College Applications

CS 556

1. Dataset

This dataset documents the test scores, school rankings, and application strengths of a series of college applicants, and correlates each with the probability that application is accepted to the school of their choice. Each of these features is provided as a continuous, qualitative numerical value, which means that we have the ability to process this data to review applications and learn what makes an application strong or not.

Every application in the dataset contains the following features:

- Graduate Record Examinations (‘GRE Score‘): A score (out of 340) on the GREs.
- Test of English as a Foreign Language (‘TOEFL Score‘): A score (out of 120) on the TOEFL.
- University Rating (‘University Rating‘): A rank from 1 to 5 (with 5 being the best) of the university this entry describes application to.
- Statement of Purpose (‘SOP‘): A rank from 1 to 5 (in increments of 0.5) of the Statement of Purpose provided as part of the application.
- Letter of Recommendation (‘LOR‘): A rank from 1 to 5 (in increments of 0.5) of the Letter of Recommendation provided as part of the application.
- Undergraduate GPA (‘CGPA‘): Undergraduate college GPA, scaled here to be between 0 and 10, with 10 being the highest.
- Research Experience (‘Research‘): A boolean ‘0‘ or ‘1‘ value indicating whether the applicant has research experience.
- Chance of Admission (‘Chance of Admit‘): The chance (as a decimal probability between 0 and 1) that the application described in the previous data points will be accepted by the target university.

2. Your Task

Your task is to train machine learning models which can predict, based on the given features, the chances of application acceptance for a given student. First, we’ll need to load the provided dataset into a Jupyter Notebook from the provided CSV file. The names of the columns will correspond to those given above.

Consider at least three features from the dataset and note their distributions, any extraneous/outlying values you might need to consider, and whether they seem correlated (on cursory look) to the chances of being accepted. Then, utilizing **only** the tools and processes outlined in this course, you’ll train a linear regression on 80% of the dataset (training set) to predict the chances of admission.

Test your model on the remaining 20% of your data and report the mean square error of your model.

Next, use PCA to reduce the dimensionality of the dataset to two and then train a new model to predict the chances of admission using the new resulting dimensions as features. Generate a scatter plot showing the date points in blue and the decision boundary in black. Test your new model and report the mean square error. Compare the performance of this new model with the previous one. Which is more effective, and why?

3. Submission

A project submission consists of four files:

1. PDF of exactly one-page reporting:
   (a) CWID / name of student
   (b) Mean Square Error for training and test sets for both models
   (c) salient project features (describe your design choices)
2. CSV files of predictions chance of admission for the test set for both models.
3. Working Jupyter Notebook that produces the CSV.
4. Jupyter Notebook downloaded as a HTML file.

4. Evaluation

Evaluation will be done based on:

- Mean Square Error
- Techniques
- Code quality

5. Code of conduct

All scripts/notebooks will be checked (1) against each other (2) against an online database, using plagiarism detection tools. Any case of plagiarism is a serious incident and is susceptible to be reported to the competent authorities of Stevens. Plagiarism is intended as sharing a large fraction of the code that performs critical operations. Plagiarism does not include sharing small isolated pieces of code that perform routine tasks (e.g. input/output, or basic normalization/imputation) if in aggregate they represent a small portion of the code.