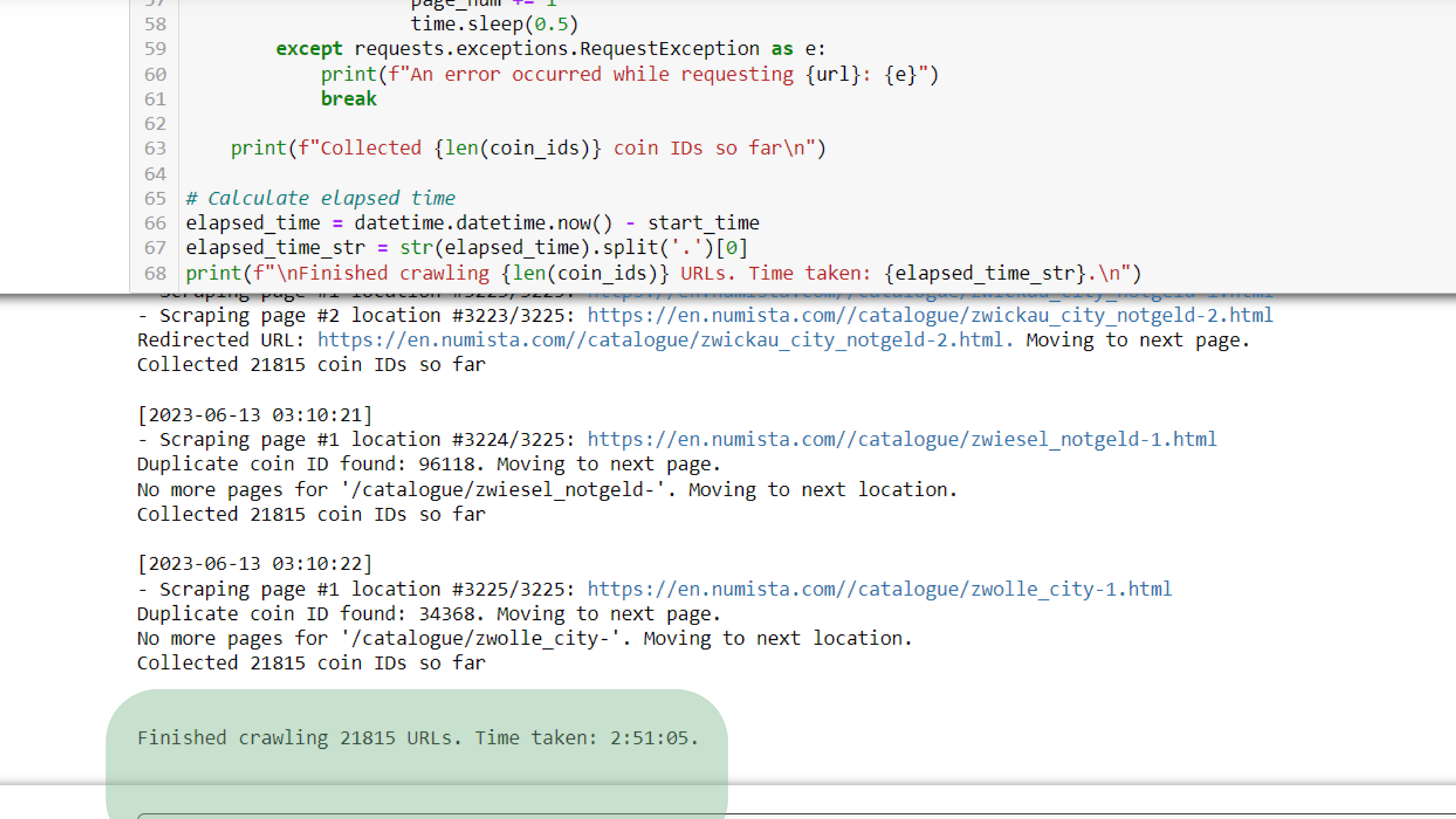
- **Acquiring the Proper HTML Responses:** Obtain the necessary HTML responses from the Numista website.
- **Investigating Website Structure:** Carefully analyze the structure of the website and identify the specific elements that contain the desired data.
- **Setting up an Efficient Web Crawler:** Establish a sophisticated and highly efficient web crawler to systematically gather the required information.
- **Handling Bugs and Roadblocks:** Address any unexpected issues and overcome obstacles that may arise during the crawling process.
- **Executing the Web Crawler:** Execute the web crawler to diligently collect the necessary data.

## Goal

- **Achieving a Comprehensive Dataset:** Attain a dataset of considerable size, enriched with an abundant amount of data, to serve as the foundation for training our remarkable machine learning models.

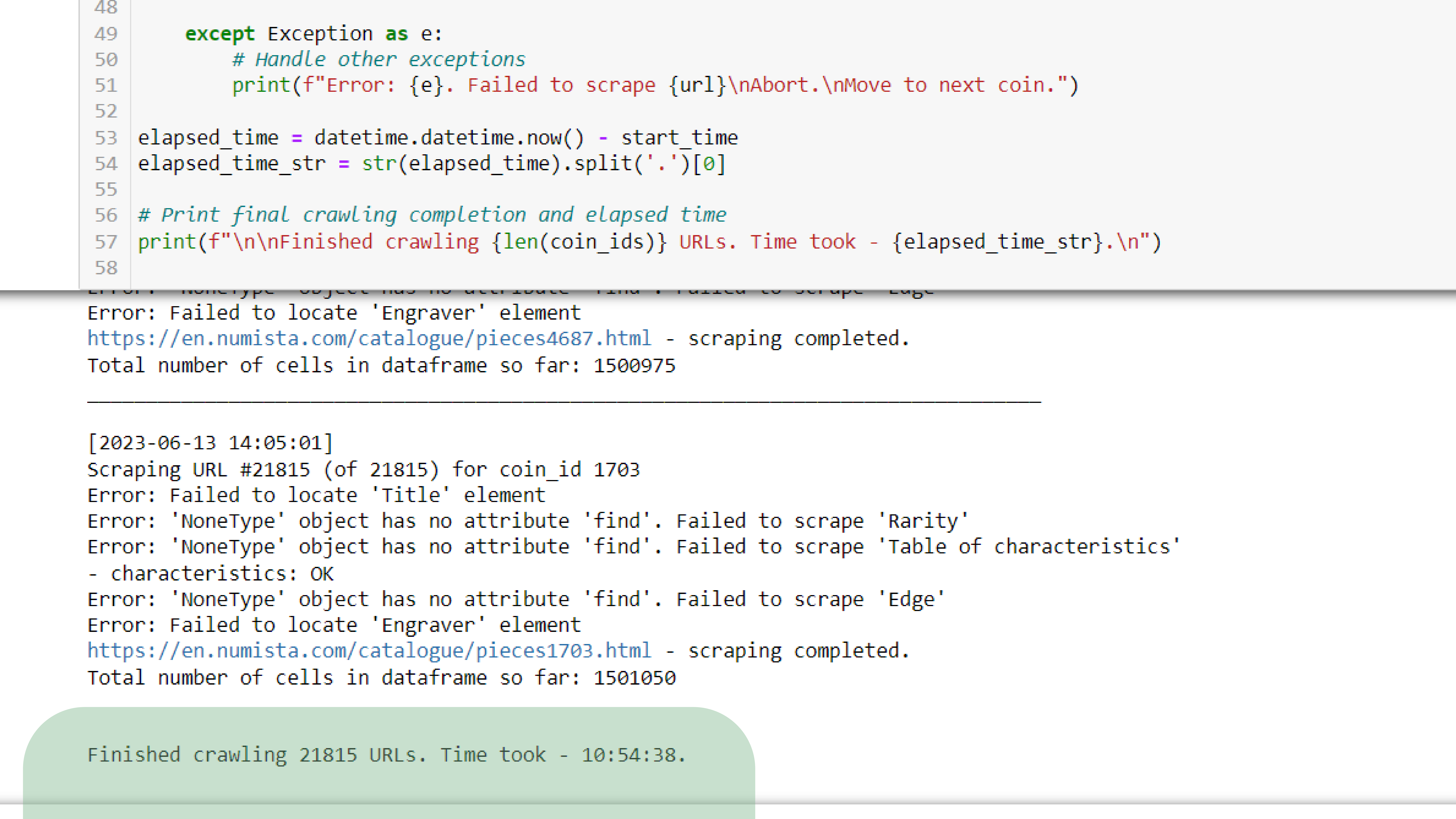
In [1]:

```
import requests
import pandas as pd
pd.set_option('display.float_format', '{:.2f}'.format)
import numpy as np
from IPython.display import display, Code
import time
import datetime
```



```
                    page_num += 1
                    time.sleep(0.5)
        except requests.exceptions.RequestException as e:
            print(f"An error occurred while requesting {url}: {e}")
            break

    print(f"Collected {len(coin_ids)} coin IDs so far\n")

# Calculate elapsed time
elapsed_time = datetime.datetime.now() - start_time
elapsed_time_str = str(elapsed_time).split('.')[0]
print(f"\nFinished crawling {len(coin_ids)} URLs. Time taken: {elapsed_time_str}.\n")
```

```
- Scraping page #1 location #3223/3225: https://en.numista.com//catalogue/zwickau_city_notgeld-1.html
- Scraping page #2 location #3223/3225: https://en.numista.com//catalogue/zwickau_city_notgeld-2.html
Redirected URL: https://en.numista.com//catalogue/zwickau_city_notgeld-2.html. Moving to next page.
Collected 21815 coin IDs so far

[2023-06-13 03:10:21]
- Scraping page #1 location #3224/3225: https://en.numista.com//catalogue/zwiesel_notgeld-1.html
Duplicate coin ID found: 96118. Moving to next page.
No more pages for '/catalogue/zwiesel_notgeld-'. Moving to next location.
Collected 21815 coin IDs so far

[2023-06-13 03:10:22]
- Scraping page #1 location #3225/3225: https://en.numista.com//catalogue/zwolle_city-1.html
Duplicate coin ID found: 34368. Moving to next page.
No more pages for '/catalogue/zwolle_city-'. Moving to next location.
Collected 21815 coin IDs so far


Finished crawling 21815 URLs. Time taken: 2:51:05.
```



```python
    except Exception as e:
        # Handle other exceptions
        print(f"Error: {e}. Failed to scrape {url}\nAbort.\nMove to next coin.")

elapsed_time = datetime.datetime.now() - start_time
elapsed_time_str = str(elapsed_time).split('.')[0]

# Print final crawling completion and elapsed time
print(f"\n\nFinished crawling {len(coin_ids)} URLs. Time took - {elapsed_time_str}.\n")

