DATA 602 @ UMBC

Homework Assignment 3

Note: Please work out the following problems in one single jupyter notebook in a coherent manner. For illustrations and questions which do not involve coding, use markdown cells.

Note: For deep learning calculations, you have the freedom to work with either of the TensorFlow or PyTorch libraries.

Note: From problems 1, 2, and 3, choose two problems to solve. If you solve all three, you'll receive extra credit!

1. **Constructing a Fully-Connected Model:** For this problem, you will be using the breast cancer dataset of scikit-learn. Use the following piece of code

   ```
   from sklearn.datasets import load_breast_cancer
   bc = load_breast_cancer()
   ```

   to download the data. Use the description provided by scikit-learn and familiarize yourself with the dataset (Use print(bc.DESCR) to view the description).

   (a) How many features does the breast cancer dataset possess? Specify the nature of the classification problem (*i.e.* binary vs. multinomial, and balanced vs. unbalanced).

   (b) Implement a random forest classifier on the whole dataset and identify the 10 most relevant features. For the rest of the problem, focus on these 10 features.

   (c) Split the data into train and test with the test size being 0.25 of the size of the whole dataset.

   (d) Construct a fully-connected neural network for this classification problem. In your training, calculate the accuracy score and the mean $F1$-score for the train and test data at each epoch. Plot your accuracy per epoch for both train and test data.

   Note: Once the training is complete, the accuracy score and the mean $F1$-score of your fully-connected model should reach to at least $\sim 90\%$ for both train and test data. Moreover, your model should not suffer from a high level of overfitting.

   (e) Employ the logistic regression for the same classification problem. Carefully assess the performance of logistic regression on this dataset.

   (f) How many weights (*i.e.* parameters) does your fully-connected model in part (d) possess? How many parameters does the logistic regression model in part (e) possess? Carefully compare and contrast the two performances and determine which model should be employed as the more successful model for this classification task.

2. **Constructing a CNN Classifier:** Define $\mathcal{D}$ to be a dataset consisting $5-10\%$ of random images of the MNIST. Redefine the target variable for $\mathcal{D}$ as follows:

   - **Positive Class:** This class is formed by all images of $\mathcal{D}$ which are 2 or 7.
   - **Negative Class:** This class is formed by all images of $\mathcal{D}$ which are different from 2 and 7 (*i.e.* images of $0, 1, 3, 4, 5, 6, 8, 9$ digits).

   (a) How many samples does $\mathcal{D}$ possess? What is the size of the positive class? What is the size of the negative class? Determine the nature of this classification problem (balanced vs. unbalanced).

   (b) Construct a convolutional neural network for this binary classification problem on $\mathcal{D}$. What metrics would you calculate through the training process?

   (c) What is the accuracy score of your CNN classifier? What is the accuracy score of a sharp classifier for this problem? Carefully assess the performance of the CNN classifier constructed in part (b).

3. **Length of Time Sequence in RNN Models:** Consider the example of Tesla's stock value we studied in lecture 11. For this example, we structured the input of the model in the form of a tensor of shape $(B, T = 50, d = 4)$. In this problem, we would like to study the effect of the length of the input sequence on the accuracy of the model.

   (a) Take the same simple RNN model we constructed in class and run this model for $T = \{5, 10, 20, 30, 40, 50, 60\}$. Collect the R^2-scores for the train and test data for different values of T in a dataframe. Plot the R^2-score for the train and test data as a function of T.

   Note: Try to automate this problem as much as you can. Do <u>not</u> execute the model 7 times by hand! Take the advantage of appropriate loops and/or functions.

   (b) Based on your result from part (a), assess the role of T in the accuracy of the model.

4. **Dimensionality Reduction and Clustering of Food Items:** For this problem, use the csv file food-nutrients.csv. This csv file which has been published by the U.S. Department of Agriculture (USDA) consists of 8618 food items for which different food nutrients have been measured. The goal in this exercise is twofold. Since the dataset includes too many features, you will first use the principal component analysis to reduce the dimensionality of the dataset. In the next step, you will apply clustering algorithms to partition the food items into a number of clusters. In order to solve this exercise, proceed as follows:

   (a) Read the csv file into a dataframe. How many columns does the dataset have? Three columns of the dataset (CommonName, MfgName, and ScientificName) have a large number of missing values. Drop these 3 columns. The dataframe should now have 42 columns.

(b) The dataframe possesses 15 columns that include USRDA in their column titles. USRDA stands for "U.S. Recommended Daily Allowance." The USRDA measure was developed by the Food and Drug Administration (FDA) since 1972, and this measure is specifically developed for the use in nutritional labeling. In our analysis, we are not interested in USRDA's, and we solely focus on the nutrients of food items. Drop all 15 columns that have USRDA in their title. Try to do this in an automatic manner (Search for the term USRDA in the titles of columns, and if the term USRDA is found, drop the corresponding column). After dropping USRDA columns, your dataframe should possess 27 columns four of which are categorical (ID, FoodGroup, ShortDescrip, and Descrip). The remaining 23 columns are continuous features that record the food nutrients, and you will focus on these columns for the rest of the exercise.

(c) Standardize the data. After standardization, all the 23 continuous features will be dimensionless.

(d) Apply principal component analysis (PCA) to the standardized data. Note that after applying PCA, you will find 23 new continuous features which are linear combinations of the old 23 features.

(e) Plot the explained variance ratio by each PCA component. You can use a bar chart for this purpose. Check that the sum of all 23 ratios is indeed 1.

(f) Plot the cumulative explained variance ratio by PCA components.

(g) How many PCA components should you choose in order to explain 80% of the whole variance? Let us call this number d_{PCA}. Commit to this many (*i.e.* d_{PCA}) PCA components for the rest of the exercise (This means that you have reduced the number of dimensions from 23 to a smaller number which is nothing but the number of PCA components (d_{PCA}) needed to explain 80% of the whole variance for this dataset).

(h) Take the first new continuous feature (*i.e.* the first PCA feature). Note that this feature (and any other PCA feature) is a linear combination of the old 23 continuous features. Find out the exact linear combination. What old features have the largest coefficients in the linear combination? These features are the dominant old features for the first PCA dimension.

(i) Make a new dataframe, pca_df, of the food items with d_{PCA} continuous features.

(j) In this part, you would like to apply K-means clustering algorithm to pca_df. How many clusters would you choose to start the algorithm with? Perform a thorough analysis to find the optimal number of clusters.

(k) Perform K-means clustering with the number of clusters you determined in part (j).

(l) Add a new column to pca_df which indicates the cluster label. Find the size of each cluster you constructed in part (k).

(m) Take two of the largest clusters and find the number of food items in each food category.

Hint: For this purpose, you can focus on the column FoodGroup and use the pandas command .value_counts() to count the number of food items for a given cluster.

(n) What are the dominant food groups in the two clusters in part (m)? Based on this result, what names would you give to these two clusters?