Rochester Institute of Technology of Dubai

ISTE.470.601 – Data Mining and Exploration

Project Report

Fall 2024

Group 3 – TEAM MEMBERS

Abdullah Faisal Naseem afn5363
Ayesha Khanak8591
Muhammad Ahmed Silat mas9468

December 28th 2024

Manifesto page

Abdullah Naseem-
Deliverable 4a,4b,4c,4d,4e,5a

Muhammad Ahmed Silat-
Deliverable 5b(i,ii,iii), 5c(i,ii,iii,iv,v)

Ayesha Khan-
Deliverable 5c(i,ii,iii,iv,v)

All the team members have attempted all parts together and double-checked everyone else's answers.

Table of Contents

Introduction

Introduction

Crowd-funding platforms like “Kickstarter " allow individuals and organizations to raise funds for their projects. Understanding all the factors that affect the success or failure of these projects is crucial.
The real problem lies in identifying the key factors behind the success of kickstarter projects and providing reasonable and actionable insights to creators and funders. We will attempt to answer the following questions:

- Can we find the correlation between features and success rates?
- Can we predict the success rate based on different features?
- What factors most influence success?
- Can we predict whether a project will succeed or fail?

Novelty of this problem

The problem has been studied before. Several researchers and analysts have explored crowdfunding platforms like kickstarter in order identify the trends and the main causes of success of these projects.

Novelty of our solution

-By comparing Logistic Regression and Random Forest, we provide insights into both linear and non-linear relationships which helps in enhancing the interpretability and predictive power of the dataset.
Random Forest's feature importance analysis captures nuanced interactions that are missed in simpler models.
-Detailed visualizations (e.g., confusion matrices, feature importances) make the results accessible to non-technical stakeholders, setting this work apart from algorithm-focused implementations.
-Dual algorithm insights: With the help of logical regression we can allow creators and stakeholders to understand clear linear relationships.

Challenges associated with this problem:

Scalability to large datasets→
Crowdfunding data sets like kickstarter projects are large and multidimensional hence due to the high amounts of data the complexity increases. Data processing also takes place at a slower speed due to the high number of records.

Low accuracy of predictive models→
The relationship between data in crowdfunding data are often nonlinear and complex hence, simple models can easily fail to capture these subtleties.

Data quality issues→
Missing, noisy or inconsistent data is common and can easily impact the reliability of our analyses.

Data Mining Techniques Used

We used various data mining techniques such as correlation analysis, Random forest, and logistic regression.

CORRELATION ANALYSIS: Correlation analysis identifies the relationship between attributes contained in the dataset.

RANDOM FOREST: Random forest is a machine learning ensemble technique that is for the classification of regression tasks.

LOGISTIC REGRESSION: Logistic regression is a statistical method used for binary classification problems.

APPROACH

Overview

i) Describe what data mining algorithms you choose and how they will help you mine the collected data.
We have chosen various data mining techniques such as correlation, random forest, and logistic regression.

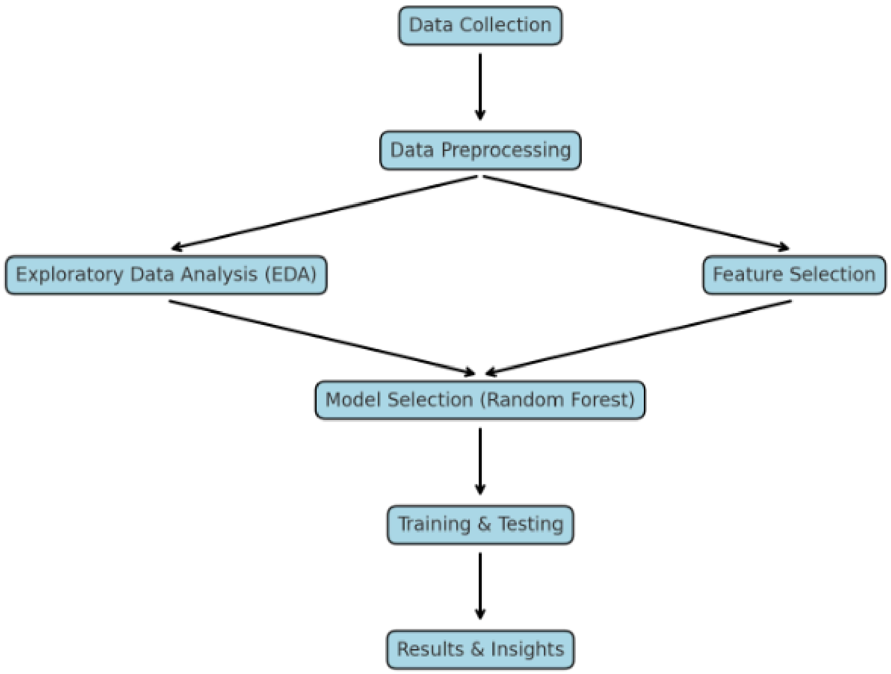
CORRELATION:
Correlation detects patterns and dependencies between attributes. It is foundational for understanding relationships between any attributes and guiding predictive modeling or decision-making processes.
Use case: Correlation will help us by identifying the dependencies between various attributes and the success or failure rate of the project.

RANDOM FOREST:
Random forest is a machine learning technique that uses an ensemble of decision trees to perform various types of classifications or regression tasks.
Use case: Random forest will greatly help us in predicting each kick starter project's success and handling any missing data.

LOGISTIC REGRESSION:
Chosen for its interpretability and effectiveness in binary classification tasks.
It helps to identify any factors that can affect the success of Kickstarter projects and provides probabilities for its success.
Use Case: Establishes a baseline model for predicting whether a campaign will succeed based on attributes like funding goal, category, backers, etc.

Diagram showing all the components of your approach.

Diagram: Approach for Kickstarter Data Analysis



Dataset

i) State the source from which got the dataset.
We obtained the dataset online from a website called Kaggle (https://www.kaggle.com/datasets/kemical/kickstarter-projects)

ii)What is the dataset about?

Our dataset shows every Kickstarter project detail up till 2018 and includes all details about including categories, sub-categories, goal amount and its state as in if it was successful or fails to reach the goal amount.

2. What does each record represent?
Each record represents a singular project that was introduced and also details how much they were looking to raise and if they reached the goal amount. Specifically it

includes data about the project, such as its name, category, launch date, funding goal, amount pledged, the number of backers, and the final state of the project (e.g., successful, failed, or canceled). Each record captures all the data that is relevant up till 2018.

3. How many data objects (e.g., customers) does the dataset contain? How many records are there in the data set? Does each record belong to a single data object?

Total number of objects in the dataset is 378,661 with a total of 378,661 records.
Each record corresponds to a single kickstarter project hence, we can say that each record belongs to one data object so they both have the same number of values.

4. Attribute description: How many attributes? What are these attributes? What are the types (e.g., discrete or continuous) and domain values for each attribute?