```

```
Error: 'NoneType' object has no attribute 'find'. Failed to scrape 'Edge'
Error: Failed to locate 'Engraver' element
https://en.numista.com/catalogue/pieces4687.html - scraping completed.
Total number of cells in dataframe so far: 1500975

________________________________________________________________________________

[2023-06-13 14:05:01]
Scraping URL #21815 (of 21815) for coin_id 1703
Error: Failed to locate 'Title' element
Error: 'NoneType' object has no attribute 'find'. Failed to scrape 'Rarity'
Error: 'NoneType' object has no attribute 'find'. Failed to scrape 'Table of characteristics'
- characteristics: OK
Error: 'NoneType' object has no attribute 'find'. Failed to scrape 'Edge'
Error: Failed to locate 'Engraver' element
https://en.numista.com/catalogue/pieces1703.html - scraping completed.
Total number of cells in dataframe so far: 1501050


Finished crawling 21815 URLs. Time took - 10:54:38.
```

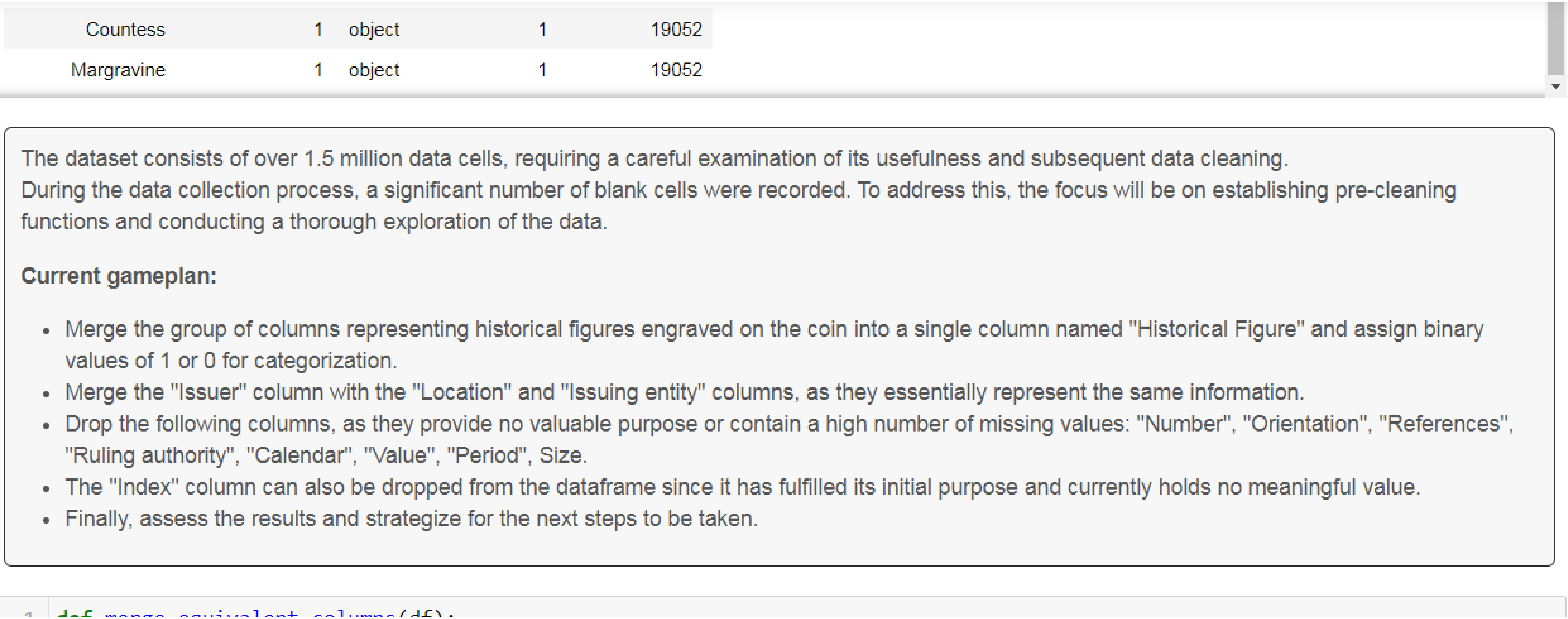
# DATA PREPROCESSING & EXPLORATORY DATA ANALYSIS

## Gameplan

- **Explore the raw dataframe:** Begin by examining the size, shape, and distribution of the data. Conduct exploratory data analysis (EDA) to gain insights into the dataset, identify patterns, and understand the data's characteristics. Look for potential issues such as missing values, duplicates, or incorrect data types that need to be addressed.
- **Clean the dataframe:** Remove any unnecessary rows or columns from the dataframe. Handle missing values using appropriate methods, such as imputation or removal.
- **Merge similar columns:** Identify columns that serve the same purpose and consolidate them into a single column. This simplifies the dataframe and improves analysis efficiency.
- **Clean values:** Apply necessary transformations to the data, such as removing special characters, converting strings to numeric values, or adjusting formatting.
- **Feature Engineering:** Create new features or modify existing ones to enhance the dataset's suitability for machine learning training or analysis. Categorize values of parameters to adjust the dimensionality of the dataframe and represent variables more effectively.

## Goal

- **Obtain a clean and prepared dataset:** Produce a dataset that is free of errors, inconsistencies, and unnecessary elements. The resulting dataset should be well-prepared for subsequent machine learning training, analysis, or other tasks.



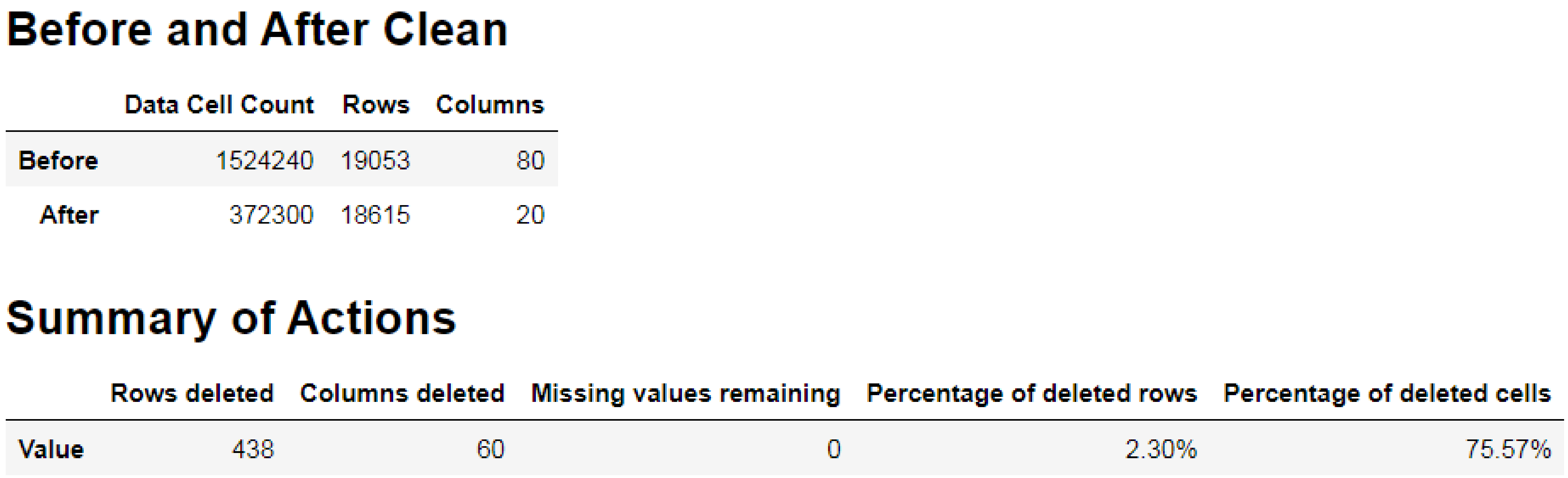
| | | | | |
|---|---|---|---|---|
| Countess | 1 | object | 1 | 19052 |
| Margravine | 1 | object | 1 | 19052 |

The dataset consists of over 1.5 million data cells, requiring a careful examination of its usefulness and subsequent data cleaning.
During the data collection process, a significant number of blank cells were recorded. To address this, the focus will be on establishing pre-cleaning functions and conducting a thorough exploration of the data.

