Technical Report -Trading Strategy

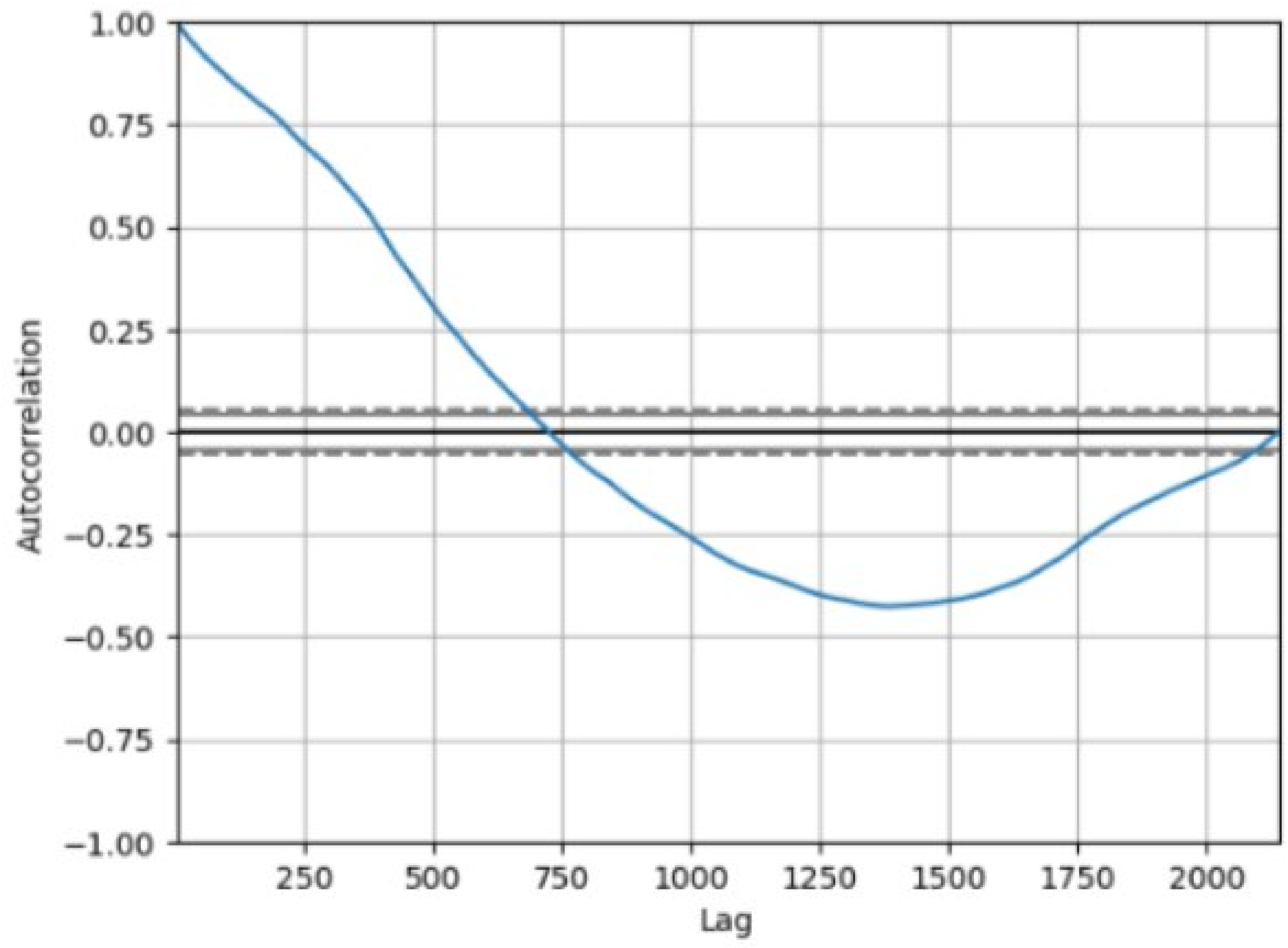
Date:06/07/23

Step 1

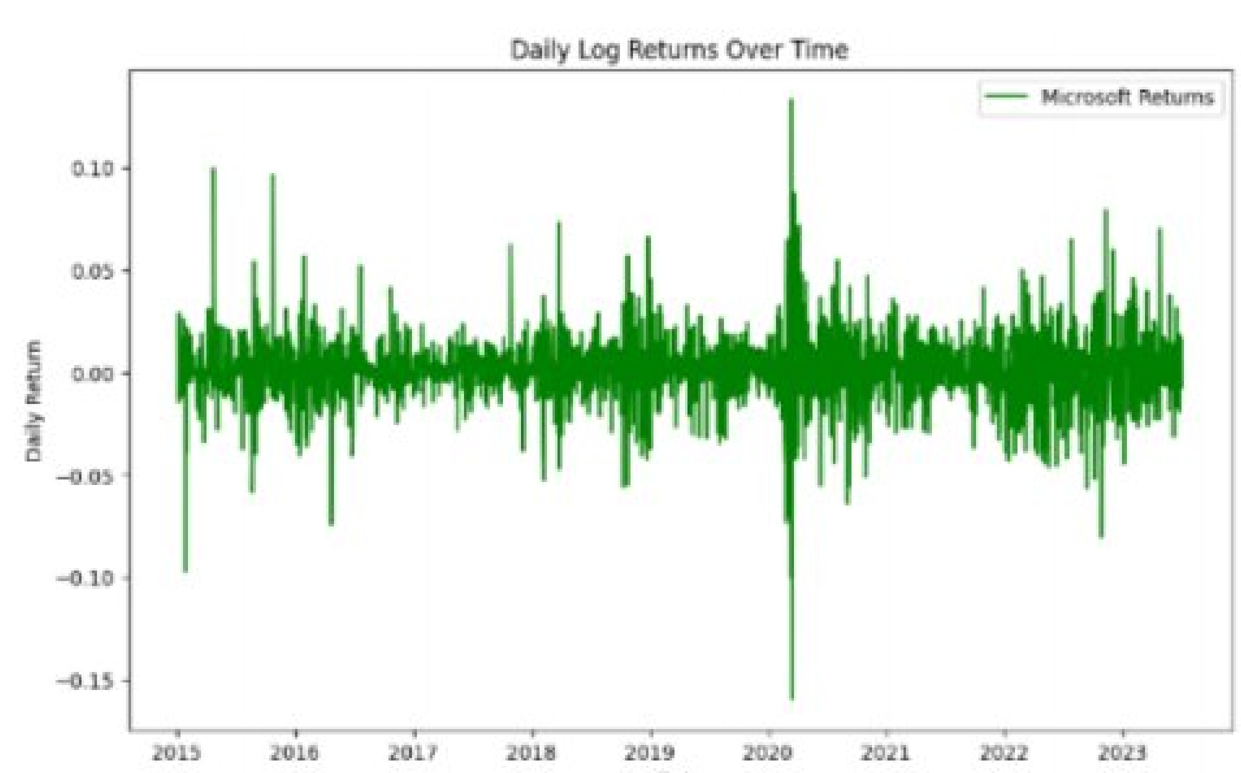
a) We have chosen Microsoft stock (MSFT) from 1st January 2015 onwards to 5th July 2023, a total of 2139 observations are there. We present below the summary statistics and autocorrelation.

```
Summary Statistics for Prices:
count    2139.000000
mean      146.572884
std        93.483414
min        35.095703
