CSci 5521 Homework 4
Due: Monday 9 December 2024 at 11:59 PM CST

- You may use anything in `numpy`, `sklearn` and `matplotlib`. If you have doubts about using a specific function or method, ask Professor Boley or a TA.
- You are encouraged and expected to do this assignment in groups. Only **one** group member should upload solutions to Canvas. However, **all** group member names, student IDs, and emails must be on the first page of the solution. When submitting, please combine all the Python files and any other files you use into one `.zip` file.

In this assignment, you will work on the Fashion MNIST dataset and another Kaggle dataset of your choice (see the end of the assignment for recommendation). You must use a dataset with two classes. In case you have a multi class dataset, make sure you select a subset of two existing classes to create binary class labels in your data. A filtered copy of the Fashion MNIST data consisting of just classes 5 & 6 is included on the class Canvas site.

1. After choosing a binary classification dataset, report the following:
   (a) Data set name, data set source (report the link from where to find the dataset and report how to reconstruct) and data set description. Also mention the sample size and the test/train split.
   (b) Brief description about the features and the target variable.
   (c) Any data cleaning or feature engineering steps you performed to get the data model-ready. You may also select any regression problem and convert the continuous target variable into binary labels using the mean or median, if it has a physical meaning. Note, feature transformation and scaling can help improve task performance.
2. Implement a linear SVM in dual form using the linear SVM classifier `sklearn.svm.LinearSVC` (you can also use `sklearn.svm.SVC` with `kernel='linear'`. Note, this choice may or may not scale better with larger number of samples). If the dataset does not come with a training/test split, divide your data into 70 % training and 30 % testing sets. Use 5 fold cross validation on the training set to choose the value of `C`. Try `C` $= \{0.01, 0.1, 1.0, 10, 100\}$. Choose the `C` with lowest average cross validation test error and then apply the trained model with that `C` to the 30 % held out test set. Report your average training and test error rates from cross validation runs for each `C`, and the final error rate on the held out test set. Also, report the confusion matrix on the held out test set.
3. Depending on the the dataset, the best decision boundary for optimal binary classification may not be a linear hyper plane. We can use kernels to create nonlinear classifiers. `sklearn.svm.SVC` or `sklearn.svm.NuSVC` can be used to implement kernel SVM. Implement rbf kernel by setting the argument `kernel='rbf'`. Similar to the previous question, divide your data into 70 % training and 30 % testing sets.
   (a) Use 5 fold cross validation (using `sklearn.model_selection.GridSearchCV`) on the training set to choose the value of `C` (`sklearn.svm.SVC` implementation) or ν (`sklearn.svm.NuSVC`

implementation). Choose the `C` / ν with lowest average cross validation test error and then apply the trained model with that `C` / ν to the 30 % held out test set. Report your average training and test error rates from cross validation runs for each `C` / ν, and the final error rate on the held out test set. Also, report the confusion matrix on the held out test set.

(b) Apart from the hyper parameter `C` / ν, `'rbf' kernel` has another hyper parameter γ, dictating the value of kernel coefficient. How do your `C` / ν and γ values affect your model complexity and task performance?

4. Do the kernel SVMs have better model performance than linear SVM? If the feature set has more than two features, perform PCA on your feature set to reduce the dimensions to two features. Divide the data set into 70 % training and 30 % testing data. Implement the best performing model and report the training and test set accuracy values. Using the test data set, plot the data points with the two features/ principal components on the axes. Distinguish the labels by coloring the points differently for the two labels, and plot the decision boundary for each SVM model. **HINT:** For visualizing the decision boundary, you can create a mesh grid covering the possible ranges of x and y axis (*i.e.*, using `numpy.meshgrid`), and then apply `plt.contourf` to different points within the grid together with their predicted labels.

5. (Extract credit question) Implement a Multi-layer Perceptron classifier using `sklearn.neural_network.MLPClassifier`. Following the model selection process in the previous question (*i.e.*, 5-fold cross validation), and experiment with different settings for the size of the hidden layer (*e.g.,* $\{50, 100, 150, 200, 250\}$). Report your average training and test error rates from cross validation runs, and the final error rate on the held out test set. Also, report the confusion matrix on the held out test set.

Instructions

All code must be written in python.

- **Things to submit**

  1. `hw4_sol.pdf`: A document which contains the solutions. The front page of the PDF file should have names and UMN email addresses of the students submitting the document. Also include the summary of results.
  2. `zip` file containing all python files and any other files you used in this assignment.

A Sample List of Kaggle Datasets

- Normal and Pneumonia chest x-rays
- Sign language digits dataset
- Sign language alphabet dataset
- Fruit images classification
- Land use classification

- Cloud vs non cloud aerial images
- Rice leaf diseases classification

Here are the URLs in case you have trouble with the embedded links.

scikit links

sklearn.svm.LinearSVC
https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html#sklearn.svm.LinearSVC

sklearn.svm.SVC
https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

sklearn.svm.SVC
https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

sklearn.svm.NuSVC
https://scikit-learn.org/stable/modules/generated/sklearn.svm.NuSVC.html

sklearn.neural_network.MLPClassifier
https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

kaggle links

Kaggle datasets
https://www.kaggle.com/datasets

Fashion MNIST
https://www.kaggle.com/zalando-research/fashionmnist

Normal and Pneumonia chest x-rays
https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

Sign language digits dataset
https://www.kaggle.com/ardamavi/sign-language-digits-dataset

Sign language alphabet dataset
https://www.kaggle.com/ash2703/handsignimages

Fruit images classification
https://www.kaggle.com/souro12/ccxzvv

Land use classification
https://www.kaggle.com/apollo2506/landuse-scene-classification

Cloud vs non cloud aerial images
https://www.kaggle.com/ashoksrinivas/cloud-anomaly-detection-images

Rice leaf diseases classification
https://www.kaggle.com/vbookshelf/rice-leaf-diseases