**Current gameplan:**

- Merge the group of columns representing historical figures engraved on the coin into a single column named "Historical Figure" and assign binary values of 1 or 0 for categorization.
- Merge the "Issuer" column with the "Location" and "Issuing entity" columns, as they essentially represent the same information.
- Drop the following columns, as they provide no valuable purpose or contain a high number of missing values: "Number", "Orientation", "References", "Ruling authority", "Calendar", "Value", "Period", Size.
- The "Index" column can also be dropped from the dataframe since it has fulfilled its initial purpose and currently holds no meaningful value.
- Finally, assess the results and strategize for the next steps to be taken.

```
def merge_equivalent_columns(df):
```



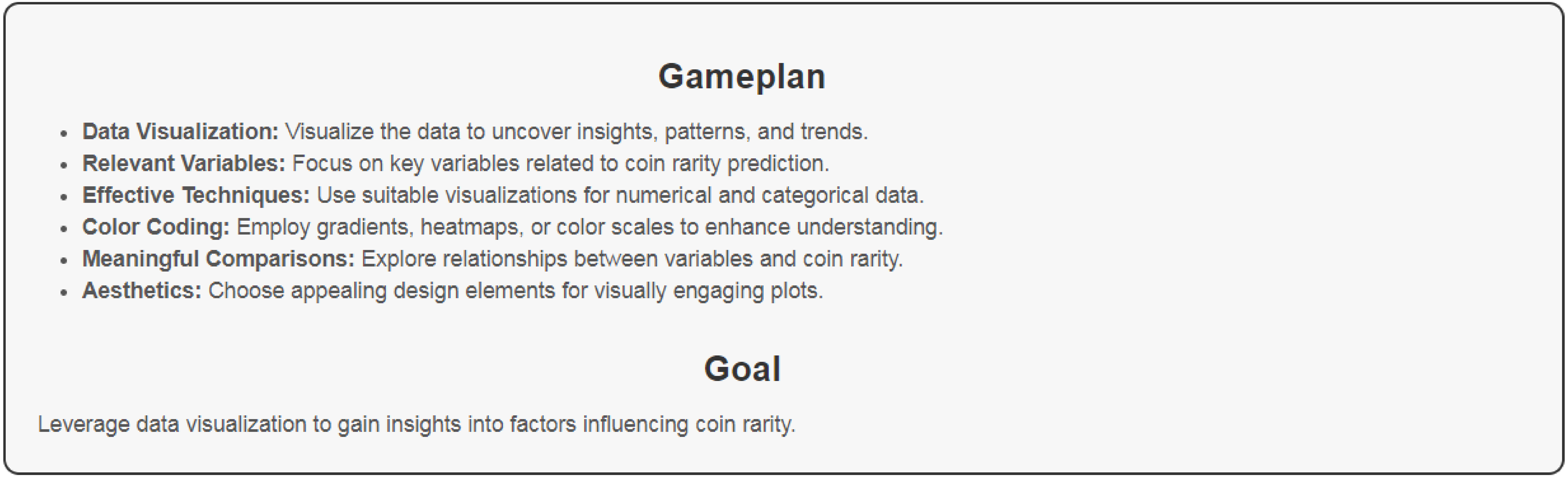
## Before and After Clean

| | Data Cell Count | Rows | Columns |
|---|---|---|---|
| **Before** | 1524240 | 19053 | 80 |
| **After** | 372300 | 18615 | 20 |

## Summary of Actions

| | Rows deleted | Columns deleted | Missing values remaining | Percentage of deleted rows | Percentage of deleted cells |
|---|---|---|---|---|---|
| **Value** | 438 | 60 | 0 | 2.30% | 75.57% |

