Multivariate Forecasting with Deep Learning Model

Serkan Comu, Chris Lee, Jiahui Li

1. Problem Statement

The goal of this homework is to forecast stock return using firm's financial and operating data. Given the quarter data of Fortune 500 companies, we tested our deep learning models and made predictions among 25 of them.

2. Methodology

The raw data consists of 14 variables: i) Company Name, ii) Ticker Symbol, iii) Asset Total, iv) Cash, iv) Cost of Goods Sold, v) Long-Term Debt-Total, vi) Income Before Extraordinary Items, vii) Inventories – Total, viii) Sales/Turnover (Net), x) Selling, General and Administrative Expenses, xi) Research and Development Expense, xii) quarterly closed price, xiii) data date, xiv) quarterly close date.

2.1 Data Preprocessing

In the data, we initially chose a random subset of 25 companies by gathering the unique values of Ticker Symbol and/or Company Name. Doing this method there was a substantial number of missing values. We then opted to choose the look for companies with no missing values in the column of Quarterly Closed Price and chose a random subset of 25 companies from that list. Doing this, even with the minute amount of data, we at least have more observations than the previous method.

After we get the 25 companies, we checked the data quality of the 12 variables. It turns out that only the variables *iii) Assets – Total*, *vi) Income Before Extraordinary Items* and *xii) quarterly closed price* have a smaller number of missing values, so we dropped other variables in the dataset to avoid the biasedness. To replace the missing values, we used a backfill method for each company. In addition, we derived a variable 'quarter' from the date and a response variable 'stock_returns' using the quarterly closed price.

Overall, we have quarterly data for 25 companies, mainly from 1962Q1 to 2019Q4.

Once we have the features we wanted to use to approach this problem, we split the data into sequences to create input and output data of a sliding window approach for forecasting. The time step we used is 8, which represents 8 quarters.

2.2 Modeling

We developed three deep learning models to predict the stock returns: Feedforward Neural Network, LSMT, and CNN. These three models are all doing the multi-step time series forecasting with multivariate input data.

2.2.1 Feedforward Neural Network:

Feedforward Neural Networks are simple neural networks. They have a similar type of framework as compared to linear regression, however, it has the exception of being able to capture the nonlinear relationships. Feedforward neural networks are generally used for classification of data like images, and not generally used for historical data. However, with manipulation of our time series data we can use Feedforward networks. This works because we are not providing the general input and output values. We are still providing input and output values, except in this case, our input values are a certain amount of timesteps of observations previous to the output data.

For our Feedforward Neural Network, we created a simple dense model as we did not want to create too many dense layers as that would cause overfitting to our data. Having too many dense layers, too many hidden layers, then the data will be overfitting on the training data and when it comes to the validation data, it would not bode too well. We initially started with three dense layers and slowly reduced it to two and one dense layer. After the deciding which number of dense layers used, we added one final dense layer of one since we only have one output.

2.2.2 LSTM:

Long Short-term memory models are made for using with feedback and time data, so it was the easiest model to work with. LSTM networks are well suited to making predictions about time series data as they can leverage past information using memory cells.

To improve our model, we tried changing the window size and the layer size as well as adding more LSTM or other layers onto the sequential model. For the PRCCQ prediction we found a timestep of 8 and a simple LSTM of 100 to be effective. For the Stock Return prediction, we used timestep of 12 and an LSTM of 120. For both predictions, we tried adding more layers of LSTM and using the return sequence function, however that seemed to cause overfitting reducing our test accuracy a lot.

2.2.3 CNN:

For the CNN method, we took advantage of 1D function in Keras to do the convolution computation. We used only one convolutional data with 16 filter and a kernel size of 3. The input sequence of 8 quarters is performed 16 times and it is read in three time-steps at a time. After this, we used a max pooling layer with pool size of 2 to reduce the features and a flatten layer. Two dense layers were applied, one with 50 neurons and a RELU activation function while anther was with 1 neuron as the output layer. We used ADAM for the optimizer and applied MeanAbsolutePercentageError() as the metrics to calculate the errors.

3. Result (First 5 of the 25)

The results from the three model would display in this part. We calculated the mean, std, min, 25%, 50%, 75% and max of the MAPE for each picked company. We have run our models for both the forecasting the Stock Return and the Quarterly Closing Price.