The dataset has 14 attributes
1.ID → A unique identifier for each kickstarter project. (Discrete) Domain value: Integer
2.Name→ The name is of the kickstarter project (Categorical) Domain value: Text
3.category→ The subcategory of the project, providing a more specific classification. (Categorical) Domain value: Text
4.main_category→ The broader, primary category under which the project falls.(Categorical) Domain value: Text
5.currency → The currency used for the project's fundraising goal and pledged amounts.(Categorical) Domain value: Text
6.deadline → The date by which the project must meet its funding goal.(Categorical) Domain value: Date
7.goal → The target amount of money the project aims to raise, expressed in the specified currency.(Continuous) Domain value: Positive float
8.launched →The date and time when the project was launched on Kickstarter.(Categorical) Domain value: Date
9.Pledged→ The total amount of money pledged to the project in its specified currency by the end of the campaign. (Continuous) Domain value: Positive float
10.state→ The final status of the project. (Categorical) Domain value: {successful, failed, cancelled, etc.}
11.backers→ The number of people who contributed money to the project.(Discrete) Domain value: Integer
12.usd pledged→ The amount pledged to the project in USD, converted at the time of the campaign.(Continuous) Domain value: Positive float

13.Usd_pledged_real→ The final pledged amount in USD, adjusted for currency conversion at the campaign's end. (Continuous) Domain value: Positive float
14.Usd_goal_real→ The fundraising goal converted to USD at the campaign's end. (Continuous) Domain value: Positive float

5. Table to summarize the dataset.

ATTRIBUTE	TYPE	DESCRIPTION/DOMAIN	DOMAIN
ID	Discrete	A unique identifier for each kickstarter project.	Integer
name	Categorical	The name of the kickstarter project.	Text
category	Categorical	The subcategory of the project, providing a more specific classification.	Text
main_category	Categorical	The broader, primary category under which the project falls.	Text
currency	Categorical	The currency used for the project's fundraising goal and pledged amounts.	Text
deadline	Categorical	The date by which the project must meet its funding goal.	Date
goal	Continuous	The target amount of money the project aims to raise, expressed in the specified currency.	Positive float

launched	Categorical	The date and time when the project was launched on Kickstarter	Date
pledged	Continuous	The total amount of money pledged to the project in its specified currency by the end of the campaign.	Positive Float
state	Categorical	The final status of the project.	{successful, failed, cancelled, etc.}
backers	Discrete	The number of people who contributed money to the project.	Integer
Usd pledged	Categorical	The amount pledged to the project in USD, converted at the time of the campaign	Positive Float
usd_pledged_real	Continuous	The final pledged amount in USD, adjusted for currency conversion at the campaign's end.	Positive Float
usd_pleged_real	Continuous	The fundraising goal converted to USD at the campaign's end.	Positive Float

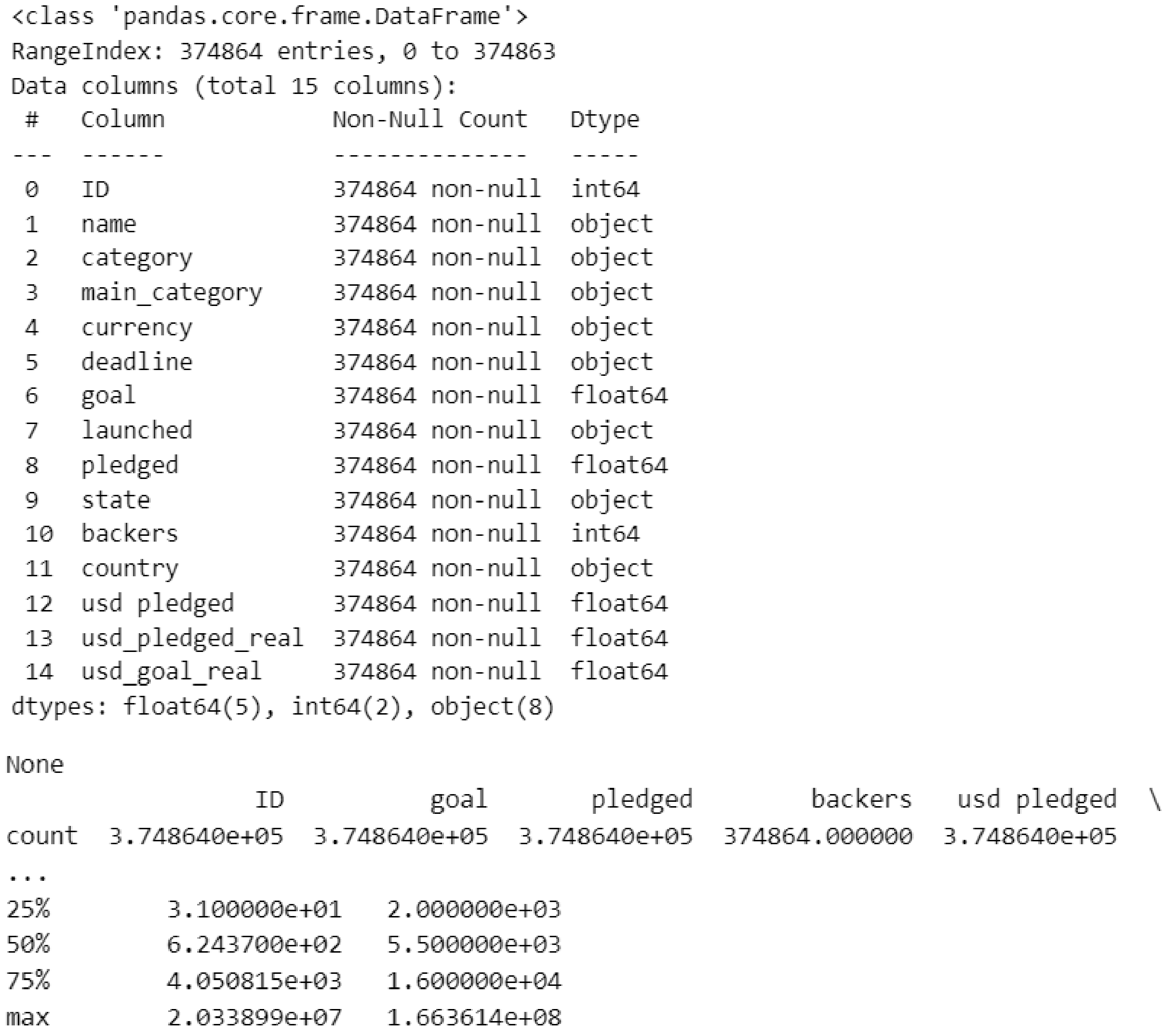
Preprocessing

1. Understanding the Dataset:

We tried to understand all the attributes and get an overview of data types and missing values:

```
print(data.head())  # View the first few rows
print(data.info())  # Get an overview of data types and missing values
print(data.describe())   # Summary statistics for numerical columns
```

This gives us an idea of what we're dealing with :



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 374864 entries, 0 to 374863
Data columns (total 15 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   ID                374864 non-null  int64
 1   name              374864 non-null  object
 2   category          374864 non-null  object
 3   main_category     374864 non-null  object
 4   currency          374864 non-null  object
 5   deadline          374864 non-null  object
 6   goal              374864 non-null  float64
 7   launched          374864 non-null  object
 8   pledged           374864 non-null  float64
 9   state             374864 non-null  object
 10  backers           374864 non-null  int64
 11  country           374864 non-null  object
 12  usd pledged       374864 non-null  float64
 13  usd_pledged_real  374864 non-null  float64
 14  usd_goal_real     374864 non-null  float64
dtypes: float64(5), int64(2), object(8)

