Homework 1: ML 80629A (FALL 2021)

Instructions:

- Please include your name and HEC ID with submission.
- The homework is due by 11:59pm on the due date.
- The homework is worth 20% of the course's final grade.
- Assignments are to be done individually.
- All code used to arrive at answers is submitted along with answers. You can convert your jupyter notebook or colab to pdf and upload it.

1 ML Fundamentals (20pt)

1. (4pt) Explain the difference between the training error and the generalization error. Make sure to describe how to evaluate the generalization error of a model in practice including pitfalls of this approach.
2. (4pt) To increase the size of your training set you first train a model and then use it to obtain labels on an unlabelled test set. You then retrain the model with the data from your train set as well as the data from your test set.

   Would you expect that your final model would obtain a lower validation error?
3. (4pt) Suggest a method for regularization a K-NN model. Hint: think of what regularization accomplishes in terms of the bias/variance trade off.
4. (4pt) Recall the task of document classification where documents must be classified based on their content. If the documents are encoded in a bag-of-words format, could you use K-NN to classify them? If so, describe a distance function that might be sensible to use. Otherwise, explain why not.
5. (4pt) Describe both the advantages and the disadvantages of using a larger K when doing K-fold cross validation.

2 Regression (15pt)

- Let's explore California housing dataset. This dataset was obtained from the StatLib repository. The target variable is the median house value for California districts, expressed in hundreds of thousands of dollars ($100,000). You can load the data from scikit-learn. You can use `.DESCR` to gather more info about this dataset.

  1. (2pt) Perform statistical analysis by using `.describe()` on the data, what do you notice about the attribute values?
  2. (2pt) Look at the distribution of the attribute values by plotting their histograms. Notice anything interesting?
  3. (3pt) Perform 10-fold cross-validation (with `shuffle=True` and `random_state=20160202`) with a LinearRegression model. Report the mean squared error (MSE) on the validation set (averaged across folds).

- Two very popular options in Linear Regression are the Lasso method and the Ridge Regression. These models are implemented in sklearn: Lasso and Ridge.

  Hint: If you'd like to understand Ridge Regression better, you might want to look at this example.

  4. (2pt) What one word captures what Lasso and Ridge do?
  5. (3pt) Perform 10-fold CV with normal LinearRegression, Lasso, and Ridge. Compare the attribute weights ($coef$) of each of these methods. What do you observe?
  6. (3pt) Report the average MSE for each method on the validation set across folds. How do these variants of linear regression perform compared to each other?

Classification (65 pt)

In this exercise, you will implement Naive Bayes classifier and neural network models to carry out a text classification task. You will implement various feature representations, i.e., BoW and TF-IDF and then you will do performance analysis of NNs and NB with different hyperparameter settings.

The data to download is movie reviews dataset. In each file (train, validation, test), each line consists of a single review text followed by a number indicating the sentiment : 0 for negative and 1 for positive. You would have to split the these data into input texts and corresponding labels.Hint: you can use the function `split('\t')`, pandas allows you to easily use this data.

3 Feature Representations (5pt)

Transform the input text data into the following feature representations to train some models. You can use scikit-learn tools to do this. To reduce the time for training, please set `max-features = 10000` to build a vocabulary that only consider the 10k top max features ordered by term frequency across the corpus. Do not change the other parameters.

For this question, we ask you for the few lines of code from sklearn that you used to encode (and only encode) the training, validation and test data into (use the default settings):

1. (3pt) Bag-of-Words features (BoW)
2. (2pt) TF-IDF features

4 Naive Bayes (15pt)

1. (5pt) Perform the sentiment classification task with these 2 different types of features. Report which NB you used and what is the validation set accuracy in each case.
2. (10pt) Based on the classification of the model, report 5 words from the texts in the train dataset that can be inferred as *positive* and 5 words that are *negative*.

5 Neural Network (30pt)

After all the required imports in your code be sure to set the `random_state=12345`.

1. (12pt) From the BoW and TF-IDF features from above, you are going to train different neural network models. Use the option `early_stopping=True` and train the networks by trying all the following hyperparameter combinations:

   - dimension of a first hidden layer: [4,8,16]
   - dimension of a second hidden layer: [0,4,8]
   - learning rate (after fixing the above three hyperparameters): [0.1, 0.01, 0.001]
   - L2 (*penalty*): [0.001, 0.01, 0.1]

   What is the best combination of hyperparameters and what is the performance on the resulting model validation set?
2. (16pt) What did you observe with the impact of different hyperparameters on the model accuracy? Write a brief recommendation to follow for someone trying to find the best hyprparameters for their model.
3. (2 pt) Can you recommend any other hyperparameters to tune?

6 Comparison (15pt)

1. (4pt) Determine the performance of the simplest baseline, i.e., majority voting, for this task.

2. (5pt) For each case of the feature representation (BoW and TF-IDF), report the best test performance accuracies of the Naive Bayes classifier and Neural Network that you observe.

3. (3pt) Based on these accuracy values, which is the best feature representation for each type of model that we have considered? Can you think of why this is the case?

4. (3pt) Based on these accuracy values, which is the best performing model? Explain briefly why it is the case.