Machine Learning & Data Mining
Miniproject 1

Caltech CS/CNS/EE 155
Released January 28^{th}, 2017

Introduction

- **Group members**
  Vaibhav Anand
  Nikhil Gupta
  Michael Hashe

- **Team name**
  The Breakfast Club

- **Division of labour**

Overview

- **Models and techniques tried**
  - **Bullet:** Bullet text.
- **Work timeline**
  - **Bullet:** Bullet text.

Approach

- **Data processing and manipulation**
  - **Bullet:** Bullet text.
- **Details of models and techniques**
  - **Bullet:** Bullet text.

Model Selection

- **Scoring**

- **Validation and Test**

Conclusion

- **Discoveries**
  The main discovery from this competition was the difficulty of fitting a "hard" and imbalanaced dataset, and in particular the challenges accompanying creating a model both sufficiently able to fit a training dataset (2008 data) and sufficiently general to adapt to an unknown and distinct new dataset (2012 data). These challenges will be analyzed in more depth in the section below.

  In terms of models used, we experimented with variants of common models examined in class. In particular, we utilized variations of trees aside from the default CART tree and various linear models. While our best performing models remained standard neural networks (and, on the test dataset, random forests), we certainly learned more about the variants of these models currently in use.

  In examining the occasionally massive changes in the leaderboard between the first and second parts of the competition, and in examining the differing performances of our own models on these two datasets, we also discovered the dangers of overfitting and the propensities of various models for doing so. In particular, we found (after submission) that our more "general" models (i.e., collections of networks and forests) tended to perform reasonably well on the new dataset, whereas models more prone to overfitting (smaller forests and networks, linear models) tended to suffer more. Considering the large drops experienced by many teams initially above us on the leaderboard, it is our belief that this "overfitting the leaderboard" was in fact a fairly general problem.

- **Challenges**
  As mentioned above, the main problem experienced in the competition was the dataset. In particular, we found that the dataset was lopsided (75% of those polled voted), rather different from the test set (in which 65% of those polled voted), and very noisy (i.e., even the best/most overfitted models were unable to break 80% classification accuracy). In dealing with such a dataset, we found that most models performed within a percent or two of each other, which made distinguishing the best models somewhat difficult. Further, it meant that it was very easy to fall into a trap of overfitting (i.e., gridsearching parameters, submitting an excessive number of models, and falsely concluding that the models that best fit the data did so because they were the best models). We managed to avoid this reasonably well, although we could have improved on our part 2 score had we selected better. Were we to do this competition again, we would have put more thought into how to best select models.

  Within the dataset, we also found that many of the parameters were either repeated, arbitrary (i.e., identification numbers), or answered for only a very small number of responders (i.e., several questions focused very specifically on certain subsets of the population, and therefore had response rates in the hundreds). We curated the dataset and removed many of these parameters from consideration, although determining whether or not a parameter should be included was not always a clear decision. Were we to redo this competition, we would have curated the dataset in a more principled (and, ideally, automated) manner, through some manner of cross-validation on model performance with various parameters removed. We would also likely have removed even more parameters; it is

our belief that this might help with the generalization accuracy of the model.

TENSORFLOW

We additionally encountered problems with resource limitations in training large models. While most of our models trained in reasonable time, we found that some (in particular, certain variants of random forests and SVMs) were slow enough to be infeasible.

- **Concluding Remarks**