25%        59.278202
50%       114.300484
75%       237.423058
max       348.100006
Name: Adj Close, dtype: float64

Summary Statistics for Daily Returns:
count    2138.000000
mean        0.000991
std         0.017731
min        -0.159454
25%        -0.006797
50%         0.000843
75%         0.009915
max         0.132929
Name: Adj Close, dtype: float64
```



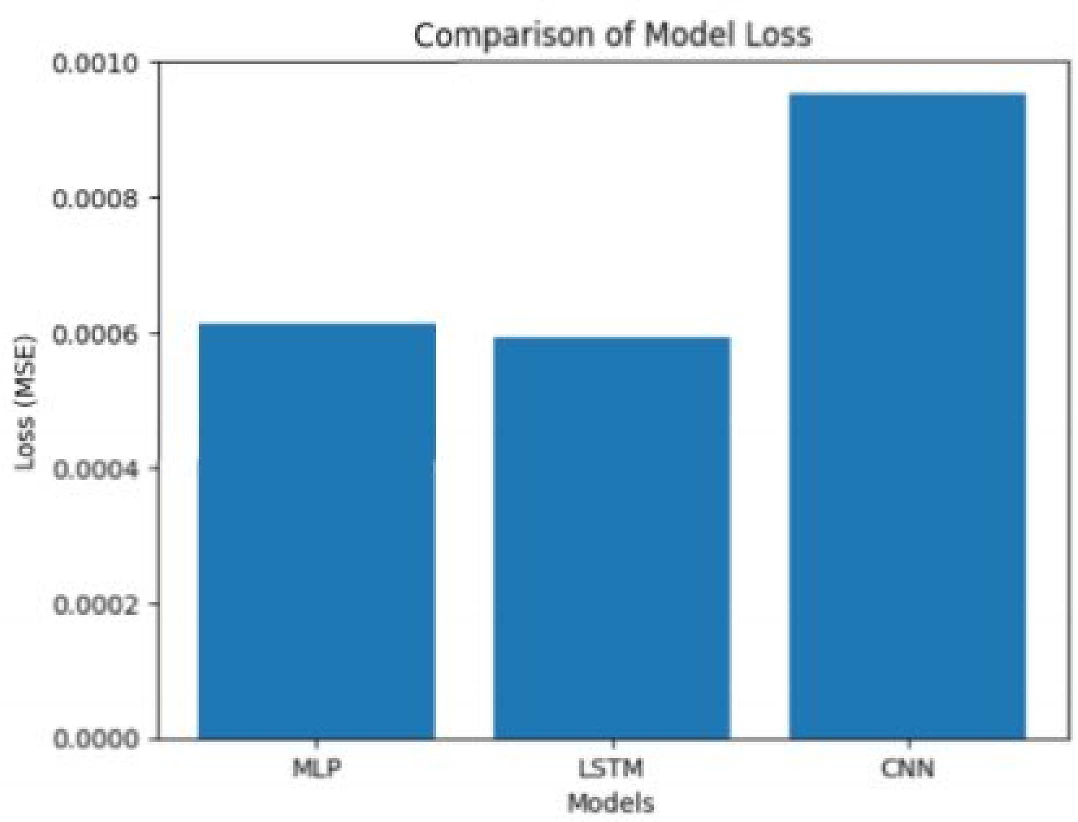
We plot the prices and log returns of the time series



