DESAUTELS | McGill

INSY 662: Data Mining and Visualization (Fall 2019)

Individual Project – Worth 15%

This project is to be done individually. All the coding involves in this project must be in Python (in .py format). You can only use techniques/algorithms covered in this class. However, you are allowed to use parameters, attributes, etc. that are not covered in this class as long as they belong to the techniques/algorithms covered in the class. For example, Support Vector Classification is the algorithm covered in the class but kernel=sigmoid is the parameter that is not covered in the class. You are allowed to use kernel=sigmoid with SVC. Meanwhile, gradient boosting is the algorithm that is not covered in the class, so you are not allowed to use gradient boosting in this project.

Your task in this assignment is to:

1. Develop a regression model (i.e., a supervised-learning model where the target variable is a continuous variable) to predict the value of the variable "usd_pledged." After you obtain the final model, explain the model and justify the predictors you include/exclude.

2. Develop a classification model (i.e., a supervised-learning model where the target variable is a categorical variable) to predict whether the variable "state" will take the value "successful" or "failure." After you obtain the final model, explain the model and justify the predictors you include/exclude.

For both tasks, only include observations where the variable "state" takes the value "successful" or "failure" (i.e., all other observations should be dropped). Note that you will be graded based on both the performance of the model and the explanations/justifications you provide. You also need to clearly articulate how realistic or useful your model would be in a business context.

There are two deliverables for this assignment:

1) Summary Report

- The report must be submitted in pdf format.
- The report must not exceed 5 double-spaced pages, **including everything**. Page margins must measure 1" around. Please use a 12-point Times New Roman font.
- Name the file as follows: "Lastname_Firstname_IndividualProject"
- The report must contain:
  - The summary statistics
  - The explanations/justifications of each model along with the results. You may submit only one model per task.
- The report is due by Thursday, November 14 at 11:59pm.

2) Python Code

Along with the report, please also submit Python code (in .py format) that you use to develop your report. The code should be complete with informative comments and able to run fully without any errors or modifications (besides the file path).

Data Description

The dataset in this project is scraped from Kickstarter, which is a popular crowdfunding platform. There are 45 variables in total. The table below contains a short description of each variable.

Column Name	Description
project_id	Unique identifier for projects
name	Project
goal	Goal amount requested by the project
pledged	Amount pledged at time of data scrape
state	Status of the project (successful, failed, etc)
disable_communication	If communication with project owners was disabled
country	Origin country of project
currency	Currency of origin country
deadline	End date of project funding period
state_changed_at	date and time the project state was modified to current state
created_at	Date and time project was created
launched_at	Date and time project was launched
staff_pick	If the project was a staff pick
backers_count	Number of backers
static_usd_rate	The conversion rate of project country currency to USD
usd_pledged	Amount pledged in USD
category	Category of project
spotlight	If the project was featured on kickstarter spotlight page
name_len	Length of project name in word count
name_len_clean	Length of project name in word count sans non- key words (such as “for” “and” etc.)
blurb_len_clean	Length of project blurb in word count sans non- key words
deadline_weekday	Weekday of deadline date
state_changed_at_weekday	Weekday of state change
created_at_Weekday	Weekday of creation date
launched_at_weekday	Weekday of launch date
deadline_month	Month of the project deadline
deadline_day	Day of the project deadline
deadline_yr	Year of the project deadline
deadline_hr	Hour of project deadline
state_changed_at_month	Month of latest state change

state_changed_at _day	Day of latest state change
state_changed_at _yr	Year of latest state change
state_changed_at _hr	Hour of latest state change
created_at_month	Month of creation date
created_at _day	Day of creation change
created_at _yr	Year of creation change
created_at _hr	Hour of creation change
launched_at_month	Month of launch date
launched _at _day	Day of launch date
launched _at _yr	Year of launch date
launched_at _hr	Hour of launch date
create_to_launch_days	Number of days between project creation and the public launch date
launch_to_deadling_days	Number of days between the launch date and the deadline
launch_to_state_change_days	Number of days between launch date to the latest status change

Hint

1) Make sure to understand the predictors that you plan to include in your model. If the information in the Data Description section above is not clear, please visit kickstarter.com to obtain additional information.

2) As discussed in the class, it is extremely important to include predictors that can be realized before the prediction starts only.

3) If your model performs too well even on the test dataset (e.g., your classification accuracy is 1), you either accidentally include the target variable as a predictor in your model or one of your predictors is perfectly correlated with the target variable.

Structuring your Code

To facilitate the grading process, please write a separate code that allows the grader to apply your regression and classification model based on the "grading" dataset. Specifically, please write a separate code that reads an input file named "Kickstarter-Grading.xlsx" (this file has exactly the same structure as "Kickstarter.xlsx", which is the dataset of this project). Should you do any preprocessing, the code should also be applied this to the grading dataset as well. Then, generate the MSE (for regression model) and accuracy score (for classification model) with this new dataset, using models that were developed based on the original data you were given. This script should essentially allow the grader to test the performance of your model on new data. Be sure that your script does NOT train a new model with this new data, and that the model used to generate MSE and accuracy scores is the one developed based on your original data. The penalty for failing to do this is up to 10% of your total grade for the individual project.

For illustration, the sample code and the sample grading dataset are provided. The content of the real grading dataset is different than that of the one provided. Do NOT develop your model based on this dataset.