None
                 ID          goal       pledged        backers   usd pledged  \
count  3.748640e+05  3.748640e+05  3.748640e+05  374864.000000  3.748640e+05
...
25%        3.100000e+01   2.000000e+03
50%        6.243700e+02   5.500000e+03
75%        4.050815e+03   1.600000e+04
max        2.033899e+07   1.663614e+08
```

2. Data Cleaning:

Finding missing values: We first checked for all the missing values in the database:

```
print(data.isnull().sum())
```

Handling Missing Values: Our first step is to remove any missing values from the dataset:

- For names, any missing names were filled with “Unknown” as the names of the projects are not important in our analysis.
  ```
  data['name'] = data['name'].fillna('Unknown')
  ```
-

Remove duplicate or unnecessary columns:

- We removed the USD_Pledged column because it has roughly the same values as USD_Pledged_Real. The difference between them is mentioned above.
- We also removed the Goal attribute as it is not useful to compare values if they are in different currencies. The USD_Goal_real attribute works better in this situation.
  ```
  columns_to_drop = ['goal', 'usd_pledged']
  data.drop(columns=columns_to_drop, errors='ignore')
  ```

3. Correct Data Types

Upon looking at the data types, we found an issue so we converted the launched and deadline attribute to the correct data type in order to process it better:

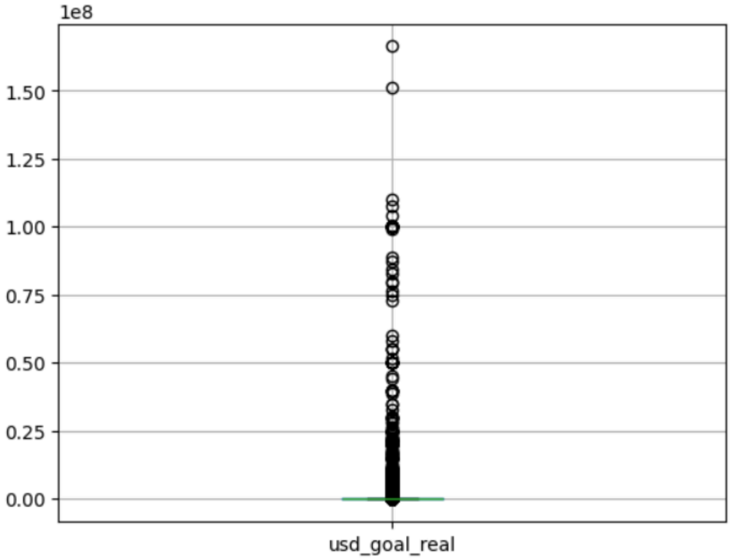
```
data['launched'] = pd.to_datetime(data['launched'], errors='coerce')
data['deadline'] = pd.to_datetime(data['deadline'], errors='coerce')
```

We also ensured that our numeric columns are properly formatted:

```
numeric_columns = ['goal', 'pledged', 'backers', 'usd pledged',
'usd_pledged_real', 'usd_goal_real'] data[numeric_columns] =
data[numeric_columns].apply(pd.to_numeric, errors='coe
```

4. Remove Outliers

To check if we had outliers we performed an outlier analysis to give the following result:



This showed that in our goal amount, there were a few outliers that could skew the data. So we removed the outliers using the following code:

