Project 2 - Multiclass Classification

In this project you will explore some techniques in solving the supervised learning task of multiclass classification. It is important to realize that understanding an algorithm or technique requires understanding how it behaves under a variety of circumstances. You will go through the process of choosing and exploring two multiclass classification datasets, tuning the algorithms you have learned about, writing a thorough analysis of your findings, and presenting your findings. The most crucial part of this assignment is the analysis and your ability to explain and justify your results.

I. Choosing Datasets

The first task in this assignment is choosing two interesting multiclass classification datasets. That is, the variable you're trying to predict must be categorical with at least three possible values. The features can be of any type, and it is recommended that you choose datasets with diverse feature sets. I don't care where you get the data from. You can download some, take some from your own research, or make some up on your own. What I do care about is that the datasets must be interesting. They should contain a decent amount of features and a sufficiently large amount of examples. Do not choose an "easy" dataset, however don't go crazy either trying to find the perfect one. Your two datasets should also differ in some way such that you can compare and contrast your results between the two. You should also be following standard machine learning practice by splitting your dataset into training and testing, and only touching the testing dataset at the very end when you are ready to report results. (Cross validation is highly recommended).

II. Coding (10%)

After choosing your datasets you will now be tasked with writing code to apply the machine learning algorithms you have learned about. Your code must be written in python, but you may use any libraries that have already implemented the machine learning algorithms (e.g scikit-learn). You are not expected to code the algorithms from scratch, and in fact I would highly discourage it. What you may not do is copy code from the internet. Below are the algorithms that you are required to "implement"

- Naive Bayes
- Support Vector Machine
- Neural Network

Your code does not have to be pretty or well written. However, it must be written in python and I must be able to run one script (main.py) that will produce all the results and figures in your report.

III. Report (80%)

You will then produce a report describing and analyzing your methods and results. Here you will describe the datasets you have chosen and why they are interesting. You will then provide an analysis on how the different machine learning algorithms performed on each dataset. The report must be limited to **10 pages maximum**. Below is an outline of what I expect in your report.

- **Introduction:** Here you will describe what problem you are trying to solve and why it is important. Briefly introduce your dataset and the experiments/analyses you will be including in the report.

- **Datasets:** A description of your two datasets and why you feel that they are interesting. What kind of features are in your dataset? How do they differ? How are they similar? What kind of pre-processing did you have to perform (missing data, scaling, etc..)? What are you hypotheses abut how your datasets will perform and why?

- **Methods:** Discuss how you partitioned your datasets and the methods you used in your analysis. What metrics will you use and why?

- **Hyperparameter Tuning:** For each dataset and for each algorithm you should be performing hyperparameter tuning. Produce hyperparameter tuning curves and explain what you observed. This is perhaps the most important part of the report for you to demonstrate your understanding each algorithm.

- **Other Analyses:** You should be exploring each algorithm even further. Ideas here include looking at how training dataset size affects performance, or training/testing timing analysis. You may choose other types of analyses that can further demonstrate your understanding.

- **Results:** After finding the optimal hyperparameters provide a table of final results of how each algorithm performed. This should include both training and testing scores.

- **Discussion:** Here you will discuss your results. Try to explain what you observed. Compare and contrast results between datasets. Compare and contrast between algorithms.

- **Conclusions**: What are the main conclusions from your report? What would you have done differently with more time/resources? What are some ideas you'd like to explore further.

- **References**: List all sources used.

You are NOT being graded on how well the algorithms perform on your datasets. What is most important is WHY? You should be explaining and justifying all of your figures and results, and demonstrating that you understand the intricate details of the machine learning process, and the machine leaning algorithms you are using.

IV. Presentation (10%)

Finally you will give a 5-10 minute presentation of your results. In this presentation you will describe your datasets, your methods, and and any interesting results you found!

What to turn in?

Below is a list of items you will be required to turn in via canvas. Please make sure all documents are named as described bellow.

- **report.pdf** - Your maximum 10 page report in pdf format. Do not use super tiny or large font. No specific formatting is required but use common sense.

- **presentation.pptx** or **presentation.key** - Your presentation slides either in a powerpoint or keynote document.

- **code.zip** - A zip file with all of the code you have written. Within the folder there should a file called **README.txt** that contains instructions on how to run your code, and a python file called **main.py** that will produce all figures and plots in your report/presentation. I should be able to reproduce your results easily.

- **data.zip** - A zip file that contains the two datasets you have chosen.

- **work_distribution.pdf** - A pdf containing 1-2 paragraphs describing how you split up the work between your group.

Grading

You are being scored on your analysis more than anything else. Roughly speaking, implementing everything and getting it to run is worth very little for this assignment. Of course though, analysis without proof of working code makes the analysis suspect. The key thing is that your explanations should be both thorough and concise, and your analysis should prove to me you have a deep understanding of the machine learning process and the machine learning algorithms you are using.