ML Plan

Due: Tuesday, October 1st at 11:59 pm
Team: Karissa Dunkerley, Lucia Fang, Sen Feng

Problem Formulation

In this project, we want to focus on identifying the features that determine whether a project gets funded or not for individual poverty levels. Therefore, we will formulate the DonorsChoose project funding success prediction as a binary classification problem with the goal of predicting whether a project from a high/highest poverty area will be fully funded within the 4-month time frame. Our Target Variable is *funded_in_4_mths* (boolean). 1 means if the project was funded within 4 months, 0 otherwise. Given the diverse nature of projects and various factors influencing funding success, this problem requires segmenting projects based on their poverty levels and analyzing the unique characteristics within each segment to understand which features contribute most significantly to successful funding. Additionally, we will focus on data only from New York and Pennsylvania, due to the size of the original dataset.

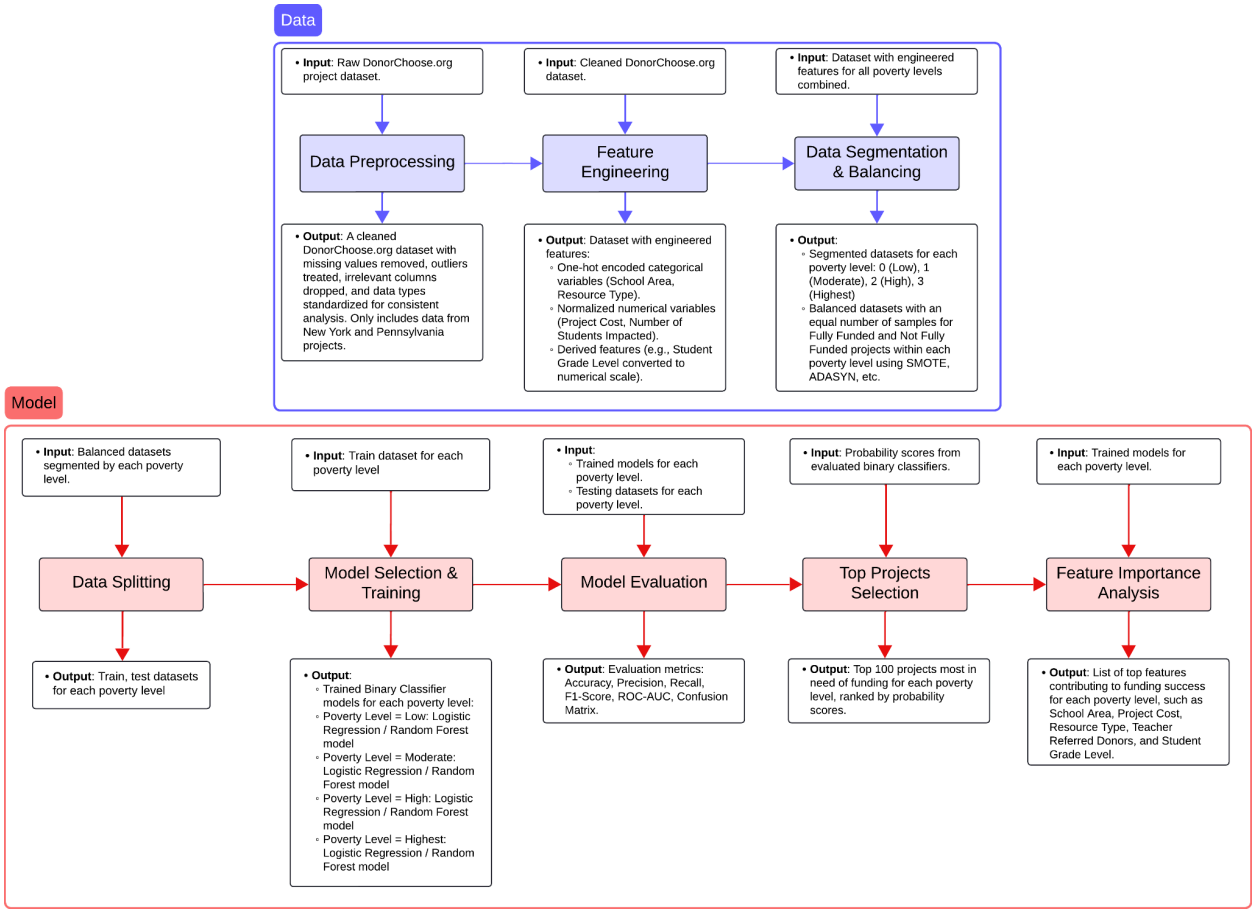
Features

Our initial set of features includes a mix of categorical and numerical variables extracted from the projects.csv and outcomes.csv datasets. These features are carefully selected to capture key characteristics of each project, the school's demographic context, and engagement metrics:

1. **Poverty Level** (*poverty_level*): A categorical variable with four possible values, corresponding to the level of poverty of the school where the project is being requested, based on the percentage of students on free or reduced lunch. The four values are highest (over 65% of students on free or reduced lunch), high (40-64%), moderate (10-39%), and low (0-9%). This variable is in the projects.csv dataset.
2. **School Area** (*school_metro*)**:** A categorical variable with three possible values, corresponding to the area that the project's school is in. The values are urban, rural, and suburban, and this variable is also in the projects.csv dataset.
3. **Teacher Referred Count** (*teacher_referred_count*): A quantitative variable and refers to the number of donors that were referred by the teachers. This variable is in the outcomes.csv dataset, which we can combine with the projects.csv dataset by joining on the project id, since there is only one row corresponding to each project in both projects.csv and outcomes.csv. Higher counts could indicate strong teacher engagement and influence project funding success.