# DATA VISUALIZATION

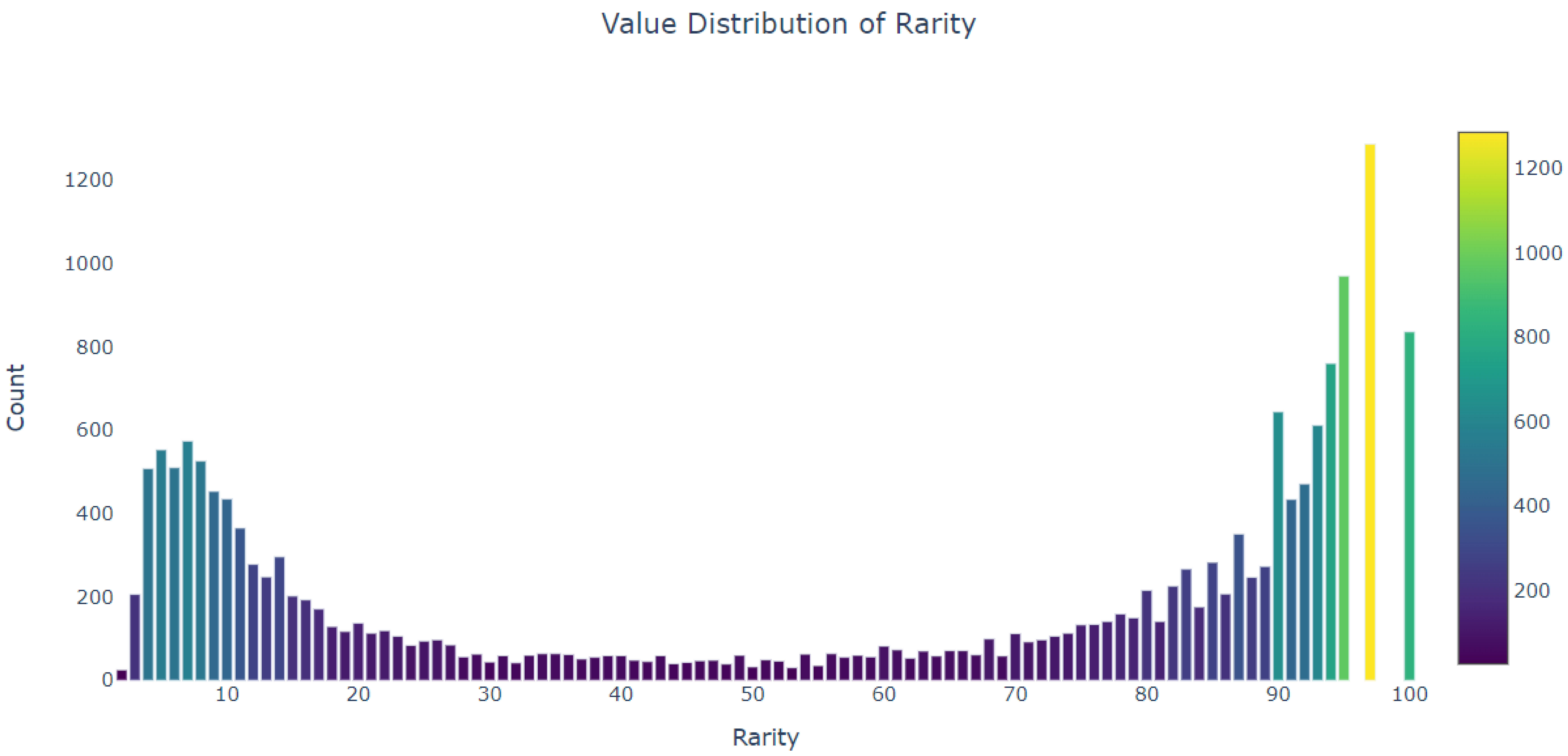


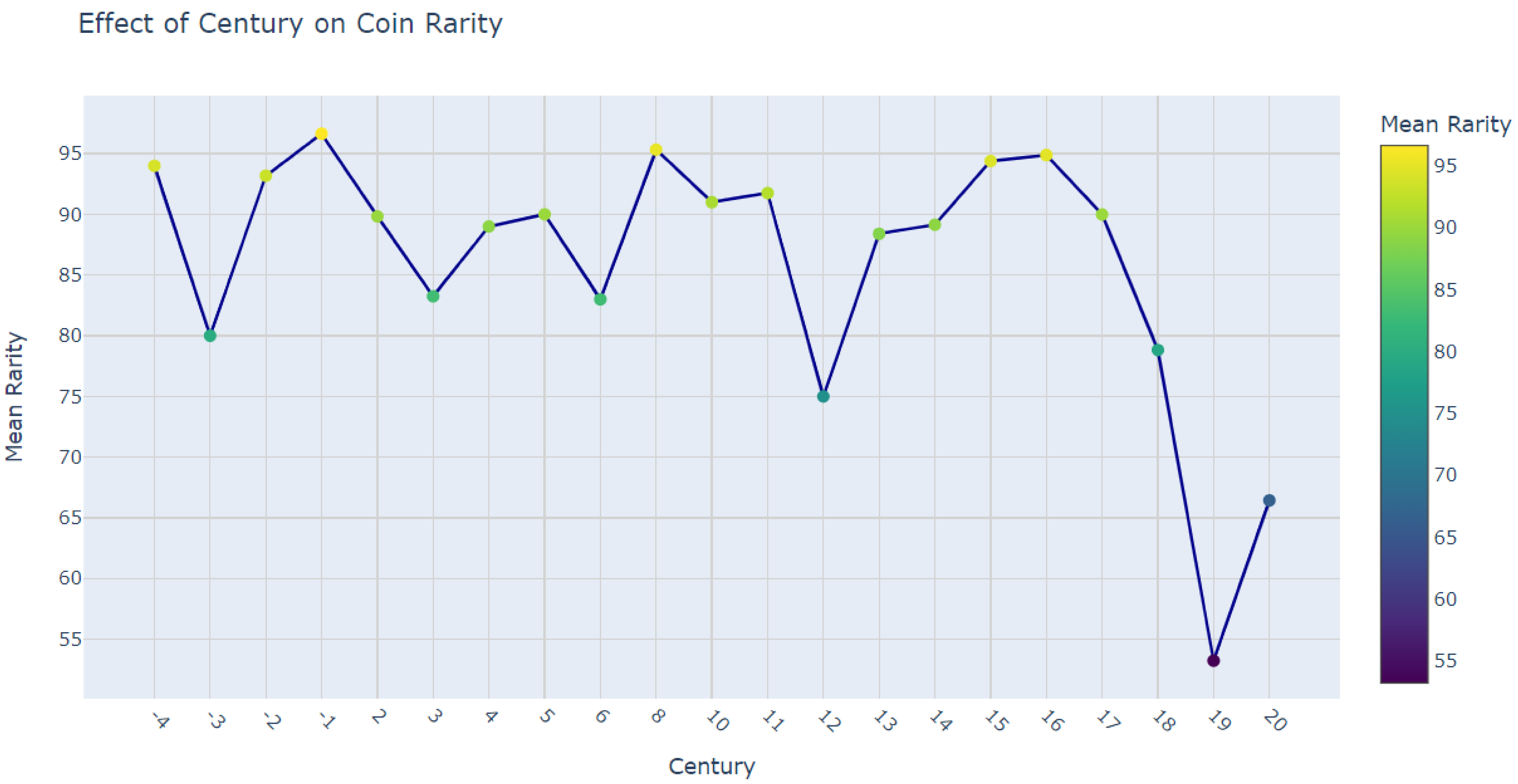
## Gameplan

- **Data Visualization:** Visualize the data to uncover insights, patterns, and trends.
- **Relevant Variables:** Focus on key variables related to coin rarity prediction.
- **Effective Techniques:** Use suitable visualizations for numerical and categorical data.
- **Color Coding:** Employ gradients, heatmaps, or color scales to enhance understanding.
- **Meaningful Comparisons:** Explore relationships between variables and coin rarity.
- **Aesthetics:** Choose appealing design elements for visually engaging plots.