```
min_threshold = 0

max_threshold = data['usd_goal_real'].quantile(0.95)  # 95th percentile as
the upper limit
```

Outliers have been removed based on the 99th percentile of the `goal` column. A total of 3,557 outliers were excluded.

One-Hot Encoding for `country` attribute:

- We used `pd.get_dummies()` to create separate binary columns for each country (e.g., `country_GB`, `country_CA`, etc.).
- `drop_first=True` ensures that one category is dropped to avoid redundancy (e.g., `country_US` might be omitted, as its presence is implied when all others are 0).

Data Analysis

Exploratory Data Analysis (EDA)

To start off, we decided to conduct an Exploratory Data Analysis (EDA) to understand the dataset better before building any predictive models. EDA is mostly the first step in the data mining process as it helps us explore the data even better and discovers patterns through tools like histograms and graphs. Our aim here is to discover more about the development of future models that can be used for classification.

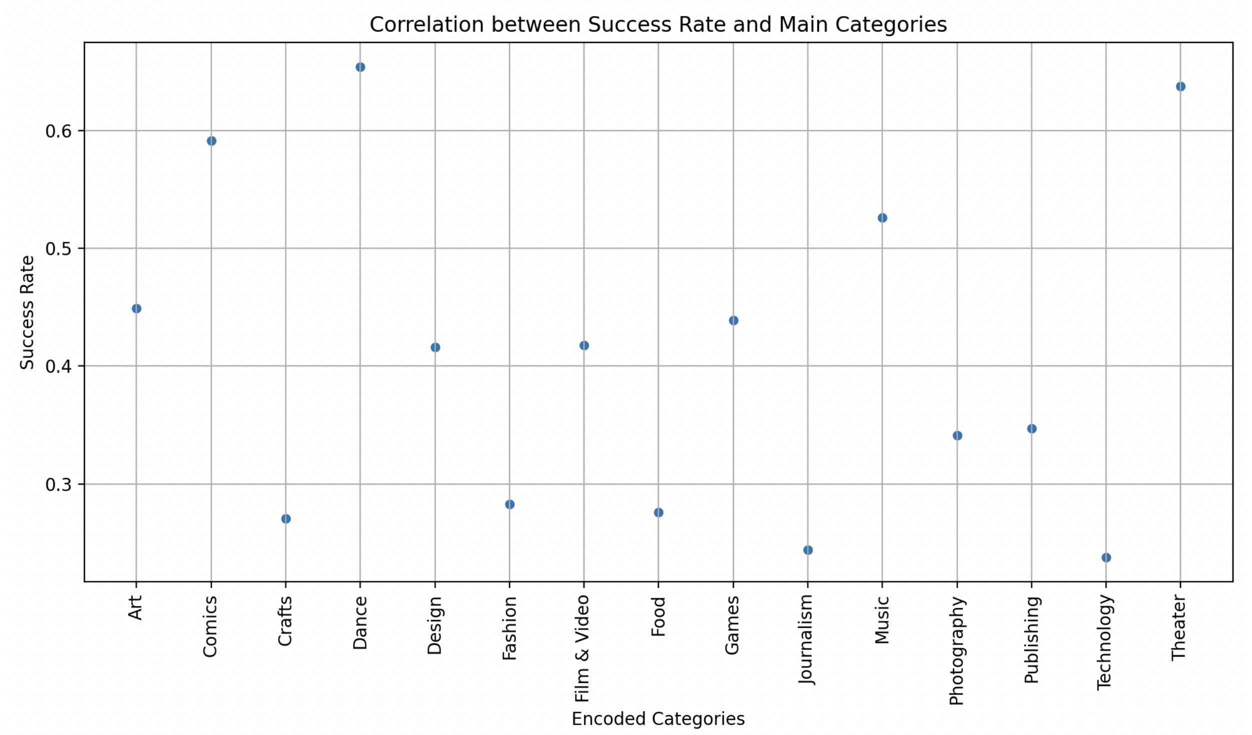
-Finding correlation between success rate and main categories

The following scatterplot diagram shows the success rate of different categories in the data set. Each category has been assessed for its success rate, and suggests that the dataset includes projects, campaigns or other initiatives grouped by these categories.

Data Mining Perspective:
-Correlation/Pattern Identification: The scatterplot highlights how the success rate for each category is different and varies.

Insights:
-Categories like music and theatre have a higher success rate than other categories like Technology and publishing.
-The success rate for different categories varies, indicating that some categories might perform good or bad.

Possible applications:
-Visualisation in EDA: Helps us understand the distribution of data and focus on areas.



We further found the correlation between the success rate and sub categories of dance and theatre, and got the following results:

Dance Subcategory Corellation:

- **Performances** and **Residencies** have high success rates (above 0.6), likely due to their broad appeal and established audience.
- **Workshops** show the lowest success rate in Dance, possibly due to a niche audience or unclear outcomes.

Theater Subcategory Corellation:

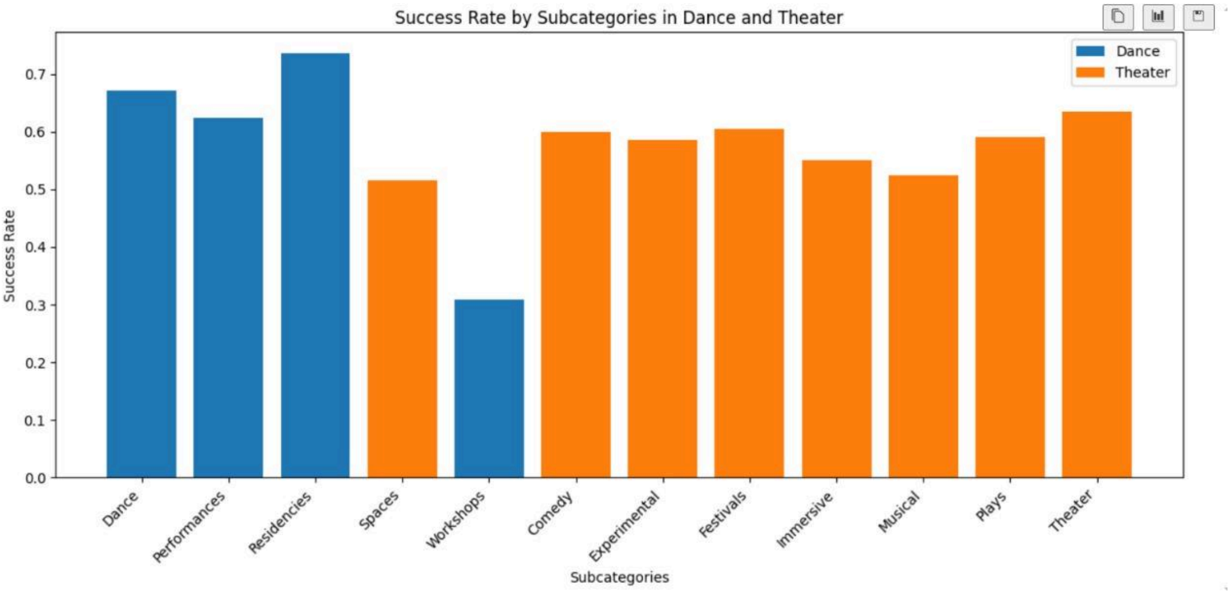
- Categories like **Plays**, **Musicals**, and **Experimental Theater** have consistently high success rates (above 0.6), which reflects the audience having a strong interest.
- Subcategories like **Spaces** and **Comedy** perform slightly worse, possibly due to higher funding requirements or narrower appeal.

General Trends:

- **Theater campaigns** are more consistent in their success rates compared to Dance.
- Dance campaigns show greater variability, with some categories (like Workshops) struggling to succeed.

