CS589 Machine Learning - Fall 2020

Homework 3

Due: October 9, 11:59 pm

Getting Started: You should complete the assignment using your own installation of Python. Download the assignment archive from Moodle and unzip the file. This will create the directory structure as shown below. You will write your code under the Submission/Code directory. Make sure to put the deliverables (explained below) into the respective directories.

```
HW01
--- Data
    |-- Breast Cancer Wisconsin (Diagnostic) Dataset
    |-- Gene Expression Dataset
--- Submission
    |--Code
    |--Figures
    |--Predictions
```

If you are stuck on a question consider attending the instructors' office hours.

Data Sets:

1. The Breast Cancer Wisconsin (Diagnostic) Dataset contains 30 features computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. The dataset contains 569 samples which are split into 419 training samples, and 150 testing samples. The task is to classify the diagnosis of the breast tissues to be malignant (label 1) or benign (label 0).

2. Gene expression dataset contains pre-processed data to identify small round blue cell tumors (SRBCTs) in children. The dataset contains 63 training samples with 300 gene features and 20 test samples with 300 gene features. The task is to classify the small round blue cell tumors (SRBCTs) into four categories using gene expression profiles.

Dataset	Training Cases	Test Cases	Dimensionality	Output
Breast Cancer Dataset	419	150	30	binary class labels {0,1}
Gene Expression Dataset	63	20	300	4-class labels {0,1,2,3}

Deliverables: This assignment has two types of deliverables: a report and code files

- **Report:** The solution report will give your answers to the homework questions (listed below). The maximum length of the report is 5 pages in 11 point font, including all figures and tables. You can use any software to create your report, but your report must be submitted in PDF format.
- **Code:** The second deliverable is the code that you wrote to answer the questions, which will involve training classifiers and making predictions on held-out test data. Your code must be Python 3.6 (no iPython notebooks, other formats or code from other versions). You may create any additional source files to perform data analysis. However, you should aim to write your code so that it is possible

to re-produce all of your experimental results exactly by running *python run_me.py* file from the Submissions/Code directory. Remember to comment your code. Points will be deducted from your assignment grade if your code is difficult to reproduce!

Submitting Solutions: When you complete the assignment, place your final code in Submission/Code. If you used Python to generate plots then place them in Submission/Figures. Finally, create a zip file called 'Submission.zip' from your 'Submission' directory only (do not include 'Data' directory). Only .zip files will be accepted for grading code (not .tar, .rar, .gz, etc.). You will upload your .zip file of code and your pdf report to Gradescope for grading. Further instructions for using Gradescope will be provided on Piazza and discussed in class.

Academic Honesty Statement: Copying solutions from external sources (books, web pages, etc.) or other students is considered cheating. Sharing your solutions with other students is considered cheating. Posting your code to public repositories like GitHub is also considered cheating. Any detected cheating will result in a grade of 0 on the assignment for all students involved, and potentially a grade of F in the course.

Note: You may use scikit-learn for the model implementations in this homework unless otherwise specified.

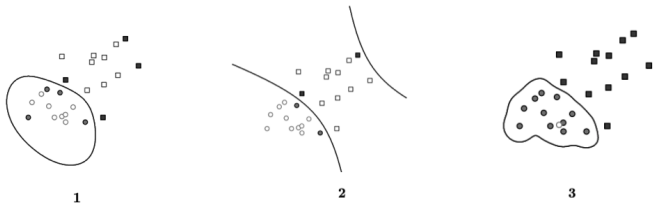
1 SVM [50 points]

1. [10 points] Figure 2 shows different SVMs with different decision boundaries. The training data is labeled as $y_i \in \{-1, 1\}$, represented as the shape of circles and squares respectively. Support vectors are drawn in solid. Match the scenarios described below to one of the 3 plots, and for each scenario explain in less than two sentences why it is the case.

   (a) A hard-margin kernel SVM with $k(u, v) = u \cdot v + (u \cdot v)^2$

   (b) A hard-margin kernel SVM with $k(u, v) = \exp(-5\|u - v\|^2)$

   (c) A hard-margin kernel SVM with $k(u, v) = \exp(-\frac{1}{5}\|u - v\|^2)$



2. [10 points] Show that an SVM using the polynomial kernel of degree two, $K(u, v) = (1 + u \cdot v)^2$, is equivalent to a linear SVM in the feature space $(1, x_1, x_2, x_1^2, x_2^2, x_1x_2)$ and hence SVMs with this kernel can separate any elliptic region from the rest of the plane.

