machine learning and the market

Justin Fischer
GA DSI11-DEN

problem statement

- Predict short term price changes for publicly traded stocks
  - Specifically the 30 companies in the DJIA
  - Just using price and volume data combined with twitter sentiment

efficient markets hypothesis

- The price of a stock reflects all known information
- 3 Forms:
  - Weak
    - No returns earned based on historical prices; No patterns
  - Semi-Strong
    - Share prices adjust quickly and unbiased, no returns can be earned by trading on new information
  - Strong
    - Reflect all information; No one can earn excess returns
    - Impossible if insider trading laws exist

data

- Downloaded tick data for each company in the Dow Jones Industrial Average
- 9/3/19 - 12/31/19
- Removed weekends and non-trading hours
- Pulled tweets for the same time period using a query that is ‘$’+ticker

making the twitter scraper work for me

- GetOldTweets3 kept timing out and giving too many requests errors
- Had to add a few lines to the library to make it sleep and retry when errors were detected
- Used the time.sleep() function and a ratelimit library

data processing

- Calculated a sentiment score for each tweet
- Resampled each dataset by minute and second
- Calculated a weighted mean for the sentiment score
- Calculated percent changes for the high and mean price in each period
  - 1, 2, 3, 4, 5, 10, 15, 30, 60

quick data summary

	average	low	high
tweet count	11,149.35	1,321 (trv)	81,967 (aapl)
trade count	255,969.5	45,818 (trv)	779,963 (aapl)

Average sentiment - .127

target summary

	periods	down	flat	up
seconds	2,012,484	2.5%	95%	2.5%
minutes	33,625	38.7%	22.3%	39%

goal

- The actual price does not matter here at all
- Predicting a short term movement in a direction is enough to build an effective trading model off of
    - Think *Flash Boys* by Michael Lewis instead of Warren Buffett

- Created a target variable based on the percentage change from the previous period

model

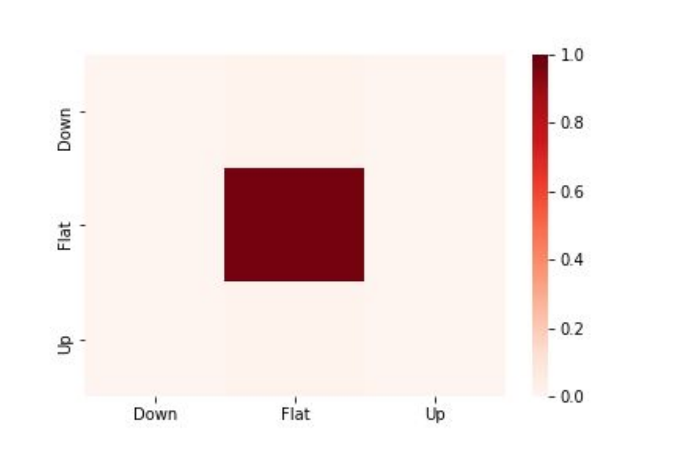
- Recurrent Neural Network on time series data
    - Used because it excels at sequential data
- Parameters:
    - Length - 5
    - Batch - 512
    - Hidden Layers:
        - GRU - 32
        - GRU - 16
        - Dense - 8
        - Dense - 4
    - 100 epochs

Sigh....unsaved results

- After testing it, for some reason the history object didn’t save.
- Accuracy numbers may not be the most accurate

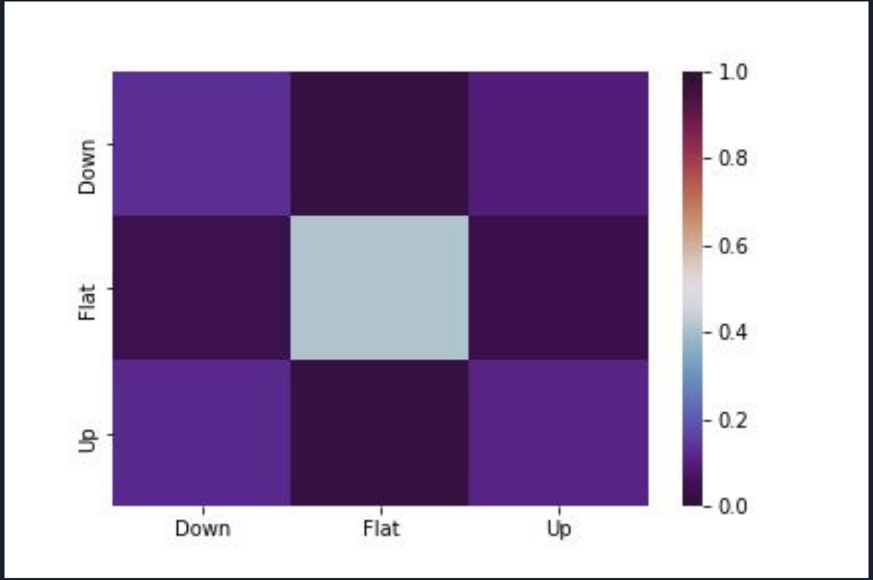
seconds

- From quick analysis of the results that did save, since the target was so imbalanced, it just picked ‘flat’ for each tick.
- A ‘flat’ prediction means we don’t do anything since you can’t make money on a stock that doesn’t move
- So, even an accuracy score of close of 95% means nothing since we just predict it to stay the same