Insights:

- ○ Dance campaigns should focus on high-performing subcategories like **Performances** and improve strategies for **Workshops**.
- ○ Theater campaigns can leverage the popularity of **Plays** and **Musicals**, while improving the approach for **Spaces** and **Comedy**.



Finding correlation between success rate and currency

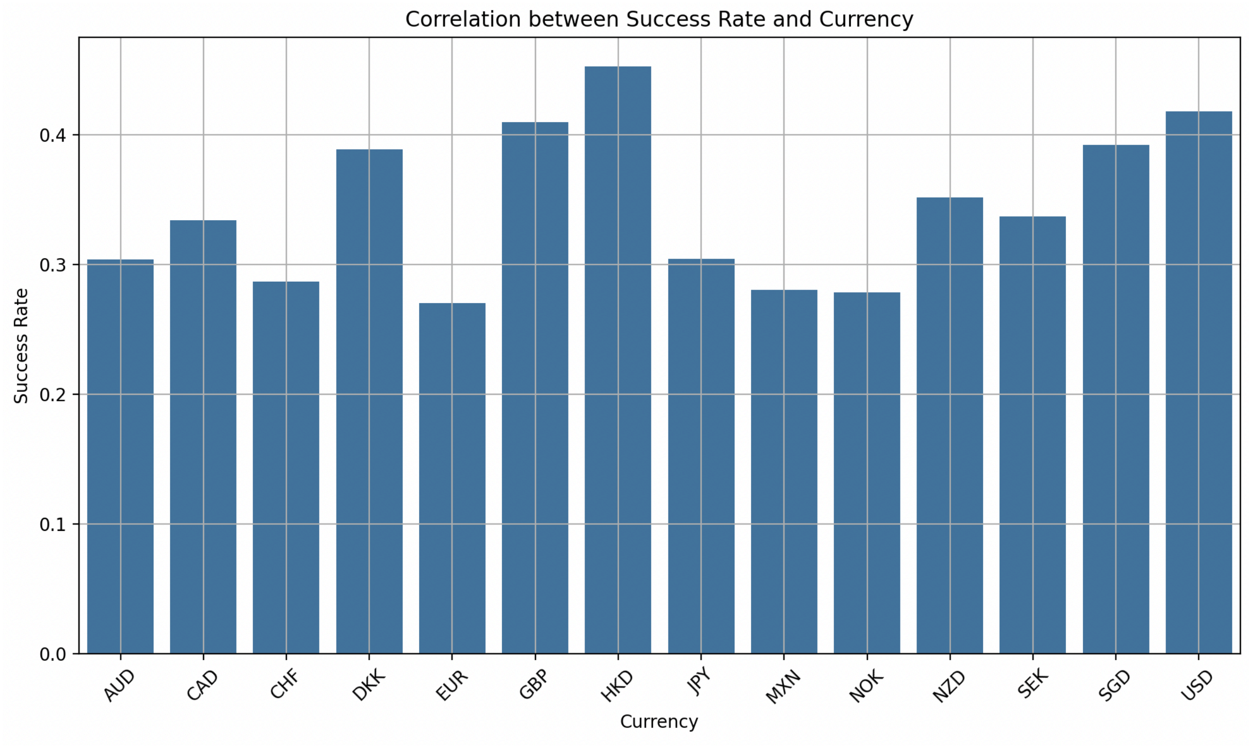
The following barplot shows the success rate of projects based on various currencies. Every bar represents a currency e.g USD or EUR and its success rate.

Data Mining Perspective:

-It highlights the relationship between currency and and the success rate(the target variable)

Insights:
-Currencies like USD and HKD have a higher success rate than currencies like EUR and NOK. This shows a more supportive and active audience for projects that use that currency.

Possible applications:
-Helping organizations decide on which currency to use when launching their campaigns and this will help improve the success rate.



	State	Main category	country
state	**1.000**	**0.119819**	**0.058256**
main_category	**0.119819**	**1.000**	**0.055165**
country	**0.058256**	**0.055165**	**1.000**

Cramér's V Interpretation:

0.00 to 0.10: Very weak or no association.
0.10 to 0.30: Weak association.
0.30 to 0.50: Moderate association.
0.50 to 1.00: Strong association.

Analysis:

1. **Currency and Success Rate**:
    - **Highest Success Rates**: Campaigns in **JPY (Japanese Yen)** and **USD (US Dollar)** perform best, likely due to larger markets and higher trust.
    - **Moderate Success Rates**: **GBP (British Pound)** and **HKD (Hong Kong Dollar)** show decent performance.
    - **Lowest Success Rates**: Currencies like **CHF (Swiss Franc)**, **MXN (Mexican Peso)**, and **DKK (Danish Krone)** may reflect smaller or niche markets.
2. **Main Categories and Success Rate**:
    - **Highest Success Rates**: **Theater** and **Dance** perform best due to niche audiences and strong community backing.
    - **Moderate Success Rates**: Categories like **Music**, **Film & Video**, and **Comics** appeal to broader audiences.
    - **Lowest Success Rates**: **Technology** and **Crafts** struggle, likely due to high risks or funding complexities.

Predictive Modelling

We will apply different techniques and then compare their results:

Approach 1: Logistic Regression

Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given data set of independent variables.

Goal: Predict whether a Kickstarter project will succeed.

- Input: Features like goal, backers, and duration.
- Output: Probability (e.g., 0.85 → classify as successful).

Using Python in Jupyter Notebook we were able to perform logistic regression:

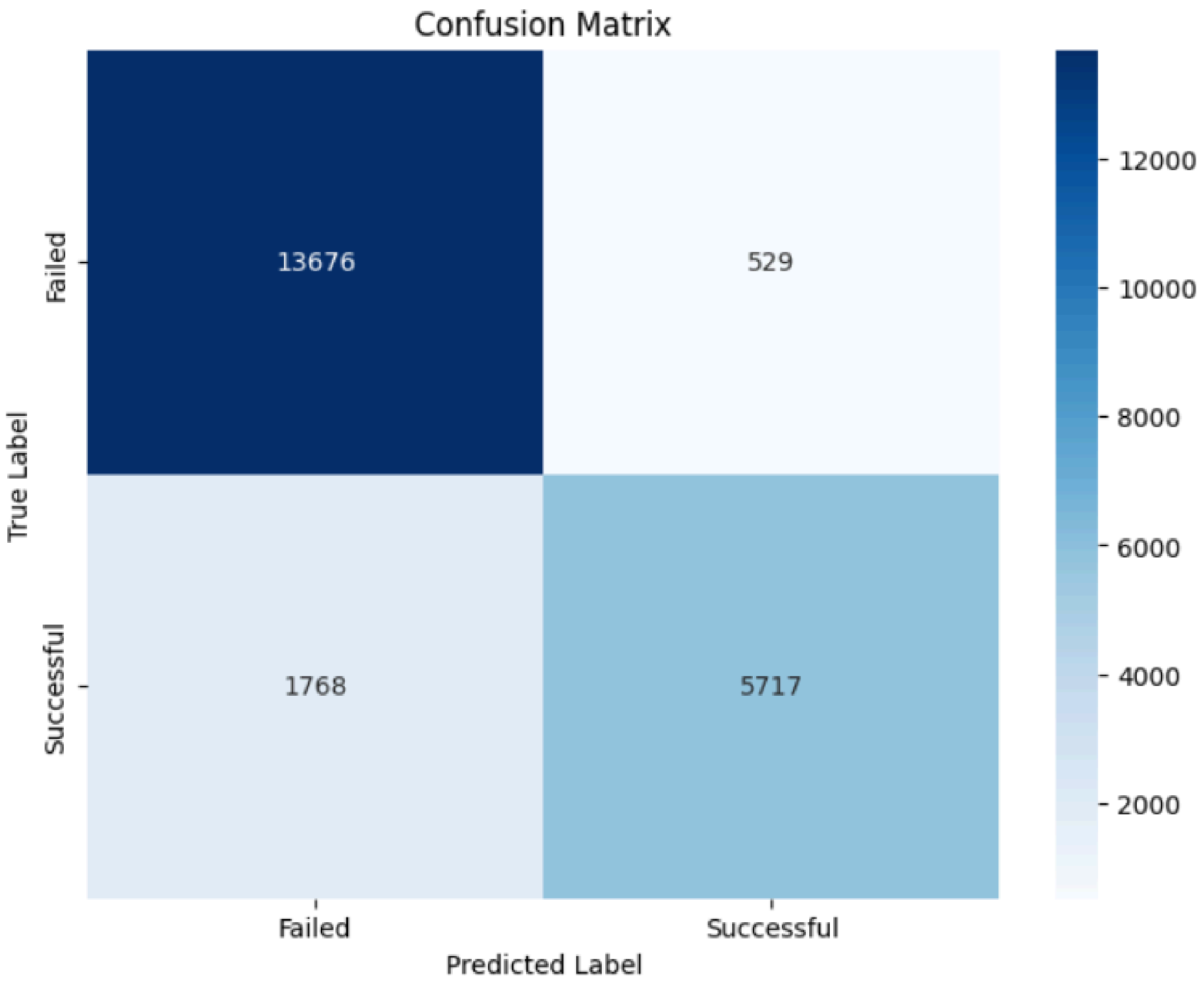
```
Logistic Regression Evaluation:
Accuracy: 0.8940986629783311
Precision: 0.915305795709254
Recall: 0.7637942551770207
F1-Score: 0.8327142961182725