3.1.1 FFNN (PRCCQ):

	BA MAPE	BA VAL_MAPE	CMA MAPE	CMA VAL_MAPE	COP MAPE	COP VAL_MAPE	DIS MAPE	DIS VAL_MAPE	DTE MAPE	DTE VAL_MAPE
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000
mean	16.794519	13.219679	11.985000	12.598897	14.999965	12.795096	20.919505	12.451544	9.168668	8.210564
std	1.258623	2.055917	2.384283	1.794376	5.323958	4.511331	2.541970	1.743589	5.635603	2.145414
min	15.111590	10.457763	10.744922	11.437852	12.313501	10.143296	18.362406	10.254009	7.234123	5.261128
25%	16.047159	11.530826	11.098270	11.680328	13.071098	10.513899	19.733342	11.347426	7.716364	6.568378
50%	16.409464	12.547908	11.396808	11.875088	13.678389	11.385415	20.239102	12.229956	8.012227	7.781499
75%	17.261793	14.614475	11.876237	12.703210	14.566181	13.330541	21.338455	12.922542	9.037271	9.709260
max	24.304295	20.039152	31.952250	21.853954	56.576427	48.282658	40.554947	19.606682	61.259579	16.679686

3.1.2 FFNN (Stock Return):

	BA MAPE	BA VAL_MAPE	CMA MAPE	CMA VAL_MAPE	COP MAPE	COP VAL_MAPE	DIS MAPE	DIS VAL_MAPE	DTE MAPE	DTE VAL_MAPE
count	100.000000	1.000000e+02	1.000000e+02	100.000000	1.000000e+02	100.000000	100.000000	100.000000	1.000000e+02	100.000000
mean	388.464875	9.018057e+05	1.138139e+06	301.774114	8.500353e+05	354.934725	271.306945	121.493014	7.102145e+05	277.087429
std	223.792643	8.720103e+05	9.434827e+05	148.207028	6.414978e+05	120.419198	135.811005	32.265772	1.021241e+06	196.709402
min	189.743240	3.083742e+03	2.579486e+05	155.573593	1.648052e+05	149.896240	186.222244	94.959694	1.001141e+05	167.695297
25%	290.452675	2.913139e+05	7.960705e+05	233.501755	5.468213e+05	257.645210	218.313492	107.940239	4.539958e+05	214.581165
50%	357.791641	8.282214e+05	1.017229e+06	287.812424	7.431948e+05	350.545746	251.138016	116.436035	5.362332e+05	236.939018
75%	425.826187	1.320441e+06	1.252590e+06	332.800552	9.951869e+05	448.334656	286.995316	126.156298	7.056370e+05	283.824936
max	2016.093872	6.651042e+06	9.831613e+06	1561.678955	6.050369e+06	747.781616	1502.840576	405.213776	9.047052e+06	2057.612305

3.2.2 LSTM (PRCCQ):

	BA MAPE	BA VAL_MAPE	CMA MAPE	CMA VAL_MAPE	COP MAPE	COP VAL_MAPE	DIS MAPE	DIS VAL_MAPE	DTE MAPE	DTE VAL_MAPE
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000
mean	17.364760	16.958536	11.666067	12.506177	12.397726	11.622496	20.435187	11.753319	7.683112	7.587975
std	3.059537	3.708585	3.530077	1.754058	2.238805	0.906815	2.665617	2.314482	4.676444	1.528814
min	14.973848	11.809104	9.918372	10.641964	11.160673	9.456348	17.621639	8.842291	6.280543	4.990706
25%	16.253733	14.583540	10.788033	11.832841	11.698596	11.005472	18.785334	10.579566	6.733569	6.275139
50%	16.930554	16.216039	11.272742	12.216736	12.078443	11.572633	19.932789	11.144541	7.050038	7.605839
75%	17.719303	18.269999	11.641784	12.705415	12.432101	12.010159	21.163120	12.020841	7.398752	8.680420
max	45.303516	33.212734	45.601109	27.668116	33.420918	14.860488	36.841282	24.647840	53.043324	12.431164

