CPSC 340 Machine Learning Take-Home Final Exam (Spring 2020)

Question 2 [30/100 points]

This part of the final is a group project that takes place on Kaggle, which can be accessed from the following url: `https://www.kaggle.com/c/CPSC340FinalPart2`. You can sign up for a new account or use an existing one; however, note that the Kaggle servers may be in the US, so bear this in mind. We recommend that for data protection purposes you use a non-identifiable (but ideally hilarious) team name. You will link your group members to your team name in your submission document.

Methods that you have learned over the semester are the foundation for solving this task, but they may not be quite enough to solve it well so we recommend that you do some additional research on new methods (as one extremely relevant suggestion, consider looking into transfer learning). Your mark for this part of the final will be based on the score from Kaggle for your test set predictions, a written report that explains your findings, and your code. **Your report should follow the format outlined in `final.tex`.**

The Kaggle competition includes code that will load a dataset of lung X-rays from patients who either have COVID-19 or not (either nothing or another form of pneumonia) if you stored the dataset in a directory called `./data/`. Unlike question 1, you **are** allowed to use built-in models from libaries such as PyTorch or scikit-learn.

Bundle your code along with a `.pdf` generated from the filled in LaTeXreport into a `.zip` file and submit it to Gradescope. Again, marks may be taken off for very messy or hard to read code, so make sure to use descriptive variable names and include comments where appropriate.

It is OK to fail to solve this task "satisfactorily." If your approach is sound and the effort is appropriately high, you will still receive extra credit even if you are unsuccessful. Trying very much counts here.

Skeleton for Question 2 Answer

Please keep the total length of your entire question 2 response to less than 2 pages. Nothing beyond three pages will be read.

1 Team

Team Members	*Yuetong Liu, Xuerong Wang*
Kaggle Team Name	*Lazy*

2 Introduction

A few sentences describing the COVID-19 X-ray classification problem and the problems with it.

As COVID-19 becoming a global pandemic scientists and epidemiologists are looking for effective way to determine whether or not a patient is infected with COVID-19. The main purpose of this project is to create a classifier capable of predicting whether the subject in the x-ray is infected with COVID-19 or is not infected with COVID-19. The dataset contains images of chest x-rays in which the patient either has or does not have COVID-19. A problem with this is that the X-ray should have some key features to determine whether a patient is affected but we don't know what it is, so we have compare the whole image instead of using the most important part.

3 Method

Several paragraphs describing the approach you took to solving the problem. Highlight in particular how you worked around the small training data problem. Transfer learning is likely something you will want to read about.

There are only 70 data entries in the training dataset. The small dataset could make the classification model to over-fit the data, and outliers might cause high variance. Our design aims to avoid these problems.
Before fitting data into models, we decided to conduct data preprocessing to standardize features. If a feature has a variance that is orders of magnitude larger than others, it might dominate the objective function and make the estimator unable to learn from other features correctly as expected.
Moreover, simple method (such as decision tress with a depth of 4) is preferred during the sample selection. Otherwise, complex methods with too many parameters can over-fit the data. Also, we don't plan to conduct many comparisons between different methods, since the performance of the training set might not be a good estimator for out of range performance.

4 Experiments

Several paragraphs describing the experiments you ran in the process of developing your Kaggle competition final entry.

During data preprocessing, we standardized features by removing the mean and scaling to unit variance. The standard score of a sample x is calculated as:

$z = \frac{x-u}{s}$
where u is the mean of the training samples, and s is the standard deviation of the training samples.

We decided to use KNN which is very simple to understand and equally easy to implement. Moreover, it is a non-parametric algorithm which means there are assumptions to be met to implement KNN and we need enough space to store all training data.
Finally, we used 5-fold cross validation to decide the most optimal parameters for our model.

5 Results

Model	Kaggle Score
KNN	*0.82352*

6 Conclusion

Several paragraphs describing what you learned in attempting to solve this problem, why your team is ranked where it is on the leader board, how you might have changed the problem to make its solution more valuable, etc.
Despite of the advantage mentioned above, KNN shows several disadvantages. The curse of dimensionality is significant, since this is a high-dimensional data sets. Therefore, we could consider feature selections for further analysis.