Data Preprocessing

First, I cleaned the dataset according to the guidelines provided. To focus on features available at the project's launch, I removed columns related to state_changed, staff_pick, spotlight. I assumed that Kickstarter's content team decides on promoting projects later, after launch.

Columns like id and name were also removed, as they are not predictive features. I calculated the goal amount in USD, retained it, and removed other redundant columns related to the goal.

Upon analyzing the correlation matrix, I noticed a high correlation between years of deadline, launched_at, and created_at. To reduce redundancy, I kept only launched_at_yr. Additionally, I excluded the created_at timestamps, assuming this is more of internal data unavailable to users when deciding whether to pledge. Instead, I introduced two derived metrics:

1. Time between the project creation and launch (measure of team's speed and seriousness)
2. Time between the project launch and deadline (a measure of urgency and momentum).

For other time-related features, I retained hour, weekday, and month, as they have the potential to affect users' behavior toward them. I tested combining weekdays into a binary weekend/weekday variable but saw no performance improvement, so I kept all seven weekdays.

After implementing these changes, I reviewed the correlation matrix and calculated the Variance Inflation Factor (VIF). The analysis confirmed that multicollinearity was no longer an issue.

A deeper analysis of some scattered categorical variables was also necessary:

1. For Country, I kept those appearing in over 1% of rows and grouped the rest as "Other."
2. For Category, I analyzed the relationship between Main category and Category. I concluded that they almost have a one-to-many relationship. Consequently, I dropped Category and used Main category as a generalized version.

At last, we converted datatypes, standardized the data, and dummified categorical variables.

Task1: Develop a classification model

I first tried logistic regression with L2 (Ridge) regularization, testing different regularization levels. With an optimal C value of 1291, I achieved 71.41% accuracy and extracted the coefficients. I also used KNN with features selected from the most important logistic regression

predictors, but it resulted in less favorable results. Finally, I tried random forest classification, and I got best model performance metrics so here I will talk about this model in detail.

I got the best result from the random forest classifier with these hyperparameters:
{'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 200}

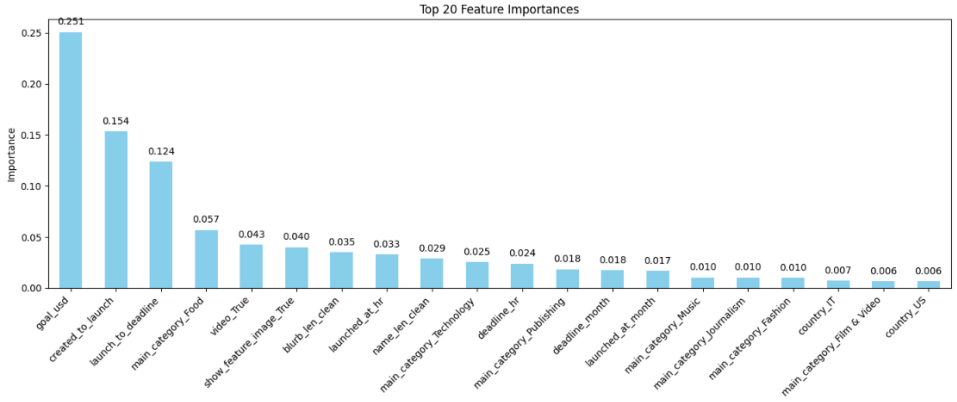
The best model performance metrics are as follows:

Accuracy on test set	Precision on test set	Recall on test set	F1 Score on test set
0.726	0.705	0.876	0.781

- ✓ In 72.6% of the time, the model correctly predicted the project's state (0 or 1). Since the state column in the dataset was balanced, accuracy is a good metric here.
- ✓ A high F1 score shows that the model achieves a good balance in identifying successful projects while keeping false positives relatively low.

Random Forest doesn't indicate the direction of a predictor's impact, so my interpretation are influenced by logistic regression coefficients' signs:

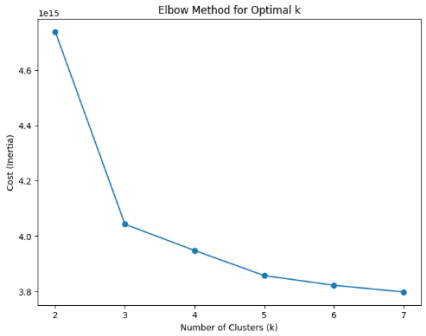
- ✓ The most important feature is the goal amount, suggesting that projects with very high goals may seem overly ambitious, deterring backers and resulting in failure.
- ✓ The two derived feautures became 2nd and 3rd in importance. Likely, shorter time from creation to launch and launch to deadline of the project increase the chance of success.
- ✓ The model predicts more accurately in Food, Technology, and Publishing categories.
- ✓ Having a video or a feauture image increases the chance of attracting users and succeed.
- ✓ A longer name and a shorter blurb (or description) will increase the chance of success.
- ✓ The model performs better at predicting the state of the project from Italy and USA.



Task 2: Develop a clustering model

Since we have mixed data, we used K-Prototype clustering. It combines K-Means and K-Modes, using Euclidean distance for numeric data and matching dissimilarity for categorical data.

Here you can see the elbow graph which shows us 5 is good choice for n_clusters (I also tried 3).



After we run the clustering, we get the silhouette score of 0.978 which indicates the clustering is highly effective, and the separation between clusters is very clear. Here are the interpretations:

- ✓ Cluster 0: High-performing campaigns, likely with large-scale goals and broad appeal. well-planned and thoroughly described, often in high-demand technology categories.
- ✓ Cluster 1: Low-performing campaigns, poor descriptions, or niche appeals that fail to engage backers. They were not staff picked and did not get promoted on spotlight page.
- ✓ Cluster 2: Moderately successful campaigns with balanced goals and engagement. Main category was Film & Video, and they made it to the spotlight page.
- ✓ Cluster 3: Poorly performing campaigns with unrealistic goals (either too high or too low) and minimal interest. They did not have a video and were not promoted anywhere.
- ✓ Cluster 4: Overambitious campaigns with no engagement. Its characteristics are similar to Cluster 3, and they could potentially be merged due to both being very small clusters.

Here are two suggestions to help businesses improve their Kickstarter projects:

1. Set Realistic and Achievable Goals: Businesses should ensure their project goals are achievable within the scope of their audience and resources. Clear, measurable objectives are essential for generating interest and maintaining momentum.
2. Focus on Engagement and Backer Interest: Businesses should prioritize strategies that actively involve potential backers, such as offering incentives, building a strong community before and during the campaign and using multimedia interactive content.