Project 5: Going further with Deep Learning

Group Members - Sruti Munukutla and Mahvash Maghrabi

Github - https://github.com/mahvashmaghrabi/CS6140_Project5

Working Pattern -
Task 1, 2 - Sruti Munukutla
Task 3, 4 - Mahvash Maghrabi

About the Dataset

The dataset given is a financial dataset and the aim of the project is Stock market prediction - to predict the future direction of movement of the 5 indexes : S&P 500, RUSSELL, NASDAQ, NYSE and DJI. The downloaded dataset has all the independent features for each market index in separate files. The data is a time-series data with each row representing the market status for each day.

Task 1

The goal of this task is to compile all the data across 5 csv files into a single csv file. Each row represents the data for all 5 indexes (or the files are joined by the date) for a particular day. To combine the data for every date, the csv files were read and converted into pandas dataframe. Then, an outer join was performed on the date. The missing data was filled with zeroes. Since the Close column for each index represents the closing market value for that day, the next day prediction is calculated as mentioned in the paper. If Close_N+1 - Close_N > 0, ‘Up’, else ‘Down’. ‘Up’ represents a positive/high market movement on the next day and ‘Down’ represents a negative/low market movement on the next day.

Task 2

The two ML models used for this task were KNN Classifier, Decision Tree Classifier.

KNN Classifier:

K-nearest neighbour classifier was used for nearest 3 neighbours. The x_train consists of all the independent variables except Date, and index strings such as ‘DJI’, ‘S&P’ since they do not reveal much information. The model was fit on x_train representing all the independent variables. y_train is all the 5 dependent variables. Hence a multi output array.
Accuracy of this model was not great, the results for each indexes are as follows:

[DJI, NASDAQ, NYSE, RUSSELL, SNP]
[54.15617128463476, 56.67506297229219, 51.13350125944584, 51.13350125944584, 51.88916876574308]

Decision Tree Classifier:

Another model used was Decision Tree. The output of this model was:

[DJI, NASDAQ, NYSE, RUSSELL, SNP]
[52.39294710327456, 51.88916876574308, 47.3551637279597, 48.86649874055416, 48.86649874055416]

We did not implement different models for each stock. Our baseline results for the above two models show lower accuracy.

Task 3 - Implement an ML method that uses more than one day to predict each market index

CSV File Creation

The first step towards working on this task was building a day2prediction.csv file. The new file was required to have two days worth of data in a single row. The approach towards building this file was to concatenate two rows and append the second row to the first row for every other row. The day1prediction.csv file has 421 columns and 1984 rows. The day2 prediction.csv file has double the number of rows that is 842 and half the number of columns that is 992. The file that has the concatenated rows also has a suffix with all of their column names '_day2' so that the machine learning model considers it as a new attribute.

Label Encoding

It converts the labels into a numeric form so that it is in the machine-readable form. The categorical values are converted to numerical values when we apply label encoding. Label encoding was performed on the day2prediction file so that all the categorical values are converted to numerical values which will make it feasible for the ML model to train on the dataset. The attributes that had categorical values were as follows : 'Name_DJI', 'Name_NASDAQ', 'Name_NYSE', 'Name_RUSSELL','Name_SNP', 'DJI_next_day', 'NASDAQ_next_day', 'NYSE_next_day', 'RUSSELL_next_day', 'SNP_next_day', 'Name_DJI_day2', 'Name_NASDAQ_day2', 'Name_NYSE_day2', 'Name_NYSE_day2', 'Name_RUSSELL_day2', 'Name_SNP_day2', 'DJI_next_day_day2', 'NASDAQ_next_day_day2', 'NYSE_next_day_day2', 'RUSSELL_next_day_day2', 'SNP_next_day_day2'

Input and Output Variables for the Model Training

The predicted values of the dependent variables ie, direction of each index has been determined by previous day closing index value and current day closing index value, if it is reduced then market direction is labelled as down else the market direction is labelled as up. So, each market index ending with the suffix '_day2', will be considered as the output variables of the model and the market index column without the suffix '_day2' will be

dropped as they are no longer required to train the ML model. The Date column as well as the Date_day 2 column will be dropped as the dates are not required while training the model and only required for concatenating two rows which is two days worth of data.

Implementing ML method

Two machine learning models were performed on the day2prediction dataset. The first model applied was Support Vector Machine and the second model was Random Forest Classifier.