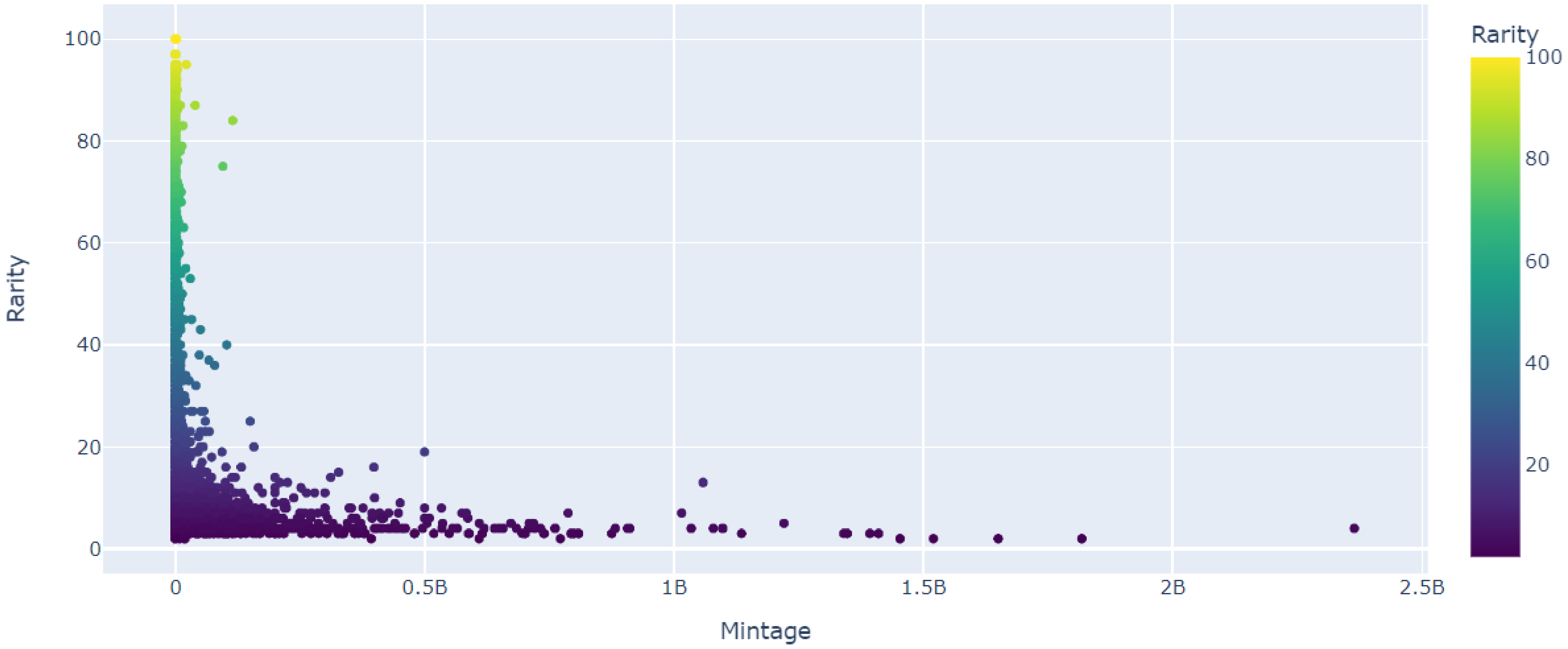
## Goal

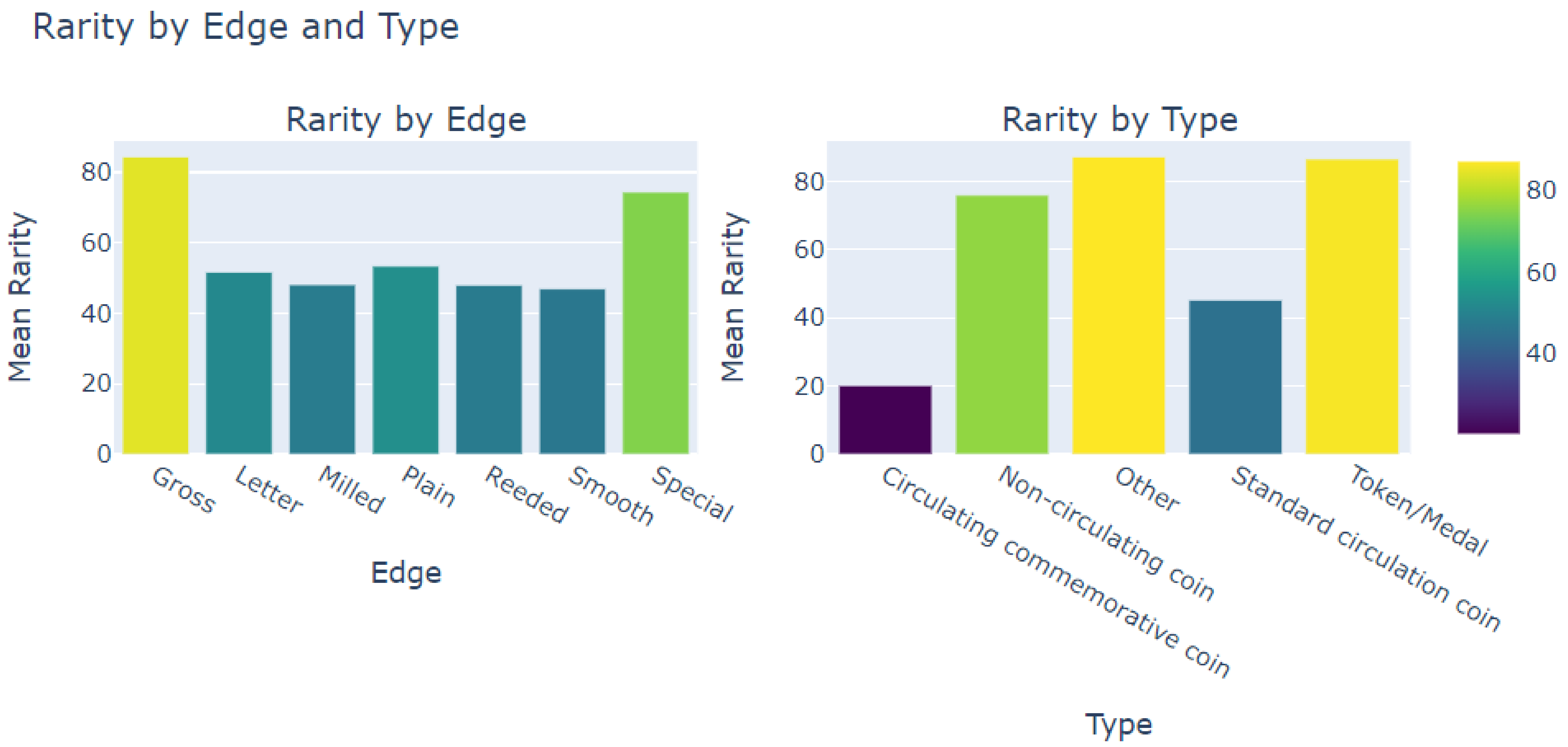
Leverage data visualization to gain insights into factors influencing coin rarity.

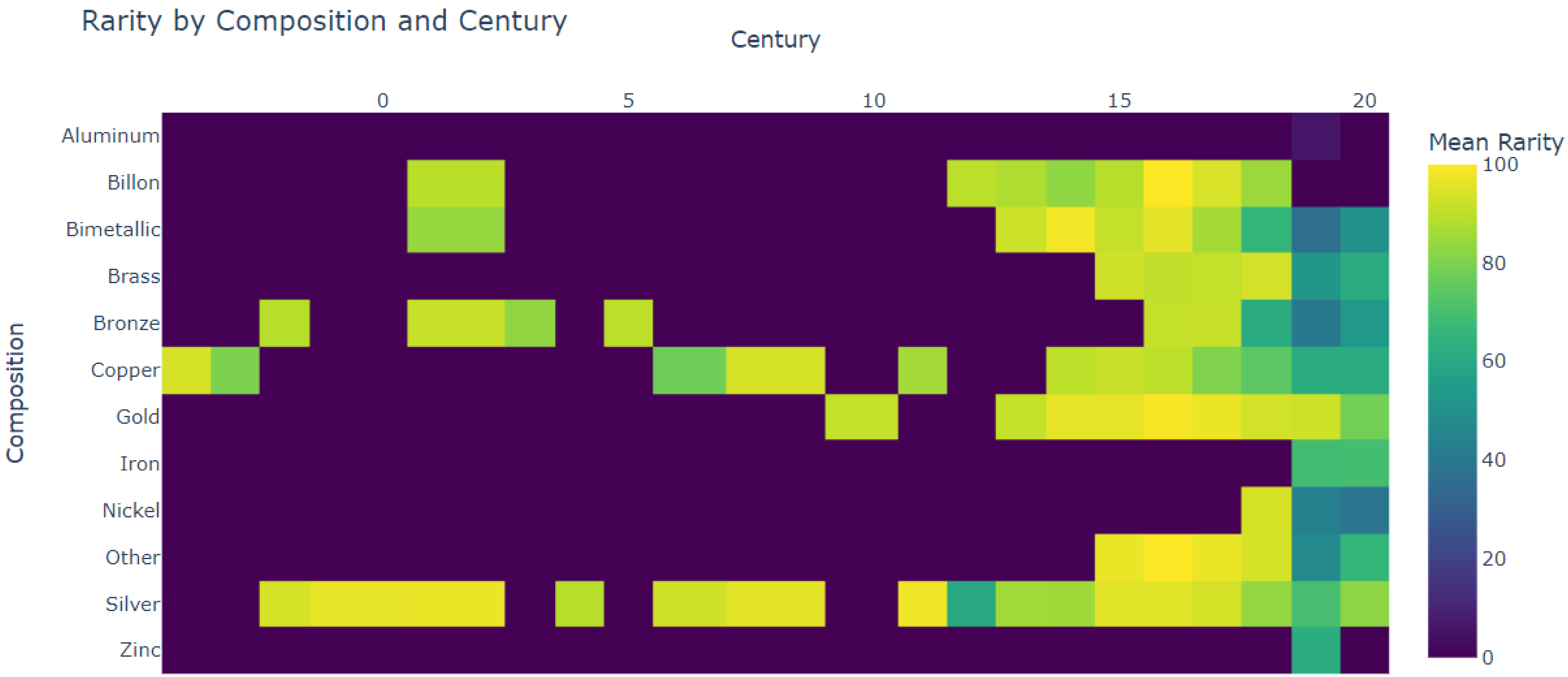

Value Distribution of Rarity
Count
0
200
400
600
800
1000
1200
10
20
30
40
50
60
70
80
90
100
Rarity
200
400
600
800
1000
1200

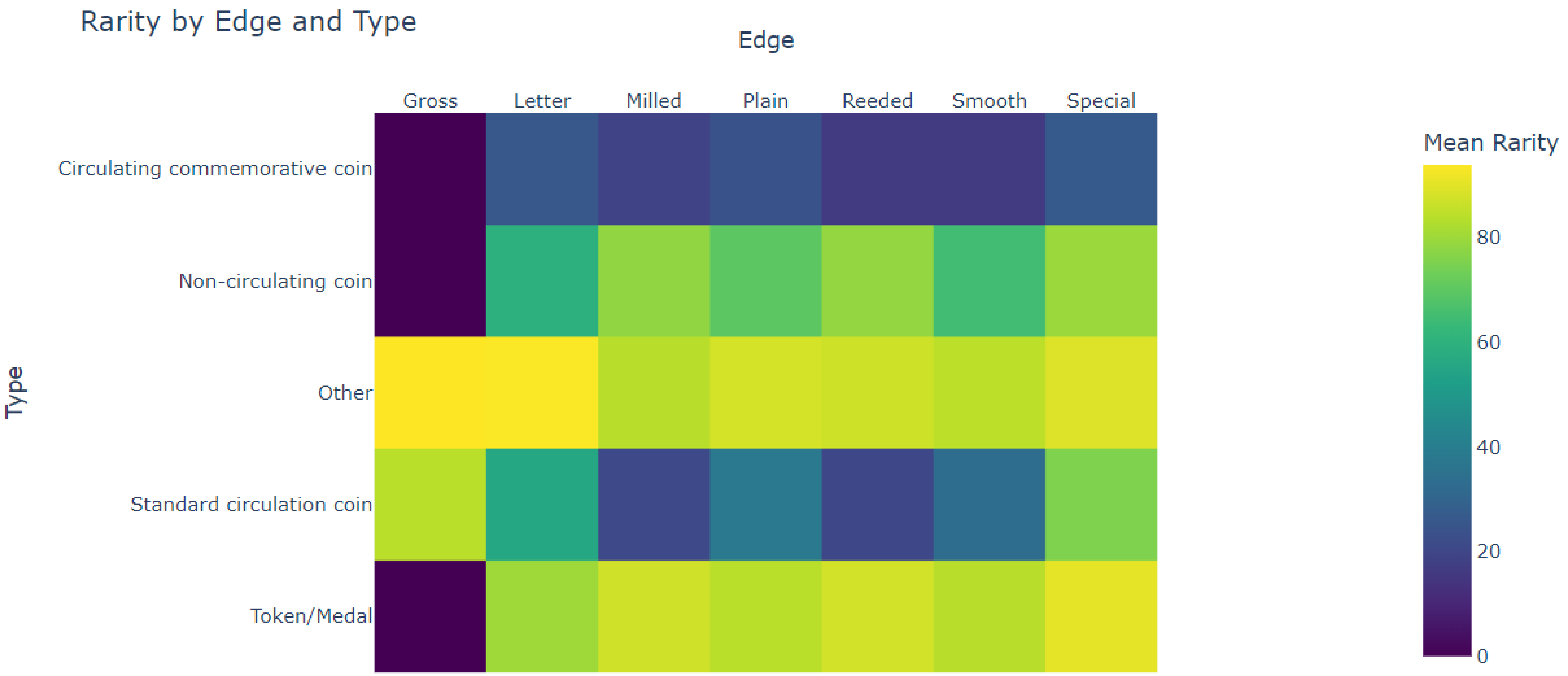

Effect of Century on Coin Rarity
Mean Rarity
95
90
85
80
75
70
65
60
55
Century
-4
-3
-2
-1
2
3
4
5
6
8
10
11
12
13
14
15
16
17
18
19
20
Mean Rarity
95
90
85
80
75
70
65
60
55