3.2.2 LSTM (StockReturn):

	BA MAPE	BA VAL_MAPE	CMA MAPE	CMA VAL_MAPE	COP MAPE	COP VAL_MAPE	DIS MAPE	DIS VAL_MAPE	DTE MAPE	DTE VAL_MAPE
count	100.000000	1.000000e+02	1.000000e+02	100.000000	1.000000e+02	100.000000	100.000000	100.000000	1.000000e+02	100.000000
mean	278.373789	1.409600e+06	6.377270e+05	177.954557	5.871399e+05	294.310900	236.759369	123.769346	4.081491e+05	167.173303
std	86.805132	1.050003e+06	4.327452e+05	39.604703	3.105978e+05	108.878607	48.604346	15.654529	1.960131e+05	43.269477
min	115.133301	3.695406e+04	9.487715e+04	111.107918	1.095283e+05	100.543091	166.787582	103.548088	9.924547e+04	98.009628
25%	216.226929	3.822532e+05	3.248100e+05	142.531742	3.520035e+05	208.291851	194.006092	111.661860	2.709799e+05	129.908787
50%	281.763412	1.186781e+06	5.239678e+05	180.338982	5.110648e+05	290.387436	221.217255	116.857098	3.901655e+05	163.735603
75%	332.919846	2.316080e+06	7.927743e+05	202.925808	7.537193e+05	379.395226	277.771431	135.576706	5.048496e+05	197.601513
max	587.562744	3.845696e+06	2.182441e+06	293.583130	1.666492e+06	534.873169	340.835022	165.984543	1.092755e+06	270.837158

3.3.1 CNN (PRCCQ)

	BA MAPE	BA VAL_MAPE	CMA MAPE	CMA VAL_MAPE	COP MAPE	COP VAL_MAPE	DIS MAPE	DIS VAL_MAPE	DTE MAPE	DTE VAL_MAPE
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000
mean	15.307662	17.105733	15.721104	17.944395	15.348411	17.048730	15.588536	17.354278	15.506343	17.403360
std	1.034856	2.046208	1.818769	3.599642	1.171432	1.477935	0.972329	1.553538	1.162364	2.883858
min	13.972986	14.769111	14.134356	15.033408	14.213552	14.939934	14.367871	15.392678	14.116027	14.853895
25%	14.680523	15.787355	14.685185	16.100735	14.649650	16.011806	14.852395	16.280994	14.705063	15.867422
50%	15.081217	16.679216	15.102663	16.726450	15.066828	16.749570	15.370565	16.981009	15.290743	16.717206
75%	15.781825	17.577335	15.882224	18.180110	15.800555	17.589958	16.080242	18.070036	16.108461	17.922693
max	21.887304	28.703806	26.794104	40.265896	23.878508	22.217237	21.473331	23.803232	22.767342	33.236523

3.3.2 CNN (Stock Return)

	BA MAPE	BA VAL_MAPE	CMA MAPE	CMA VAL_MAPE	COP MAPE	COP VAL_MAPE	DIS MAPE	DIS VAL_MAPE
count	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000	100.000000
mean	328799.340313	141693.725430	314128.131563	83864.566543	341503.224062	100127.176270	339891.152813	103953.665654
std	72311.974673	63453.875082	55155.096143	36758.287244	63958.720796	43696.697097	66455.051202	38619.227498
min	206308.109375	34394.277344	187675.718750	12569.835938	210736.812500	31584.970703	190306.328125	11295.159180
25%	269318.625000	91989.982422	268554.351562	58942.333984	292441.726562	71791.816406	295824.421875	76937.570312
50%	318345.234375	133902.093750	313989.203125	79659.335938	345836.437500	86838.417969	345851.171875	96906.367188
75%	376639.671875	189478.187500	348488.882812	102952.765625	387647.546875	125079.419922	385584.375000	132730.468750
max	589517.250000	297015.250000	493461.000000	285400.312500	505664.656250	261915.031250	597406.062500	211197.515625

4. Conclusion

To recap, Feed Forward Neural Networks (FFNN) are not originally intended to handle time series forecasting. For us to use this model, we had to engineer the data to an acceptable input and output. FFNN are simple models and is the least complex of the ones we have tried. Saying this, our FFNN model performed very poorly. CNN model results in the worse MAPE while compare to other two approaches. This might be caused by the overfitting in our model. In conclusion, LSTM seems to be performing the best out of the different model types. This is to be expected since LSTM is designed to work with time sensitive data and our dataset contained date trends. However, even with LSTM our results weren't good since the original dataset had many null values and we didn't know how to best fill them since we were missing industry knowledge.