Support Vector Machine(SVM)

It is a supervised learning machine learning algorithm that can be used for both classification or regression problems. In this algorithm we plot each data item as a point in n-dimensional space where n is the number of features you have where the value of each feature is the value of a particular coordinate. Then, we perform classification by finding the optimal hyper-plane that differentiates the two classes very well.
The SVM model is built taking into account the input and output variables. The dataset is divided into train and test set using the train_test_split function. Also the code drops columns 'Date', 'Date_day2', 'DJI_next_day', 'NASDAQ_next_day', 'NYSE_next_day', 'RUSSELL_next_day', 'SNP_next_day' as these are not required for training the SVM model. The code then iterates over a list of output variables which is basically the different market indexes which are 'DJI_next_day_day2', 'NASDAQ_next_day_day2', 'NYSE_next_day_day2', 'RUSSELL_next_day_day2', 'SNP_next_day_day2' and performs the model training for each output variable/market index and calculates the accuracy for all.

The results for accuracy for SVM obtained are as follows :
Accuracy of SVM model for DJI_next_day_day2: 63.82%
Accuracy of SVM model for NASDAQ_next_day_day2: 64.82%
Accuracy of SVM model for NYSE_next_day_day2: 64.32%
Accuracy of SVM model for RUSSELL_next_day_day2: 61.81%
Accuracy of SVM model for SNP_next_day_day2: 62.31%

Random Forest Classifier

The Random forest is a supervised Machine learning algorithm used for classification or regression using decision trees. Random forest consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. It creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees from a randomly selected subset of the training set and then it collects the votes from different decision trees so as to decide the final prediction.
The dataset is trained using the Random Forest Classifier model. It divides the data set in training and testing sets using the trin_test_split function. The code drops columns 'Date', 'Date_day2', 'DJI_next_day', 'NASDAQ_next_day', 'NYSE_next_day', 'RUSSELL_next_day', and 'SNP_next_day' as these are not required for training the ML model. The code has list of target variables named targets which includes 'DJI_next_day_day2', 'NASDAQ_next_day_day2', 'NYSE_next_day_day2', 'RUSSELL_next_day_day2', and 'SNP_next_day_day2'. These are the market indexes which shows the direction of each

individual market as the model is applied on all of the them. The results obtained by Random Forest Classifier were very good compared to other models.

The results of accuracy for Random Forest are as follows :
Accuracy of Random Forest model for DJI_next_day_day2: 98.99%
Accuracy of Random Forest model for NASDAQ_next_day_day2: 98.49%
Accuracy of Random Forest model for NYSE_next_day_day2: 95.98%
Accuracy of Random Forest model for RUSSELL_next_day_day2: 93.47%
Accuracy of Random Forest model for SNP_next_day_day2: 94.47%

Task 4 - Improve your model

This task was all about improving the models created in the previous step. The two models created were SVM and RF. The accuracy of these models were improved in this task. Two methods were followed to improve the accuracy of the models :

1) Hyperparameter Tuning using GridSearchCV
2) Taking four days worth of data in a single row

1) The hyperparameter tuning using GridSearchCV was performed on the SVM model's each market index. The best hyperparameters were obtained and then the model was trained using the best hyperparameters. The output obtained showed the best hyperparameters along with the model accuracy for each index.

The results along with best hyperparameters obtained are as follows :

Accuracy of SVM model for DJI_next_day_day2: 63.82%
Best parameters for DJI_next_day_day2: {'C': 1, 'kernel': 'linear'}
Accuracy of SVM model for NASDAQ_next_day_day2: 64.82%
Best parameters for NASDAQ_next_day_day2: {'C': 100, 'kernel': 'linear'}
Accuracy of SVM model for NYSE_next_day_day2: 65.83%
Best parameters for NYSE_next_day_day2: {'C': 100, 'kernel': 'linear'}
Accuracy of SVM model for RUSSELL_next_day_day2: 64.82%
Best parameters for RUSSELL_next_day_day2: {'C': 10, 'kernel': 'linear'}
Accuracy of SVM model for SNP_next_day_day2: 60.30%
Best parameters for SNP_next_day_day2: {'C': 10, 'kernel': 'linear'}

It showed that the model accuracy for DJI market and the NASDAQ market remained as it was. The model accuracy for NYSE increased from 64.32% to 65.83%. Also the accuracy for RUSSELL market increased from 61.81% to 64.82%. The model accuracy decreased for the SNP market from 62.31% to 60.30%.
Then hyperparameter tuning using GridSearchCV was also performed on the Random Forest Classifier model's every market index. The model was then trained on the best hyperparameters for each market index.