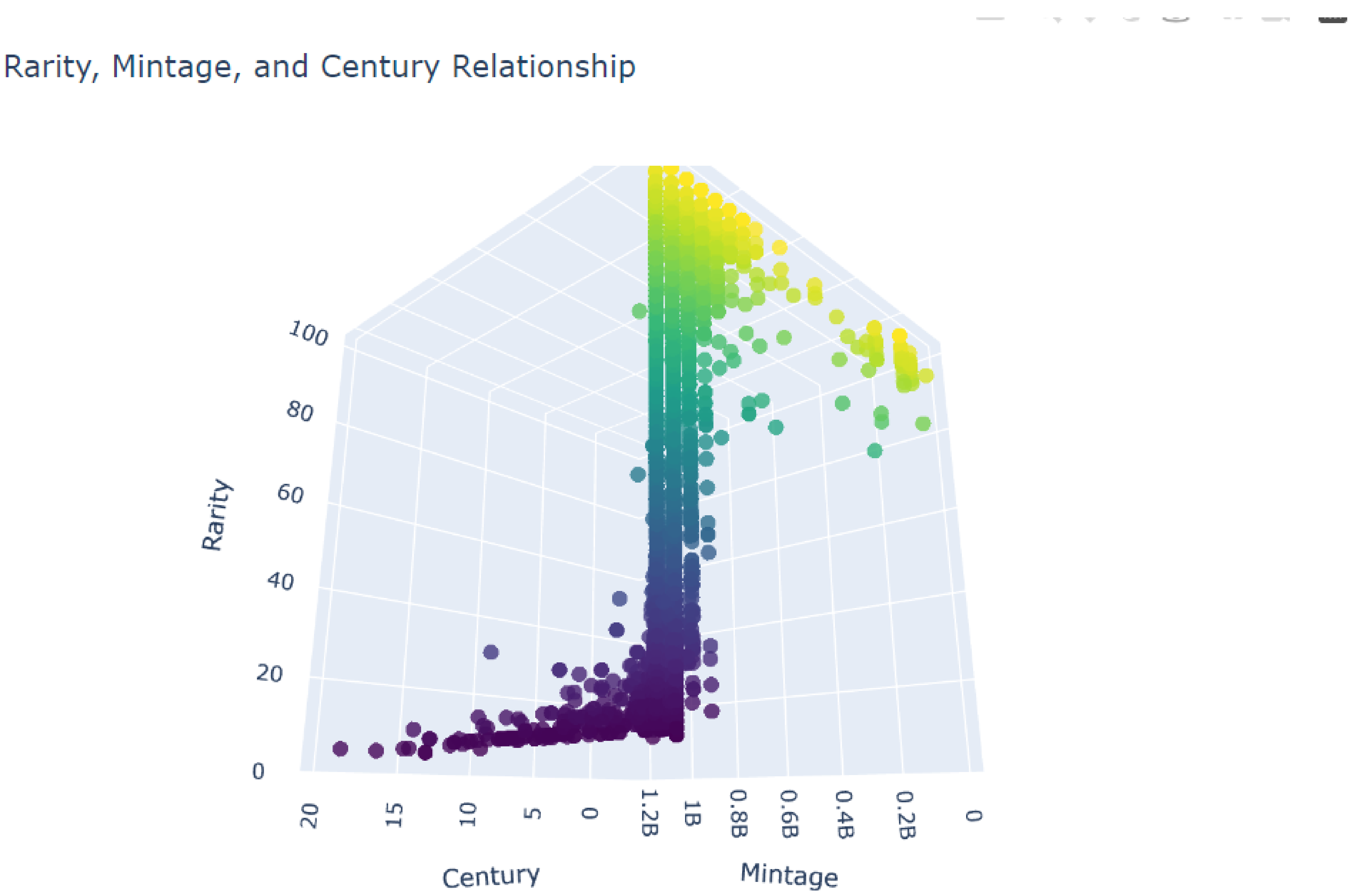
Rarity vs. Mintage




Rarity by Edge and Type
Rarity by Edge
Mean Rarity
0
20
40
60
80
Gross
Letter
Milled
Plain
Reeded
Smooth
Special
Edge
Rarity by Type
Mean Rarity
0
20
40
60
80
Circulating commemorative coin
Non-circulating coin
Other
Standard circulation coin
Token/Medal
Type
40
60
80


Rarity by Composition and Century
Century
0
5
10
15
20
Aluminum
Billon
Bimetallic
Brass
Bronze
Copper
Gold
Iron
Nickel
Other
Silver
Zinc
Composition
Mean Rarity
100
80
60
40
20
0


Rarity by Edge and Type
Edge
Gross
Letter
Milled
Plain
Reeded
Smooth
Special
Type
Circulating commemorative coin
Non-circulating coin
Other
Standard circulation coin
Token/Medal
Mean Rarity
80
60
40
20
0


Rarity, Mintage, and Century Relationship
Rarity
100
80
60
40
20
0
20
15
10
5
0
Century
1.2B
1B
0.8B
0.6B
0.4B
0.2B
0
Mintage

# FINAL FEATURE ENGINEERING

## Gameplan

- **Exploratory Data Analysis:** Understand the data and its characteristics before model selection.
- **Feature Engineering:** Transform and create meaningful features to enhance model performance.
- **Model Selection:** Explore a variety of algorithms and techniques to identify the most suitable model.
- **Hyperparameter Tuning:** Optimize model performance by tuning hyperparameters through techniques like grid search or randomized search.
- **Training and Validation:** Split the data into training and validation sets for model training and evaluation.
- **Model Training:** Train the selected model on the training set using appropriate algorithms.

## Goal

Utilize machine learning techniques to select and train the most suitable model for predicting coin rarity.

```
Feature Engineering Debug Report:
----------------------------------------
✓ The 'Title' column is successfully dropped.
✓ The 'Thickness' column is successfully dropped.
✓ The 'Century' column is successfully dropped.
✓ The 'Engraver' column is successfully converted to uint8.
✓ The '-1' values in the 'Mintage' column are successfully replaced with NaN.
✓ 1 is successfully added to every value in the 'Times Issued' column.
✓ The 'Mintage_Missing' column is successfully created.
✓ Outliers in the 'Diameter' column are successfully handled.
✓ The 'Diameter' column is successfully scaled using MinMaxScaler.
✓ The 'Rarity' column is successfully scaled using MinMaxScaler.
✓ One-hot encoding is successfully applied to the 'Edge' column.
✓ One-hot encoding is successfully applied to the 'Composition' column.
✓ One-hot encoding is successfully applied to the 'Demonetized' column.
✓ One-hot encoding is successfully applied to the 'Orientation' column.
✓ One-hot encoding is successfully applied to the 'Shape' column.
✓ One-hot encoding is successfully applied to the 'Type' column.
✓ One-hot encoding is successfully applied to the 'Technique' column.
✓ The 'Issuer' column is successfully label encoded.
✓ The 'Currency' column is successfully label encoded.

Feature Engineering Debug Report Complete.
```

```
In [88]: print_info_summary(df_ready)
```