We use the ADF test to conclude that the time series is not stationary but the returns are stationary.

b) We purposely introduce leakage by choosing the first 80% as our training data and remaining 20% as our testing data . Without any gap in between these two data , we are predicting the returns of the stock on the 30th Day. We use GramianAngularField to represent the data in image format for the CNN. We make the labels as 1 when stock gives a +ve return and 0 when -ve return . We verify the sizes of the various datasets which are as expected.

c) We create 3 model classes which will be conveniently used in steps 2 and 3 as well where we have described the architecture of a MLP , LSTM and CNN. Next we have created evaluate function where we get the minimum MSE loss of the models. The MSE turns out to be 0.259 for MLP , 0.250 for LSTM and 0.396 for CNN. We tried multiple architectures for CNN and report the one with the minimum MSE. Since for CNN the test samples were more we normalise the result and present the MSE in a bar graph format here:-



d) We backtest our models for 3 strategies , namely buy and hold, long only , and both long and short and provide the graphs here.

```
For  1b vanilla MLP with leakage in training  Strat Return Long Only = 31.918328314692126 %
For  1b vanilla MLP with leakage in training  Strat Return = 52.959497163819094 %
For  1b vanilla MLP with leakage in training  Buy and Hold Return = 10.73045820054308 %
```

```
For  1b vanilla LSTM with leakage in training  Strat Return Long Only = 44.95959774988718 %
For  1b vanilla LSTM with leakage in training  Strat Return = 84.87150834782804 %
For  1b vanilla LSTM with leakage in training  Buy and Hold Return = 10.73045820054308 %
```


```
For  1b vanilla CNN with leakage in training  Strat Return Long Only = -1.6770451294114919 %
For  1b vanilla CNN with leakage in training  Strat Return = -19.849488995919263 %
For  1b vanilla CNN with leakage in training  Buy and Hold Return = 2.8102425782930984 %
```

Here we observe that we both MLP and LSTM perform well on the dataset but CNN turns out to be the looser amongst the three. This is expected because it induces the most amount of information leakage . Also Our training loss was the highest in this.

Step 2

a) We use the training size as 500 and testing set size as 500 and perform non anchored forward walk method and report our backtesting. The results are as follows.

```
For  2 MLP with leakage in training  Strat Return Long Only = 225.15706064875712 %
For  2 MLP with leakage in training  Strat Return = 130.33908247409855 %
For  2 MLP with leakage in training  Buy and Hold Return = 310.9624674366798 %
```


```
For  2 LSTM with leakage in training  Strat Return Long Only = 149.73206603028353 %
For  2 LSTM with leakage in training  Strat Return = 39.014445858551674 %
For  2 LSTM with leakage in training  Buy and Hold Return = 310.9624674366798 %
```


```
For  2 CNN with leakage in training  Strat Return Long Only = 30.556266632520934 %
For  2 CNN with leakage in training  Strat Return = -69.67638822397717 %
For  2 CNN with leakage in training  Buy and Hold Return = 262.74357592082396 %
```