The results along with best hyperparameters obtained are as follows :

Accuracy of Random Forest model for DJI_next_day_day2: 96.98%

Best parameters for DJI_next_day_day2: {'max_depth': 15, 'n_estimators': 100}
Accuracy of Random Forest model for NASDAQ_next_day_day2: 97.49%
Best parameters for NASDAQ_next_day_day2: {'max_depth': 15, 'n_estimators': 50}
Accuracy of Random Forest model for NYSE_next_day_day2: 95.98%
Best parameters for NYSE_next_day_day2: {'max_depth': 10, 'n_estimators': 100}
Accuracy of Random Forest model for RUSSELL_next_day_day2: 95.48%
Best parameters for RUSSELL_next_day_day2: {'max_depth': 15, 'n_estimators': 150}
Accuracy of Random Forest model for SNP_next_day_day2: 97.99%
Best parameters for SNP_next_day_day2: {'max_depth': 15, 'n_estimators': 150}

The model accuracy of the DJI market decreased from 98.99% to 96.98%. For the NASDAQ market the model accuracy decreased from 98.49% to 97.49%. For NYSE the model accuracy remains the same. The model accuracy increased for the remaining two markets, for RUSSELL it increased from 93.47% to 95.48% and for SNP it increased from 94.47% to 97.99%.

2) The next task to improve both the models was to take into consideration four days of data to train the machine learning models. A csv file was created called day4prediction.csv using the same method as the day2prediction.csv file that was created for Task 3. Four days of data was appended to one row. Label encoding was performed on categorical values. Now the Random Forest and SVM models were applied on the day4prediction.csv file.

The results obtained for SVM are as follows :
Accuracy of SVM model for DJI_next_day_day4: 59.00%
Accuracy of SVM model for NASDAQ_next_day_day4: 51.00%
Accuracy of SVM model for NYSE_next_day_day4: 67.00%
Accuracy of SVM model for RUSSELL_next_day_day4: 63.00%
Accuracy of SVM model for SNP_next_day_day4: 60.00%

The model accuracy increased for the NYSE market from 64.32% to 67.00% and for RUSSELL market it increased from 61.81% to 63.00%. For the remaining three markets the model accuracy decreased for DJI it decreased from 63.82% to 59.00%, for NASDAQ it decreased from 64.82% to 51.00% and for SNP it decreased from 62.31% to 60.00%

The results obtained for Random Forest are as follows :
Accuracy of Random Forest model for DJI_next_day_day4: 95.00%
Accuracy of Random Forest model for NASDAQ_next_day_day4: 96.00%
Accuracy of Random Forest model for NYSE_next_day_day4: 95.00%
Accuracy of Random Forest model for RUSSELL_next_day_day4: 83.00%
Accuracy of Random Forest model for SNP_next_day_day4: 98.00%

The model accuracy when using four days worth of data increased only for the SNP market from 94.47% to 98.00% while for the other market the model accuracy decreased.

EXTENSIONS

For Task 3 instead of using one ML model two ML models were used. The first ML model used was Support Vector Machine and then Random Forest Classifier was used. The results obtained by RF model were quite good as compared to the results obtained by SVM. Hence Random Forest Classifier worked best out of all the models used for training.

SUMMARY

Task 1 prepared the data to combine all the csv files and join by the date into a single file day1prediction.csv. Task 2 trained the ML model to predict stock market movement for the test data. Two models were used: Decision Tree and KNN Classifier. It was observed that both the models gave relatively low accuracy.Task 3 gave insights about how two days worth of data can be used to train the ML model instead of using a single day data. The results obtained by Random Forest Classifier were quite good as compared to SVM. Task 4 was about improving the model accuracy. While working on improving model accuracy one thing that was noticed was that the model accuracy for different markets gave different results. For some the model accuracy increased , for some it decreased whereas for others it remained as it is. Overall the project was a great learning experience as it also taught a lot about different stock markets as well as the use of Machine Learning to predict the direction of the stock market.

REFERENCES

1. https://arxiv.org/pdf/1810.08923.pdf
2. https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.merge.html
3. https://www.geeksforgeeks.org/how-to-concatenate-two-or-more-pandas-dataframes/
4. https://www.geeksforgeeks.org/random-forest-classifier-using-scikit-learn/
5. https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47
6. https://www.geeksforgeeks.org/ml-label-encoding-of-datasets-in-python/

ACKNOWLEDGEMENT

I, Mahvash Maghrabi, and my group partner, Sruti Munukutla, would like to express our gratitude and appreciation to all those who gave us the opportunity to complete this project and the report. We extend our special thanks to Professor Bruce Maxwell who provided us the opportunity to work on this project and always gave us stimulating suggestions and encouragement during the lectures. We would also like to express our gratitude to our Teaching Assistants, Srishti Hedge and Yiming Ge, who were always available to resolve our doubts and queries. Lastly, we would like to thank each other as we both worked together and supported each other throughout the project.