# A Prediction System for Movie Revenue

CSCI 5502 Data Mining Final Project

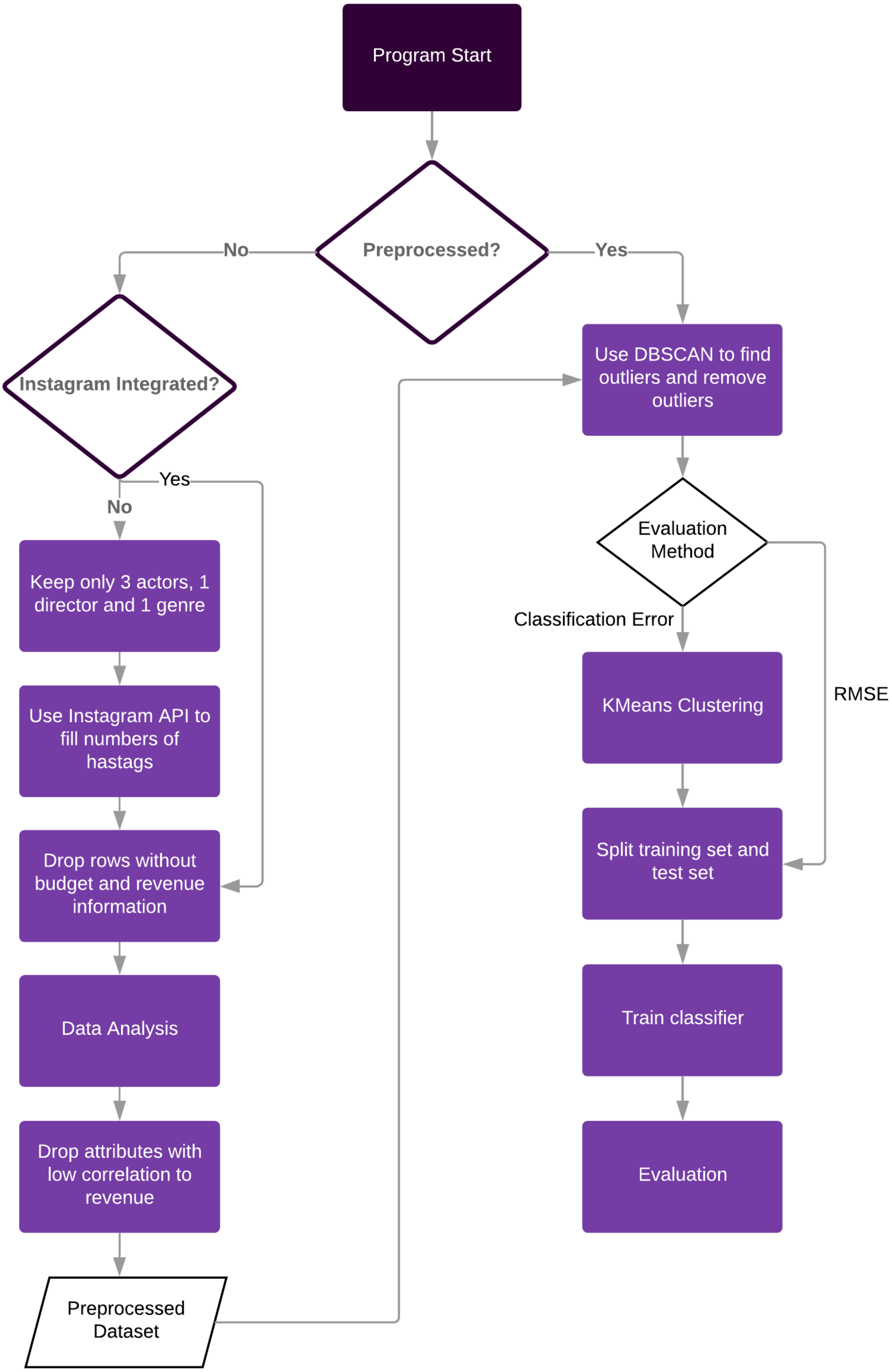
## Structure

- **datasets**: store all csv files generated from original TMDb datasets
- **notebook**: Jupyter notebooks (mainly used for analyzing correlations between attributes)
- **src**: source Python files of the core system
  - **classification.py**: implement classification approaches used in this project
  - **clustering.py**: implement clustering approaches used in this project
  - **DBSCAN_tuning.py**: a program can automatically tune parameters of DBSCAN algorithm
  - **preprocessing.py**: implement functions for preprocessing conveniently
  - **movie_revenue_predictor.py**: the main program
- **src/utilities**: program not relate to core functions
  - **instagram_data.py**: program to get Instagram hashtags

## Environment & Dependency

- Python 3
- Pandas
- scikit-learn
- Matplot
- TMDb datasets in the directory `datasets`

## Diagram


Program Start
Preprocessed?
No
Yes
Instagram Integrated?
Yes
No
Keep only 3 actors, 1 director and 1 genre
Use Instagram API to fill numbers of hastags
Drop rows without budget and revenue information
Data Analysis
Drop attributes with low correlation to revenue
Preprocessed Dataset
Use DBSCAN to find outliers and remove outliers
Evaluation Method
Classification Error
RMSE
KMeans Clustering
Split training set and test set
Train classifier
Evaluation

## How to run the system

The main program is `movie_revenue_predictor.py`. You need to execute this program in the directory `Movie_Revenue_Predictor`. Then, the program can be executed by the instruction `python3 src/movie_revenue_predictor.py`. Before execution, you can modify the parameters mentioned below.

# Parameters

## movie_revenue_predictor.py

There are paramters can be changed to conduct different tests.

```
classification_method = 1 # 0: single classifier 1: boosting
plotting = False # plotting classification result or not
evaluation_method = 0 # 0: classification error 1: RMSE
test_times = 10 # how many rounds of tests
```

## classification.py & clustering.py

Users can tune parameters of classification and clustering methos in different classifiers. To get more detailed information, please take a look at scikit-learn.

# Datasets

Make sure that the TMDb datasets are in the directory `datasets`.