**Dataframe Summary:**

```
Number of Rows:  18614
Number of Columns:  49
Number of Data Cells:  912086
```

# MACHINE LEARNING

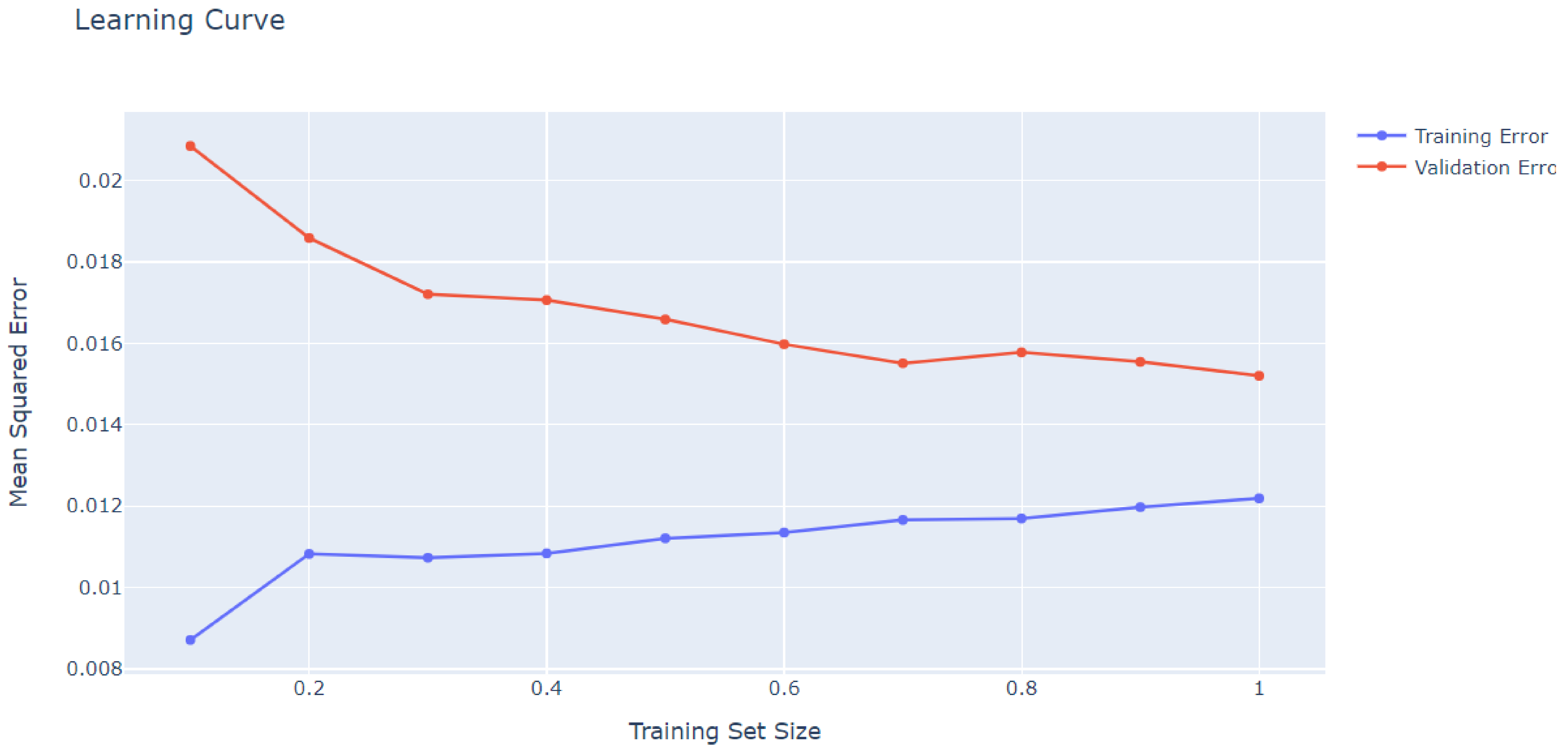
## Gameplan

- **Feature Selection:** Relevant features are selected using mutual information, which measures the dependence between features and the target variable.
- **Model Selection:** Several regression models, including Linear Regression, Decision Tree, Random Forest, and Support Vector Machine, are considered for predicting coin rarity.
- **Hyperparameter Tuning:** The models are fine-tuned using grid search and cross-validation to find the optimal combination of hyperparameters for improved performance.
- **Ensemble Learning:** An ensemble model is created using the voting regressor technique, which combines the predictions of multiple models to enhance prediction accuracy.
- **Model Evaluation:** The performance of each individual model and the ensemble model is evaluated using metrics such as mean squared error, mean absolute error, and R-squared score.

