Lab 2

It is an individual programming assignment. This lab assignment is graded based on 100 points and is an individual effort (no teamwork is allowed).

Part 1 (20 pts) Recommender System

Download m1.zip file from the link (https://grouplens.org/datasets/movielens/1m/)

You will be using movies.dat and rating.dat for building your recommender.

1. Create m x u matrix with movies as row and users as column. Normalize the matrix.

2. Compute SVD to get U, S and V. Use np.linalg.svd()

3. From your V.T select 50 components.

4. Implement a function that take movieID as input and then implement cosine similarity along with sorting to recommend top 10 movies.

5. Repeat the same process except now instead of using SVD you will use PCA to get the eigenvectors.

6. You will require co-variance matrix as an input to your eig function.

Use np.cov() for getting co-variance matrix.
Use np.linalg.eig() for getting eigen vectors.

7. Use that same steps after that to get 50 components. Use cosine similarity to get the results.

8. Compare the results for SVD and PCA.

Submission Details: .ipynb file (Write your comments on result comparison as markdown)

Part 2 (30 pts) Use Neural Networks + Minimum 3 Hidden Layers

For the dataset, hcc-data-complete-balanced.csv, do the following:

1. Data clean up – feature engineering

2. Impute the missing values with mean, median and mode. You need to evaluate which method is better based on the F1 score.

3. Build a neural network with at least 3 hidden layers and 4 neurons each. Train and test.
4. Which activation functions are you using? Why?
5. Tune the hyperparameters using cross-validation and see what precision you can achieve
6. Is using Adam optimization and early stopping helpful in this problem? Why?
7. Now try adding Batch Normalization and compare the learning curves: is it converging faster than before? Does it produce a better model?
8. Is the model overfitting the training set? Try adding dropout to every layer and try again. Does it help?
9. What is the final model you've arrived? Draw the neural network to explain your solution.
10. Mention your F-1 score for each development in your model

Submission Details: .ipynb file (Write your comments as markdown)

Part 3 (20 pts) Use XGBoost

Using XGBoost, predict avocado price.

1. Load the dataset
2. Train and test the data
3. Perform feature engineering
4. What features are the most correlated?
5. Build a model
6. Use XGBoost
7. Fine tune the parameters – explain each and every step in detail. Which parameter? Why this value?
8. Evaluate the performance of your model
9. Explain in detail what is happening inside your model? How have you built this model?

Submission Details: .ipynb file (Write your comments as markdown)

Part 4 (30 pts) NLP

Dataset having 50K movie reviews for natural language processing or Text analytics for binary sentiment classification.

1. Download data from Lab 2 folder on Canvas.

2. Split the data into 80% training set and 20% test set.

3. Predict the number of positive and negative reviews using various algorithms to increase the performance of the prediction of the sentiment expressed in the review.

Note: Grading for this question will be done based on parameters such as research on the topic, different algorithms taken into consideration, a justification for the selected algorithm, implementation, and the accuracy achieved.

Submission Details: .ipynb file (Write your comments as markdown)

Please upload all the solution files in a zipped folder (FirstName_LastName_Lab2.zip) on the canvas before the deadline.