4. **Grade Level** (*grade_level*): A categorical variable representing the range of grade levels associated with the project, with four possible values. The values are Grades PreK-2, Grades 3-5, Grades 6-8, and Grades 9-12. Projects targeting specific grade levels may have varying chances of receiving funds based on perceived impact. It comes from projects.csv.

5. **Project Cost** (*total_price_excluding_optional_support*): A quantitative variable representing the total cost of the project, excluding optional tips that donors can give to DonorsChoose.org while funding a project. This variable is expected to have a direct relationship with funding success. It comes from projects.csv.
6. **Primary Focus Subject** (*primary_focus_subject*): A categorical variable, representing the main subject (e.g., Math, Science, Literacy). This feature can help identify which subjects receive more funding support. It comes from projects.csv.
7. **Students Reached** (*students_reached*)**:** A quantitative variable representing the number of students that would benefit from the project if funded. Projects impacting more students might attract more donors. It comes from projects.csv.
8. **Resource Type** (*resource_type*)**:** A categorical variable representing the main type of resources requested (e.g., books, technology, supplies). This feature can provide insights into which resource requests resonate most with donors. It comes from projects.csv.

Diagram



Handling Missing Values

Because the majority of the missing values are not quantitative, we cannot really use imputation to mitigate the problem. If there are a lot of missing values in columns that we do not plan on using to create features, we will drop those columns. After dropping those columns, we will drop the rows that contain null values, since we have so much data that dropping columns with null values should not make a significant impact on how much data we have.

1. Projects Dataset (projects.csv):
   Dropped Columns: These features are either not relevant for our initial feature set or have a high proportion of missing values.
   - *secondary_focus_subject*
   - *secondary_focus_area*
   - *fulfillment_labor_materials*
   - *school_ncesid*

   Handling Remaining Missing Values: Rows with null values in important features like *total_price_excluding_optional_support* and students_reached will be dropped. Dropping these rows only affects about 6.55% of the rows, which is acceptable given the large size of the dataset.