Confusion Matrix:
 [[13676   529]
 [ 1768  5717]]

Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.96      0.92     14205
           1       0.92      0.76      0.83      7485

    accuracy                           0.89     21690
   macro avg       0.90      0.86      0.88     21690
weighted avg       0.90      0.89      0.89     21690
```



Overall Model Performance

- The **accuracy** of 89% indicates the model is effective at predicting Kickstarter project success.
- However, the imbalance between precision and recall for the `successful` class suggests:
  - The model is better at predicting **failures** than **successes**.
  - **Recall for the `successful` class (74%)** means the model misses some successful projects (False Negatives).

Feature Analysis

To identify the most influential features in logistic regression, we use the **model coefficients**. Logistic regression assigns a coefficient to each feature, representing its impact on success. We got the following results:

```
Top 10 Positively Influential Features:
                      Feature  Coefficient
131           category_Shorts     0.588605
177       main_category_Dance     0.499033
163       main_category_Dance     0.499033
25   category_Classical Music     0.407692
74        category_Indie Rock     0.392145
32    category_Country & Folk     0.325077
38             category_Dance     0.320752
138   category_Tabletop Games     0.309134
188     main_category_Theater     0.299096
174     main_category_Theater     0.299096

Top 10 Negative Influential Features:
                      Feature  Coefficient
150      category_Video Games    -0.868011
70            category_Hip-Hop    -0.718215
168       main_category_Games    -0.622516
182       main_category_Games    -0.622516
155        category_Webseries    -0.498219
153              category_Web    -0.368622
162      main_category_Crafts    -0.339948
176      main_category_Crafts    -0.339948
10              category_Apps    -0.299642
173  main_category_Technology    -0.282955
```

We have successfully found what features influence the success and most and which decrease the likelihood of success.

The coefficients represent the relationship between each feature and the target (`state`). The state was broken down in preprocessing into binary values.

Positive coefficients: Increase the likelihood of success.

Negative coefficients: Decrease the likelihood of success.

We hit an error: **"STOP: TOTAL NO. of ITERATIONS REACHED LIMIT"**. this was fixed by adjusting the `max_iter` parameter to allow more iterations. We set it to 10,000 iterations which increased the time it took to run by 5 times but we got the results in 57 seconds.

Summary Table:

Feature	Type	Coefficient	Effect
category_Shorts	Category	+0.588605	Positive Influence
main_category_Dance	Main Category	+0.499033	Positive Influence
category_Classical Music	Category	+0.407692	Positive Influence
category_Indie Rock	Category	+0.392145	Positive Influence
category_Country & Folk	Category	+0.325077	Positive Influence
category_Dance	Category	+0.320752	Positive Influence
category_Tabletop Games	Category	+0.309134	Positive Influence
main_category_Theater	Main Category	+0.299096	Positive Influence