b) We use the training size as 500 and testing set size as 500 and perform non anchored forward walk method and report our backtesting. The results are as follows.

```
For  2 MLP with leakage in training  Strat Return Long Only = 44.782060098931 %
For  2 MLP with leakage in training  Strat Return = 2.2698081145101634 %
For  2 MLP with leakage in training  Buy and Hold Return = 97.07127611533213 %
```


```
For  2 LSTM with leakage in training  Strat Return Long Only = 54.0021973265568 %
For  2 LSTM with leakage in training  Strat Return = 16.279146725995155 %
For  2 LSTM with leakage in training  Buy and Hold Return = 97.07127611533213 %
```


```
For  2 CNN with leakage in training  Strat Return Long Only = 10.453706196933643 %
For  2 CNN with leakage in training  Strat Return = -41.24376733620069 %
For  2 CNN with leakage in training  Buy and Hold Return = 93.57213495354453 %
```

c) We see that our strategies are able to make good profit in step 2 a and b as compared to 1 . the Reason is that we have removed the bias to learn the information about just the last part of the time series. Specially in the case of MLP we can see a clear spike in performance here.

d) There is a lot of difference between parts a and b , we can clearly see that for some intervals our model is performing well , but for some intervals we were not able to learn well enough to predict good values hence making a loss making strategy in those intervals. Our LSTM graph looks quite off here, the reason might be that we were not able to train our lstm well enough with the information provided. We see a decrease in performance in Step 2 b as compared to step a . The issue might be that since test set is small , even a small amount of information leakage amounts to a huge percentage of the information leakage , leading to not too promising results.

Step 3

a) We have purged 30 instances in the training set which were overlapping with the testing set in each of our walk forward iterations in order to create a gap of time between the ending of training set and starting of testing set. This has allowed us to reduce the leakage of information from testing to the training set.

b) We use the training size as 500 and testing set size as 500 and perform non anchored forward walk method and report our backtesting. WE use purging of 30 instances in our training data to prevent leakage. The results are as follows: -

```
For  3 MLP without leakage in training  Strat Return Long Only = 288.7858165457566 %
For  3 MLP without leakage in training  Strat Return = 243.35016511707641 %
For  3 MLP without leakage in training  Buy and Hold Return = 310.9624674366798 %
```

```
For  3 LSTM without leakage in training  Strat Return Long Only = 308.0804550146416 %
For  3 LSTM without leakage in training  Strat Return = 308.0804550146416 %
For  3 LSTM without leakage in training  Buy and Hold Return = 310.9624674366798 %
```


```
For  3 CNN without leakage in training  Strat Return Long Only = 11.67735695745915 %
For  3 CNN without leakage in training  Strat Return = -77.9826668842922 %
For  3 CNN without leakage in training  Buy and Hold Return = 262.74357592082396 %
```

c) We use the training size as 500 and testing set size as 100 and perform a non anchored forward walk method and report our backtesting. WE use purging of 30 instances in our training data to prevent leakage. The results are as follows: -

```
For  3 MLP without leakage in training  Strat Return Long Only = 63.48442913719234 %
For  3 MLP without leakage in training  Strat Return = 30.74076533088801 %
For  3 MLP without leakage in training  Buy and Hold Return = 97.07127611533213 %
```


```
For  3 LSTM without leakage in training  Strat Return Long Only = 54.0021973265568 %
For  3 LSTM without leakage in training  Strat Return = 16.279146725995155 %
For  3 LSTM without leakage in training  Buy and Hold Return = 97.07127611533213 %
```

```
For  3 CNN without leakage in training  Strat Return Long Only = 18.889082084940757 %
For  3 CNN without leakage in training  Strat Return = -31.743677740647925 %
For  3 CNN without leakage in training  Buy and Hold Return = 93.57213495354453 %
```

d) Here we see even worse results than from Step 2 because we have substantially reduced overfitting and it has given us a close to real world prediction of our strategies. We see the same difference between 3c and 3b as we have seen in 2a and 2b. Our best explanation is that we are not able to train our model properly for some of the ranges.

References

- Gray, Chad. "Stock Prediction with ML: Walk-forward Modeling." The Alpha Scientist, 18 July 2018, https://alphascientist.com/walk_forward_model_building.html
- Audhiaprilliant. "Walk-Forward Optimization: Cross-Validation Technique for Time-Series Data." Medium, 19 Feb. 2021, https://audhiaprilliant.medium.com/walk-forward-optimization-cross-validation-technique-for-time-series-data-61739f58f2c0
- Unger Academy. "How to Use Walk Forward Analysis: You May Be Doing It Wrong!" Unger Academy, 16 Jan. 2022, https://ungeracademy.com/posts/how-to-use-walk-forward-analysis-you-may-be-doing-it-wrong
- Scikit-Learn. "Cross-Validation." Scikit-Learn, 2022, https://scikit-learn.org/stable/modules/cross_validation.html.