   Note: the equation of an ellipse in the two-dimensional plane is $c(x_1 - a)^2 + d(x_2 - b)^2 = 1$.

3. [10 points] For the Breast Cancer Wisconsin (Diagnostic) Dataset, train a SVM classifier for 10,000 iterations using the template provided (**Code/Template_SVM.py**) with hyper-parameter setting $C = 1$, $lr = 0.002$(C is the regularization parameter, lr is the learning rate), and use the sub-gradient with a learning rate scheduler $lr_t = lr_0/\sqrt{iter + 1}$ (lr_0 is the base learning rate, $iter$ is the current iteration which starts at 0). Report the F1 score, precision and recall of the model on the training set and the testing set (in the table 1).
   Hint: objective function: $\arg\min_{\mathbf{w}} C \sum_{i=1}^{n} max(0, 1 - y_i(\mathbf{w}^T\mathbf{x} + b)) + ||\mathbf{w}||_2^2$; you should be getting a training f1 score at around **0.85**, and a testing f1 score at around **0.70** for the parameter setting.

Note: sklearn.svm is not allowed for this question.

	F1 score	Precision	Recall
training set			
testing set			

Table 1: Results for Question 1.4

4. [10 points] For the Breast Cancer Wisconsin (Diagnostic) Dataset, train a SVM classifier with linear kernel, polynomial kernel, RBF kernel, report the F1 score, Precision and Recall of the model on the training set and testing set in the table 2.
Note: You should use sklearn.svm.SVC for this question.

	F1 score	Precision	Recall
linear train			
linear test			
poly train			
poly test			
RBF train			
RBF test			

Table 2: Results for Question 1.5

5. [10 points] Use the model you built from the last question, retrain the models on the first two dimensions of the Breast Cancer Wisconsin (Diagnostic) Dataset, and plot the decision boundary of the three trained model on the first two dimensions as your report for this question (Three models: linear kernel, polynomial kernel, RBF kernel).
Note: The plotting function is provided in the template.

2 Ensemble Methods [40 points]

1. [4 points] If one of the features is a strong predictor of the class label, will **ALL** the trees in a random forest have that feature as root node? Explain your answer.

2. [3 points] Can the different learners in **bagging** based ensembles be trained in parallel. Explain your answer.

3. [3 points] Can the different learners in **boosting** based ensembles be trained in parallel. Explain your answer.

4. [10 points] Train a random forest classifier on the provided gene expression data using $sklearn.ensemble.RandomForestClassifier$. Vary the number of trees from 1 to 150, using parameter *n_estimators*. Provide one combined plot containing three graphs showing test classification error (1-accuracy) vs number of trees for three different values of *max_features* [p/3, p, $\sqrt{(p)}$], where p is the total number of features in gene expression data. *Note* We have provided *gene_test_x.csv* and *gene_test_y.csv* files to calculate the test classification error.
Hint: for max_features = p and 150 estimators we got an error of 0.15

5. [5 points] In the previous question, explain how different values of *max_features* and *n_estimators* effect the test classification error.

6. [10 points] Now train an Adaboost classifier on the provided gene expression data using $sklearn.ensemble.AdaBoostClassifier$ module. Vary the number of trees from 1 to 150, using parameter *n_estimators*. Provide one combined plot containing three graphs showing test classification error vs number of trees for three different values of *max_depth* of {1, 3, 5} of decision tree *base_estimator*. Keep learning rate fixed at 0.1 for all the models.
Hint: for max_depth = 3 and 150 estimators we got an error of 0.1

7. [5 points] In previous question, explain how different *max_depth* values of the base estimator effect the test classification error.

3 Stack different models [10 points]

1. [10 points] Now, for the Breast Cancer Wisconsin (Diagnostic) Dataset, try stacking multiple models using $sklearn.ensemble.StackingClassifier$ module in two levels (you may have to upgrade your scikit-learn package as this is a relatively new functionality and introduced in version 0.22). Specifically, you'll need to train some models and then use their predictions as input to another model which then learns to combine them together and predicts the final class (StackingClassifier does this automatically for you for more information please read the documentation of StackingClassifier: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.StackingClassifier.html) together. Try different combinations of models and report the best validation f1-score (weighted average) you are getting and explain the model you used.
Hint: try to get a weighted f1-score of greater than or equal to 0.95