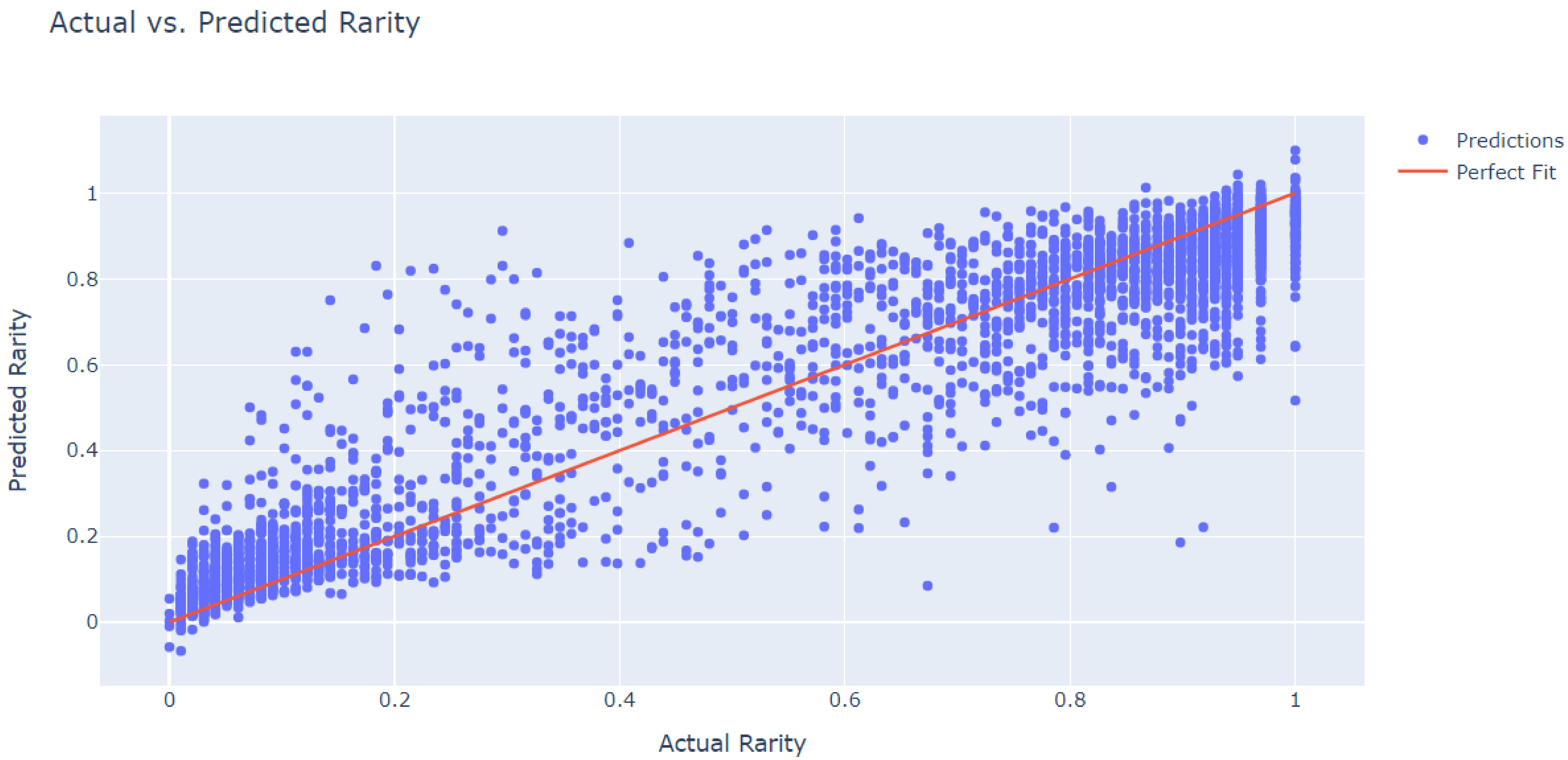
## Goal

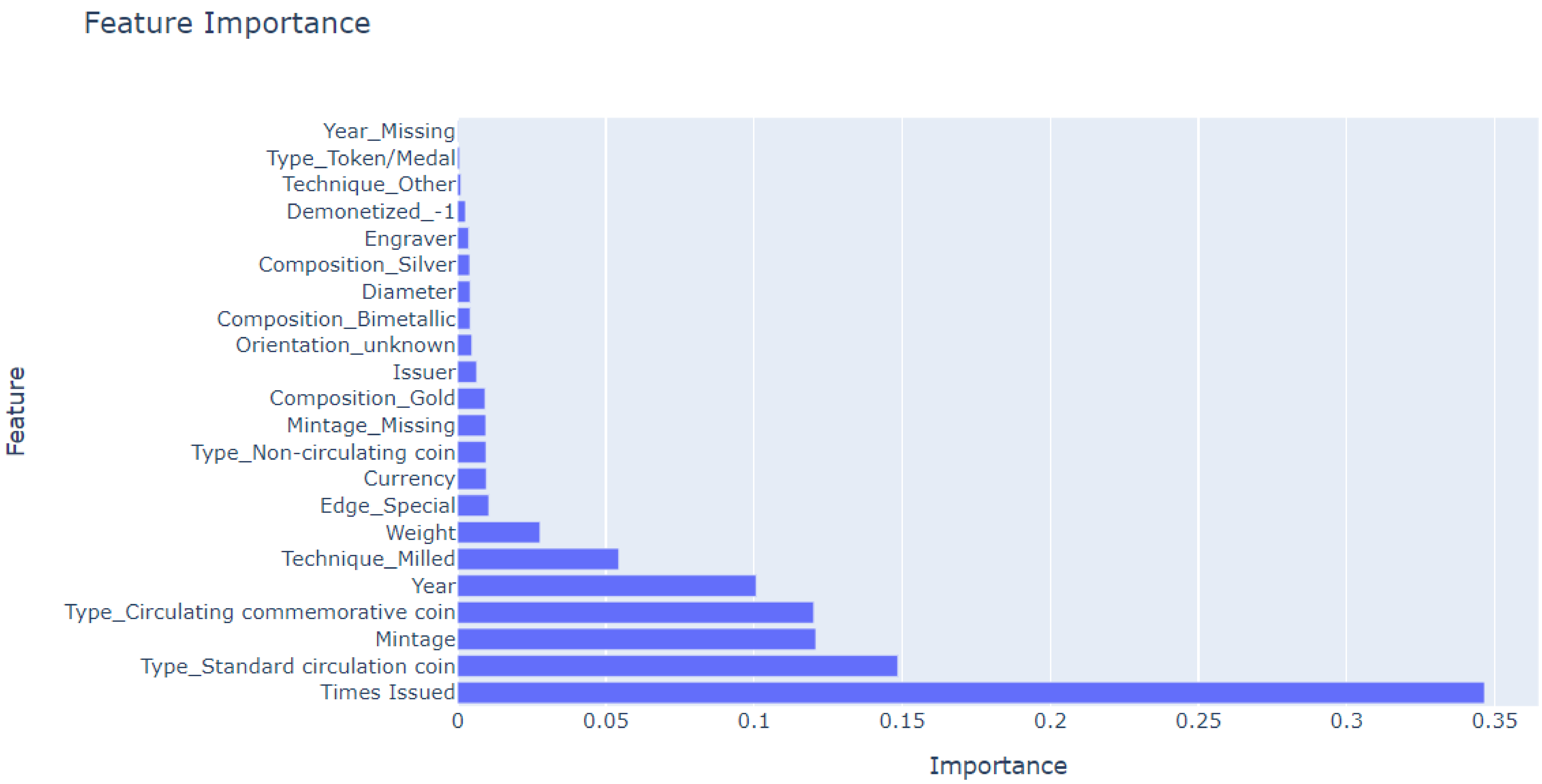
The ultimate goal of this project is to develop a highly accurate model for predicting coin rarity.
Through rigorous data preprocessing, feature selection, model selection, hyperparameter tuning, and ensemble learning.
The performance of the models and the ensemble model will be thoroughly evaluated to assess their effectiveness in achieving this objective.

```
Model Evaluation:

Linear Regression:
MSE: 0.0369
MAE: 0.1467
R2 Score: 0.7413

Decision Tree:
MSE: 0.0185
MAE: 0.0851
R2 Score: 0.8701

Random Forest:
MSE: 0.0137
MAE: 0.0764
R2 Score: 0.9043

Support Vector Machine:
MSE: 0.0196
MAE: 0.0988
R2 Score: 0.8630

Ensemble Model:
MSE: 0.0152
MAE: 0.0871
R2 Score: 0.8936
```

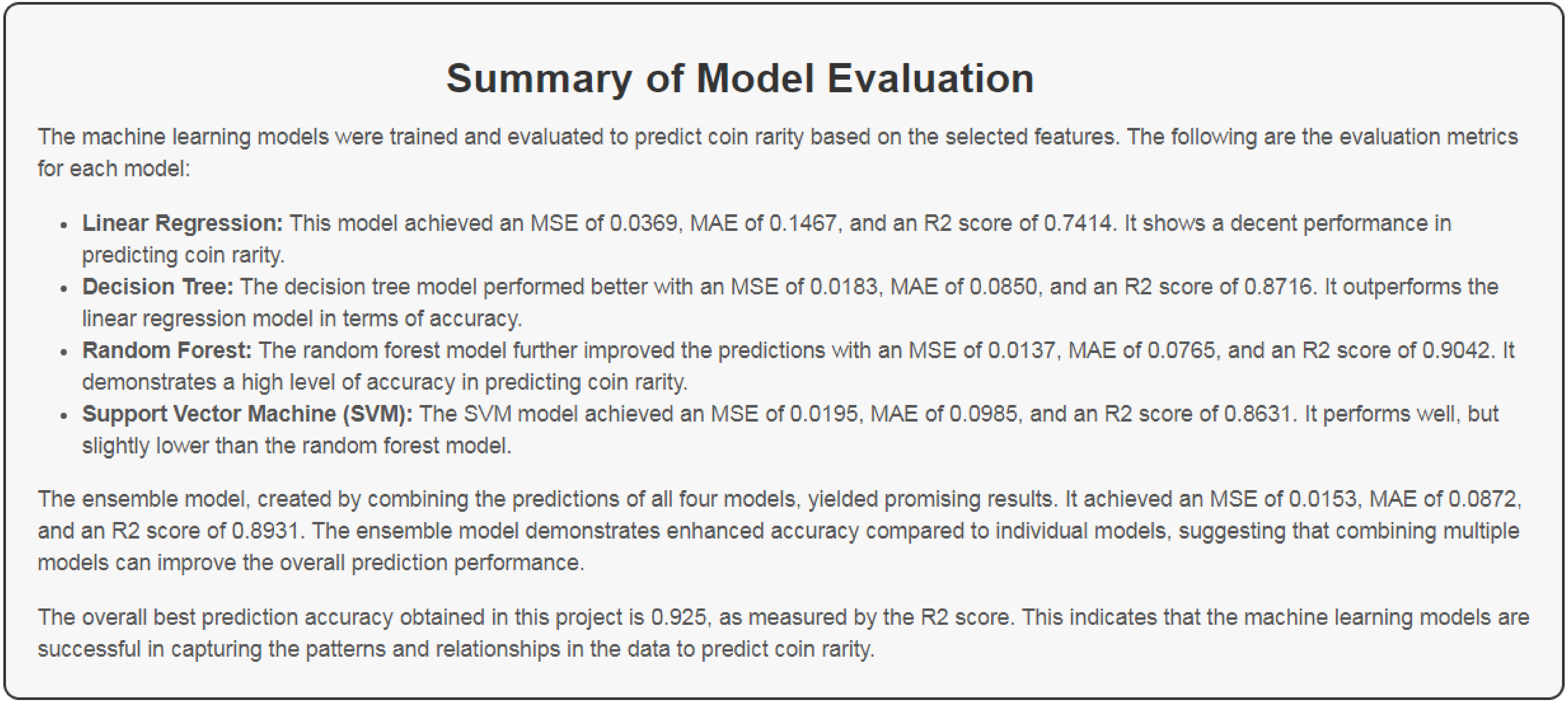


```
Overall Best Prediction Accuracy: 0.9249098672785655

Total Elapsed Time: 01:22:08
```


Actual vs. Predicted Rarity
Predictions
Perfect Fit
Predicted Rarity
Actual Rarity
0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1


Feature Importance
Feature
Year_Missing
Type_Token/Medal
Technique_Other
Demonetized_-1
Engraver
Composition_Silver
Diameter
Composition_Bimetallic
Orientation_unknown
Issuer
Composition_Gold
Mintage_Missing
Type_Non-circulating coin
Currency
Edge_Special
Weight
Technique_Milled
Year
Type_Circulating commemorative coin
Mintage
Type_Standard circulation coin
Times Issued
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
Importance

# SUMMARY



## Summary of Model Evaluation

The machine learning models were trained and evaluated to predict coin rarity based on the selected features. The following are the evaluation metrics for each model:

- **Linear Regression:** This model achieved an MSE of 0.0369, MAE of 0.1467, and an R2 score of 0.7414. It shows a decent performance in predicting coin rarity.
- **Decision Tree:** The decision tree model performed better with an MSE of 0.0183, MAE of 0.0850, and an R2 score of 0.8716. It outperforms the linear regression model in terms of accuracy.
- **Random Forest:** The random forest model further improved the predictions with an MSE of 0.0137, MAE of 0.0765, and an R2 score of 0.9042. It demonstrates a high level of accuracy in predicting coin rarity.
- **Support Vector Machine (SVM):** The SVM model achieved an MSE of 0.0195, MAE of 0.0985, and an R2 score of 0.8631. It performs well, but slightly lower than the random forest model.

The ensemble model, created by combining the predictions of all four models, yielded promising results. It achieved an MSE of 0.0153, MAE of 0.0872, and an R2 score of 0.8931. The ensemble model demonstrates enhanced accuracy compared to individual models, suggesting that combining multiple models can improve the overall prediction performance.

The overall best prediction accuracy obtained in this project is 0.925, as measured by the R2 score. This indicates that the machine learning models are successful in capturing the patterns and relationships in the data to predict coin rarity.