Machine Learning & Data Mining
Miniproject 1

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

Introduction

- **Group members**
  Sarthak Sahu and Andrew Kang
- **Team name**
  Supervised Learning
- **Division of labour**
  Sarthak implemented deep neural networks and conditional random fields, running them in parallel to get a 100% accuracy on both the train and test set. He also invented a novel algorithm, coined the "Sahu Algorithm," for which he will be receiving a Turing Award.

  Andrew slept.

Overview

- **Models and techniques tried**
  - **Classifier C1:** We used sklearn.ClassifierC1, resulting in a score of 80%.
  - **Technique T1:** We implemented Technique T1 on the dataset. We were motivated by high variance seen using classifier C1. As a result, we saw a 5% increase in our scores.
  - **Custom method M1:** We implemented our own custom method M1() to attempt to lower bias.
- **Work timeline**
  - **Week 1:** We did some preliminary tests to see what kinds of methods would work best. We also preprocessed the data.
  - **Week 2:** We used classifiers C1, C2, and C3, achieving the best results with model C2. We used model C2 for the remainder of the week to obtain the highest scores.

Approach

- **Data processing and manipulation**
  - We used method M1() to normalize our data. We chose not to use features F1 and F2. We transformed the data using NumPy's method M2().
- **Details of models and techniques**
  - **Classifier C1:** We decided to use classifier C1 as it was suitable for the probabilistic nature of the data. We experimented with parameters P1, P2, and P3 in the model. Ranging the parameters between [1, 100] in increments of 1, we got scores in the range of 60-70%. We used the built-in methods from scikit-learn in our implementation. The advantage of using this model was that it was simple. The disadvantage of using this model was that it did not score very well. A figure is included below for reference.

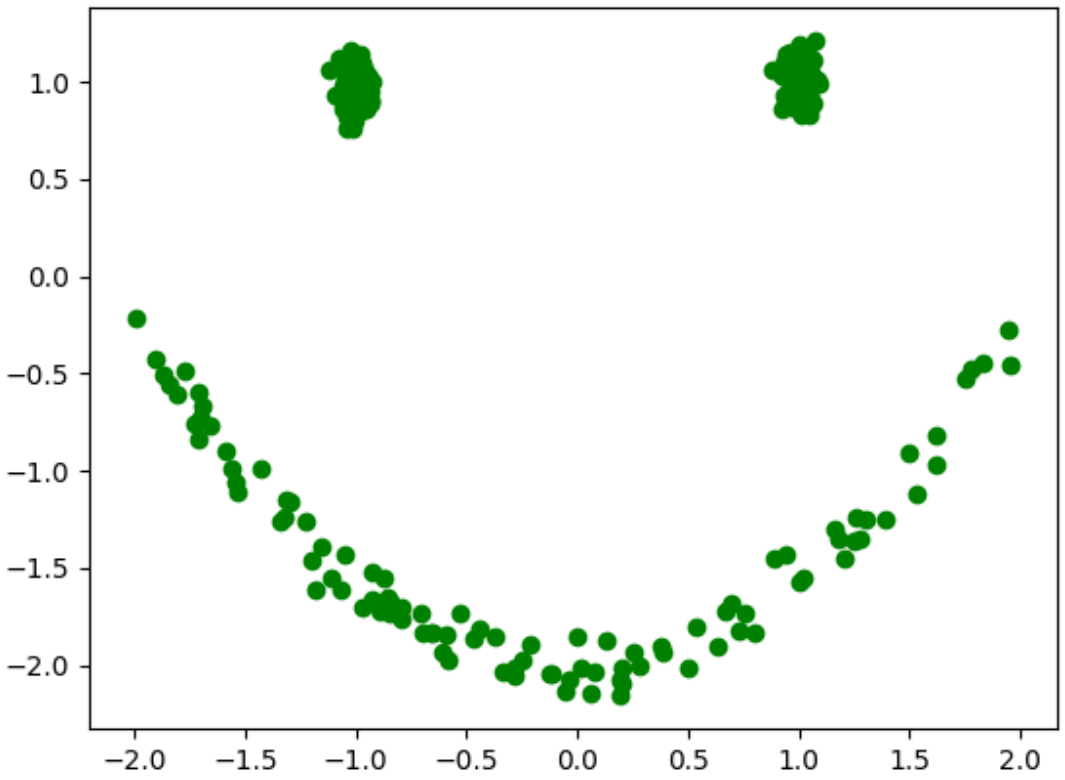

Figure 1: Visualization of classifier C1. The model is happy.

Model Selection

- **Scoring**
  We used scoring function S1, with the lowest scores being around 60% with model M1 and the highest scores being around 80% with model M2.
- **Validation and Test**
  We used cross-validation to choose parameter P1 for classifier C1. We chose classifier C2 based on the leaderboard scores.

Conclusion

- **Discoveries**
  We learned that classifier C1 can be used in datasets with many features. We learned that classifier C2 is suitable for a problem of this nature.

- **Challenges**
  We found that normalizing the data did not help us achieve better scores. We struggled to work with the data due to its complex features.
- **Concluding Remarks**
  We got a successful solution and are happy with the results (see figure 1)!