category_Video Games	Category	-0.868011	Negative Influence
category_Hip-Hop	Category	-0.718215	Negative Influence
main_category_Games	Main Category	-0.622516	Negative Influence
category_Webseries	Category	-0.498219	Negative Influence

category_Web	Category	-0.368622	Negative Influence
main_category_Crafts	Main Category	-0.339948	Negative Influence
category_Apps	Category	-0.299642	Negative Influence
main_category_Technology	Main Category	-0.282955	Negative Influence

Interpretation:

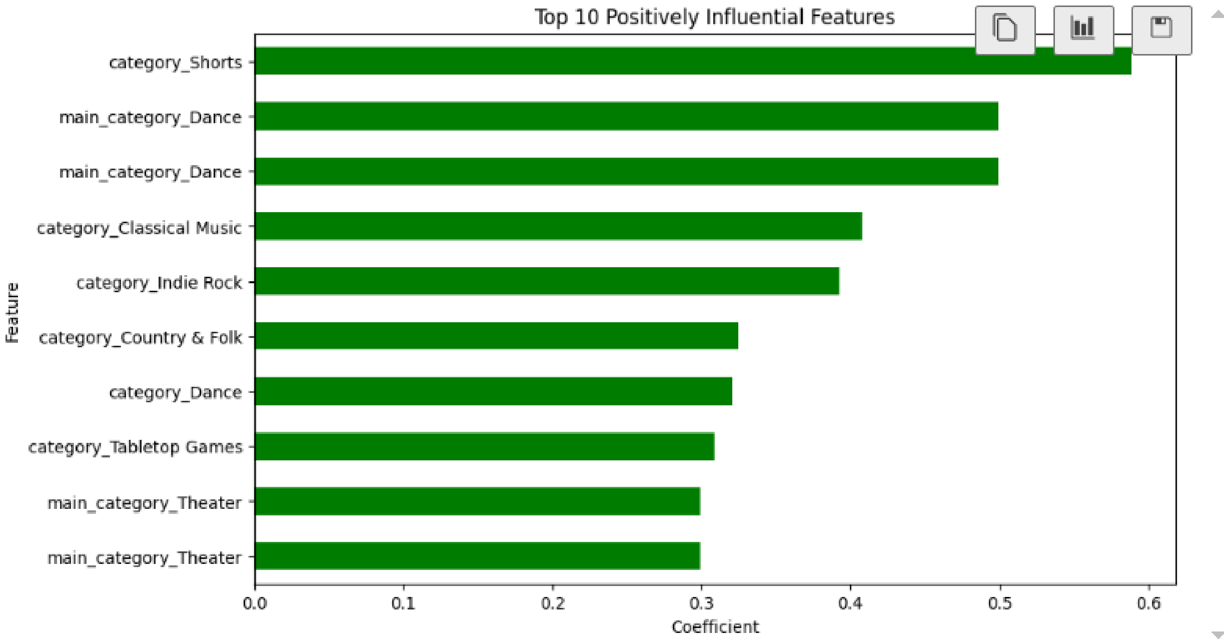
Positively Influential Features:

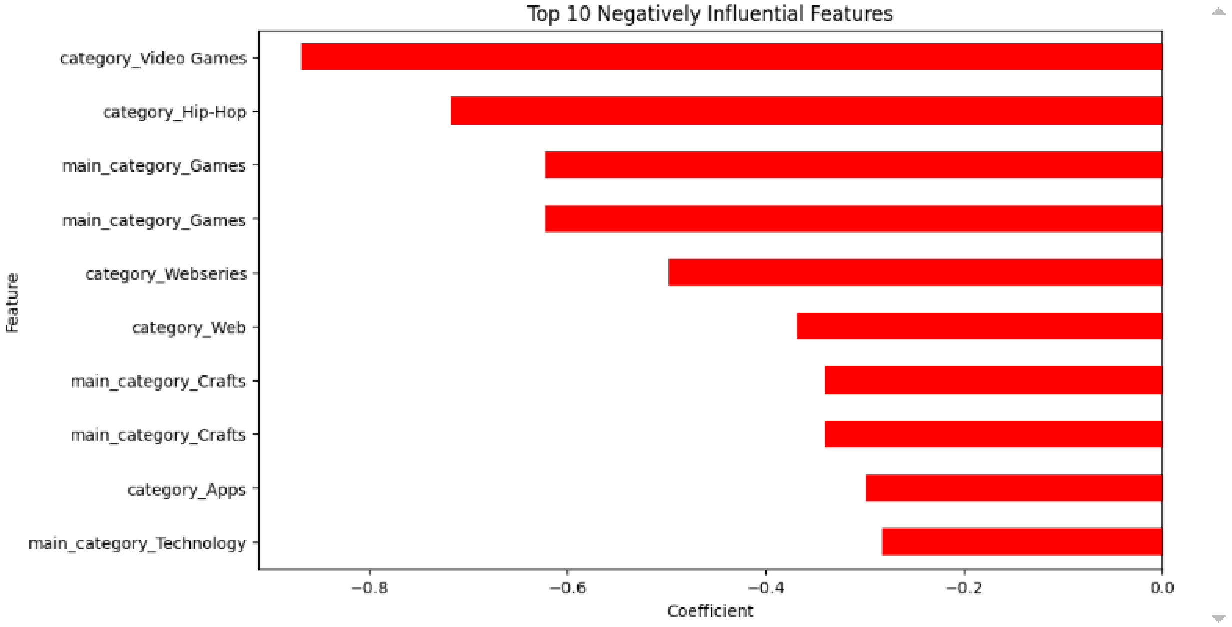
  - Categories like `Shorts`, `Dance`, and `Classical Music` significantly increase the likelihood of a project’s success.
  - Main categories like `Theater` and `Dance` are also associated with higher probabilities of success.

Negatively Influential Features:

- Technology-related categories, such as `Apps` and `Video Games`, tend to underperform

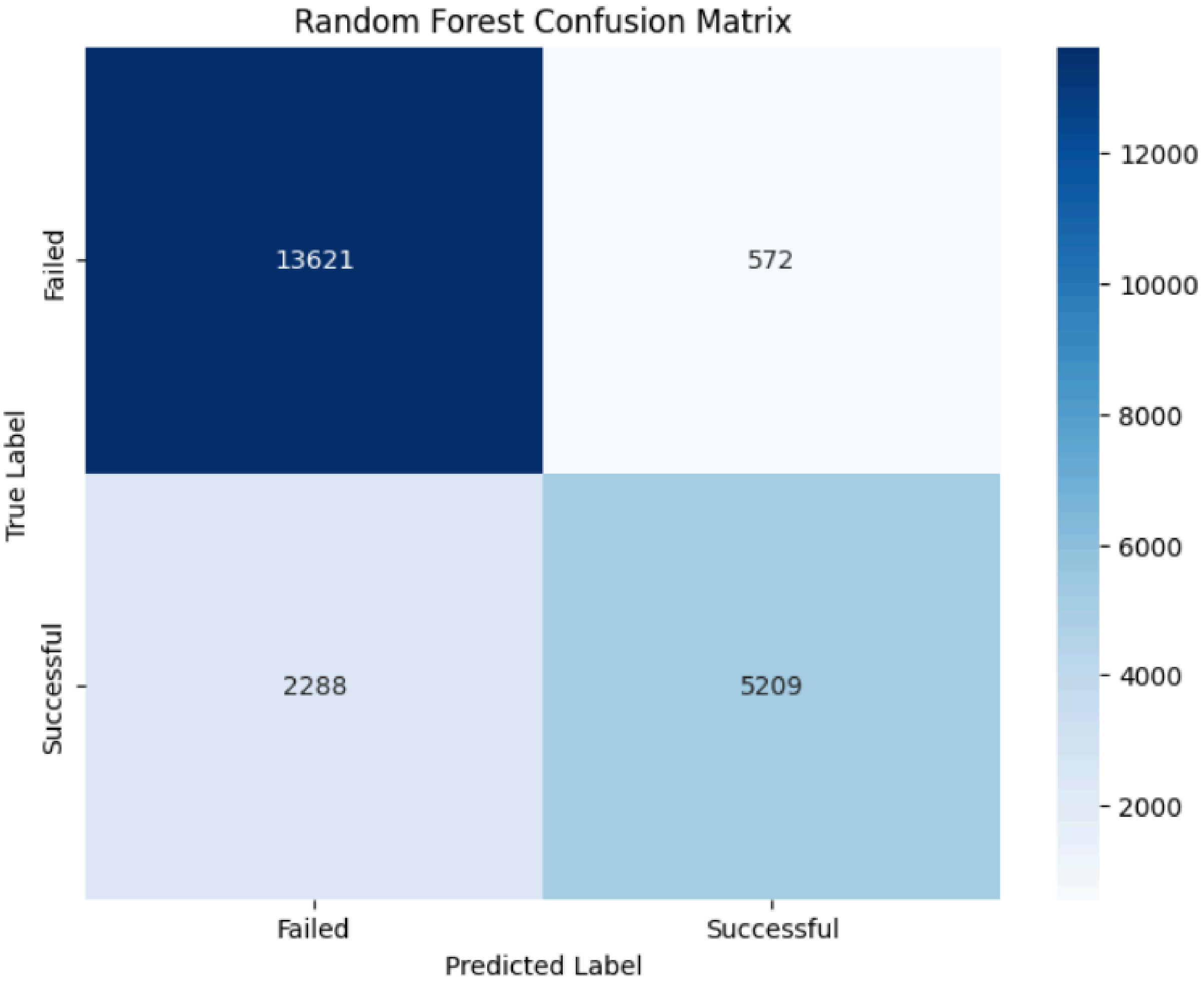
Data Visualisation of Features:





Approach 2: Random Forest Prediction Model

Random Forest is a robust ensemble learning technique that builds multiple decision trees and aggregates their outputs for classification tasks. The confusion matrix provides insights into how well the model performs across both classes.



The following is a confusion matrix that is for evaluating its capability and the performance of classification algorithms. It is essentially a table that summarizes how well a classification model like a Random Forest here has performed in terms of predicting different classes

Objective:

- The goal is to predict whether a Kickstarter project will be **successful** or **not successful** using features like:
  - `main_category`

        - `usd_goal_real`
        - `backers`
        - `country`.

Model:

    - A **Random Forest Classifier** is trained:
        - Uses 100 decision trees.
        - Limits tree depth to prevent overfitting.
    - The model predicts success on a test dataset.

Confusion Matrix:

    - Visualized with a heatmap showing:
        - **True Positives (TP)**: Correctly predicted successful projects.
        - **True Negatives (TN)**: Correctly predicted unsuccessful projects.
        - **False Positives (FP)**: Wrongly predicted unsuccessful as successful.
        - **False Negatives (FN)**: Wrongly predicted successful as unsuccessful.

Confusion Matrix Details:

- **True Negatives (TN)**: 13,621 — Instances that are correctly classified as "Not Successful."
- **False Positives (FP)**: 572 — Instances incorrectly classified as "Successful" when they are actually "Not Successful."
- **False Negatives (FN)**: 2288 — Instances incorrectly classified as "Not Successful" when they are actually "Successful."
- **True Positives (TP)**:$ — Instances correctly classified as "Successful."

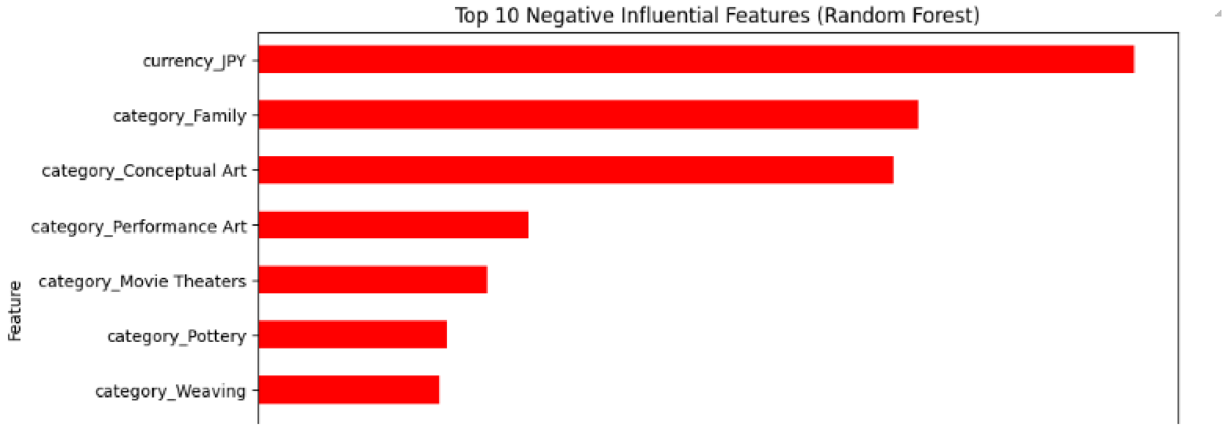
Key Insights

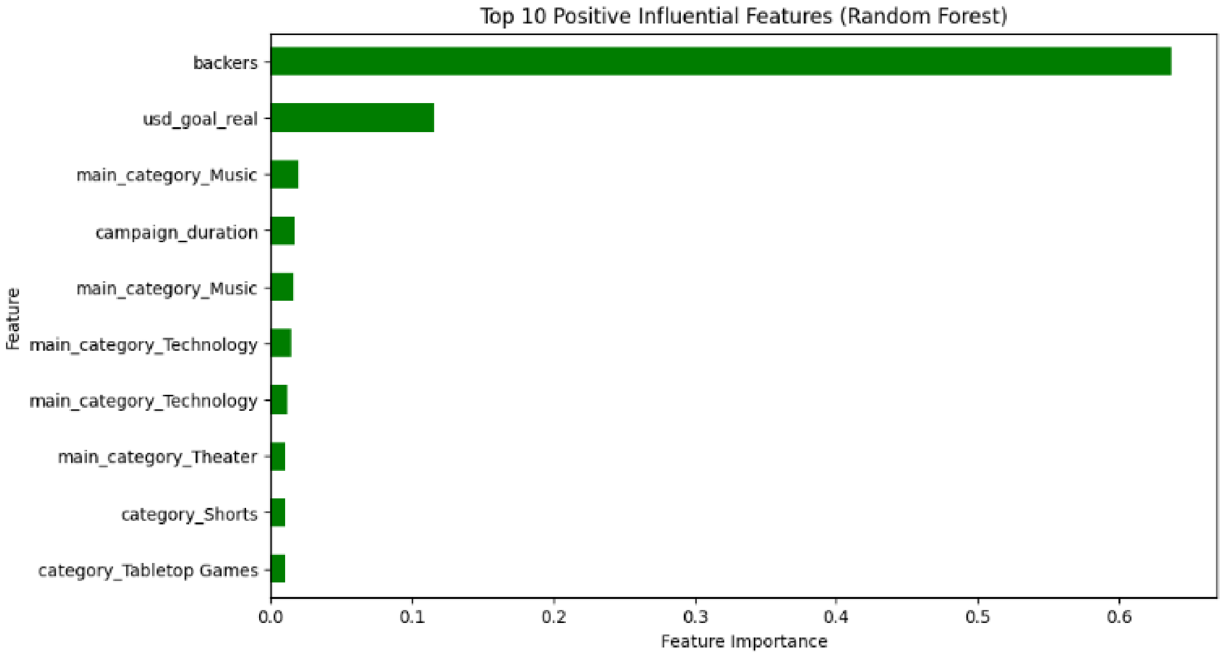
- **Good Predictions**: A high count in diagonal cells (TP and TN) indicates strong performance.
- **Errors**: Off-diagonal cells (FP and FN) highlight misclassifications.
- **Features**: Variables like `backers` and `usd_goal_real` are likely key drivers of success.