2. Outcomes Dataset (outcomes.csv):
   Dropped Columns: These features are either not relevant for our initial feature set or have a high proportion of missing values.
   - *great_messages_proportion*
   - *donation_from_thoughtful_donor*
   - *one_non_teacher_referred_donor_giving_100_plus*
   - *three_or_more_non_teacher_referred_donors*
   - *at_least_1_green_donation*

   Handling Remaining Missing Values: Dropping rows with missing values in remaining features reduces the dataset size by approximately 15.24%, which is acceptable.

3. We do not plan on using the donors.csv dataset. In the future, for the rows missing location-related values, we will use other available location variables in the same row. To incorporate donor-related features into the projects.csv dataset, we would need to aggregate donor information at the project level, creating new features that capture donor characteristics and engagement patterns for each project.

Aggregation Strategies

1. **Temporal Aggregation**: Aggregate donation counts and amounts over specific time intervals (e.g., monthly, quarterly) to create features like *avg_donations_per_month* or *total_donations_last_3_months*. This will help capture donation trends that could influence funding success.
2. **Geographic Aggregation**: Group projects by state, city, or school district to calculate features such as *total_projects_in_city* or *avg_funding_per_state*. This could reveal patterns in project funding success across different regions.
3. **Donor Aggregation**: Create project-level features based on donor attributes such as *avg_donor_contribution*, *number_of_repeat_donors*, or *donor_state*. These features can provide insights into the impact of donor engagement on project success.

Class Balancing Using SMOTE or ADASYN

To address class imbalance within each poverty level, we will use either SMOTE (Synthetic Minority Over-sampling Technique) or ADASYN (Adaptive Synthetic Sampling) to create balanced datasets. SMOTE generates synthetic samples for the underrepresented class (not funded) by interpolating existing samples, ensuring that the model does not become biased towards the majority class. ADASYN, on the other hand, adaptively generates more synthetic samples for the minority class data points that are harder to classify, focusing on regions where the decision boundary is less clear. We will experiment with both techniques to determine which approach best enhances model performance and generalizability across different poverty levels.

Metrics

1. Accuracy: To measure the proportion of correctly classified projects for each poverty level. However, accuracy might not be reliable due to class imbalance.
2. Precision and Recall: To evaluate how well the model identifies fully funded projects (precision) and how many actual funded projects are correctly identified (recall). These metrics are crucial for assessing model performance on minority classes.
3. F1 Score: A harmonic mean of precision and recall, which will help evaluate the trade-off between identifying fully funded projects and minimizing false positives.
4. Confusion Matrix: To visualize model performance by comparing true positives, true negatives, false positives, and false negatives for each poverty level.
5. ROC-AUC: To assess the overall discriminatory power of the model. A high ROC-AUC indicates that the model is effectively distinguishing between funded and not-funded projects.

Biases, Trade-offs, and Baselines

One possible bias could be introduced from dropping all rows with null values because there could be some underlying reason why those rows all have null values. For example, there might be some projects from a specific metropolitan area that did not have a metro area listed that happened to be of a certain type or have a specific subject area, which could bias the model against those projects. The *primary_focus_subject* feature might also introduce bias if certain subjects historically receive more funding and could potentially disadvantage important but less popular subjects. This also applies to *resource_types* where certain resources might be historically favored by donors.

Regarding tradeoffs, the current feature set is relatively simple which helps interpretability but might miss some nuanced factors affecting funding success. There's a trade-off between keeping the model simple and potentially including more complex features for better prediction. With features like poverty_level and school_metro, there might be a trade-off between optimizing for overall accuracy and ensuring fair predictions across different subgroups.

By prioritizing recall, we aim to identify as many potentially unfunded projects as possible. This approach ensures we don't miss projects that need funding support but may lead to more false positives. We'll use precision-recall curves to find an optimal threshold that balances metrics based on specific needs and constraints.

Our baseline will be a random classifier that assigns projects to funded/not funded classes with equal probability. This will serve as a comparison point to gauge the performance of our classification models.