```
Top 10 Influential Features in Random Forest:
```

```
                         Feature  Importance
2                        backers    0.637635
0                  usd_goal_real    0.115032
184          main_category_Music    0.019885
1              campaign_duration    0.016763
170          main_category_Music    0.016238
187     main_category_Technology    0.014478
173     main_category_Technology    0.012110
188        main_category_Theater    0.010209
131              category_Shorts    0.009814
138      category_Tabletop Games    0.009729
```

Data Visualization of Features





Repeating the feature analysis for Random Forest, we get the most influential features

Conclusion:

Summary of Correlation Results:

Currency and Success Rates:

- Campaigns in **JPY** and **USD** exhibit the highest success rates, indicating strong market presence and trust in these currencies. Creators in these markets should leverage this trust to maximize their chances of success.
- Lower success rates in currencies like **CHF**, **MXN**, and **DKK** suggest challenges in smaller markets.

Main Categories and Success Rates:

- Categories like **Theater** and **Dance** stand out as highly successful due to their niche audiences and strong community backing, offering clear pathways for creators to engage their target markets.
- Categories such as **Technology** and **Crafts** face significant struggles, likely due to inherent risks with tech.

Summary of Analysis Results:

- **Logistic Regression**:
    - Achieved **89% accuracy**, effectively predicting the success of Kickstarter campaigns.
    - Top positively influential features include categories such as Shorts, Dance, and Classical Music, which significantly increase the likelihood of success.
    - Negatively influential features, such as Video Games and Hip-Hop, were associated with lower success rates.
    - Logistic regression highlighted clear linear relationships between features and outcomes, making it interpretable and effective as a baseline model.

- **Random Forest**:
    - Provided a robust non-linear model with the ability to capture complex relationships.
    - The confusion matrix showed a balanced classification, with significant improvement in identifying successful campaigns compared to logistic regression.
    - Feature importance revealed key predictors like the number of backers and the campaign duration, along with specific categories like Tabletop Games and Indie Rock.

Algorithm Performance

- **Logistic Regression**:
    - Strengths:
        - Easy to implement and interpret.
        - Highlights clear relationships between features and the target variable.
    - Limitations:
        - Limited in capturing non-linear relationships.
        - Struggled with imbalanced datasets.
- **Random Forest**:
    - Strengths:
        - Robust to overfitting with parameter tuning.
        - Handles non-linear relationships and feature interactions well.
        - Feature importance provides actionable insights.
    - Limitations:

        - Less interpretable than logistic regression.
        - Computationally intensive, especially with large datasets.
- **Why Some Algorithms Were Not Used**:
    - **Deep Learning**:
        - Overkill for a relatively structured dataset like Kickstarter.
        - Computationally expensive without significant performance gains.
    - **K-Nearest Neighbors**:
        - Computationally inefficient for large datasets.
        - Poor performance in high-dimensional spaces.

Feature Overlap Between Logistic Regression and Random Forest:

Feature overlap between the two models enhances confidence in their importance:

- Shared features like `backers` and `campaign duration` are consistently predictive of success.
- Differences in feature importance (e.g., granular categories vs. broad predictors) highlight the complementary nature of the models:
    - **Logistic Regression** excels at identifying clear, interpretable trends.
    - **Random Forest** uncovers nuanced interactions that might otherwise be missed.

Insights:

Our data mining revealed that **Project Creators** should Focus on categories with higher probabilities of success, such as Dance, Shorts, and Tabletop Games.

    - Keep fundraising goals realistic, as overly ambitious goals reduce success rates.
    - Keep a reasonable duration for the campaign as it will affect their success rates.

Future Work:

    - Explore additional features, such as sentiment analysis of campaign descriptions, to refine predictions.
    - Continuously update models with new data to capture new trends as this dataset was last updated in 2018.

Final Thoughts

Both logistic regression and random forest provided meaningful insights into the Kickstarter campaign's success. Logistic regression excelled in simplicity and interpretability, while random forest captured nuanced relationships, making it a more powerful tool for prediction. Future models could integrate additional features, address biases, and enhance fairness to create a more holistic and equitable system.