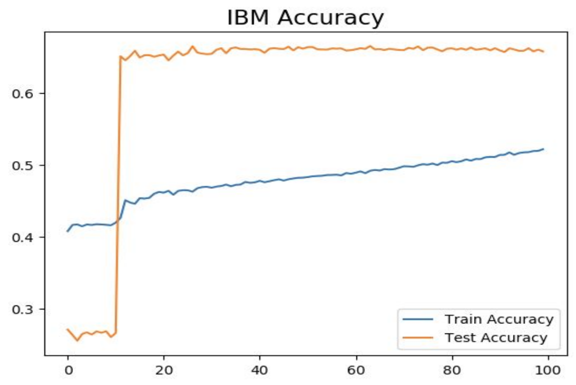
minutes

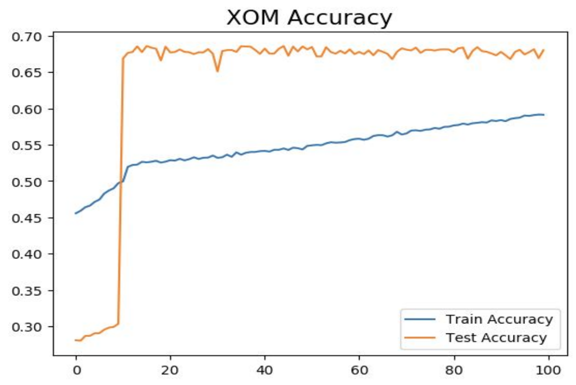
- We have stopped picking flat periods
- Still are not perfect or close enough since each corner is very similar

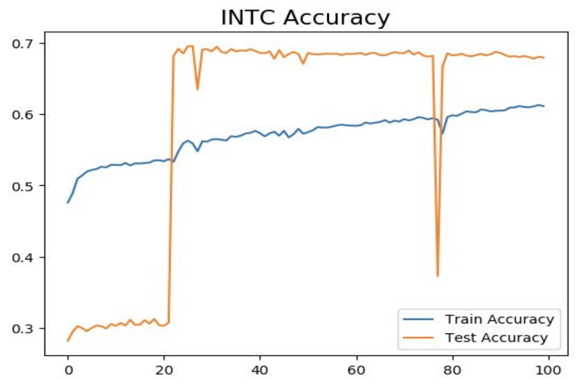


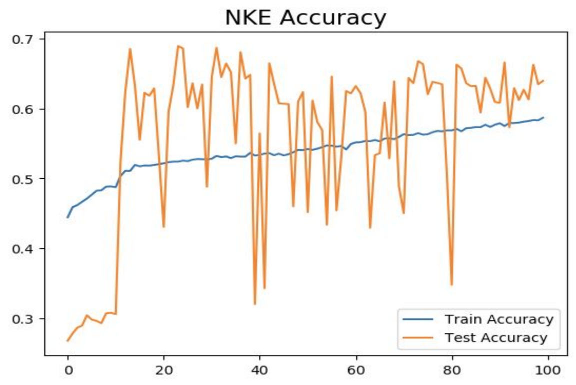
accuracy

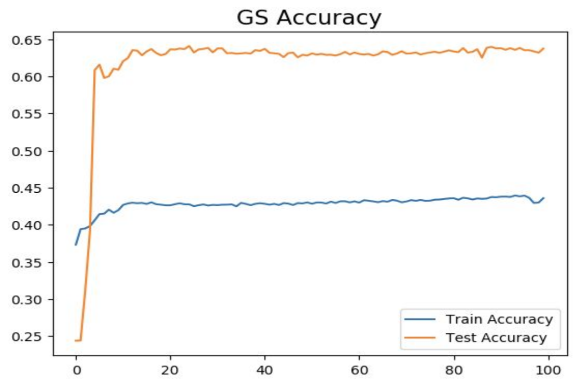
- Most companies have a 60-70% accuracy score on the test data
- A few still had the potential to continue to increase with more iterations
- Others were very unpredictable and bounced around

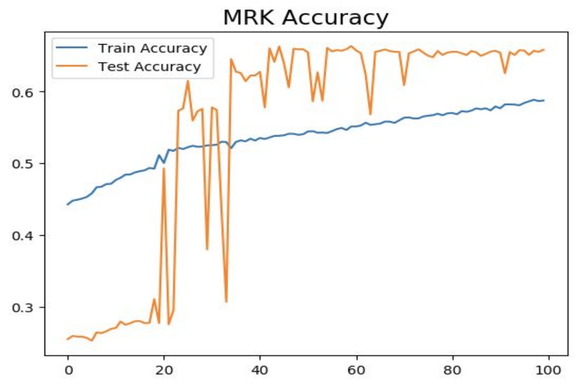

IBM Accuracy
0.6
0.5
0.4
0.3
0
20
40
60
80
100
Train Accuracy
Test Accuracy


XOM Accuracy
0.70
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0.30
0
20
40
60
80
100
Train Accuracy
Test Accuracy


INTC Accuracy
0.7
0.6
0.5
0.4
0.3
0
20
40
60
80
100
Train Accuracy
Test Accuracy

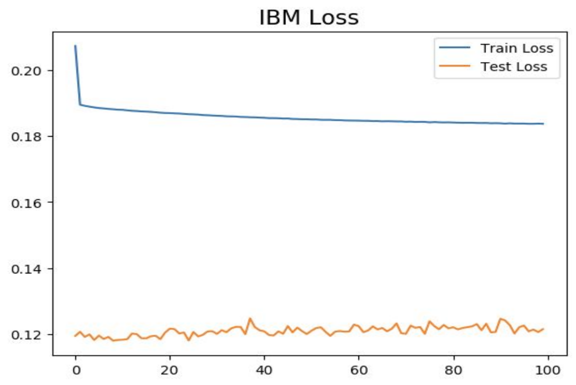

NKE Accuracy
0.7
0.6
0.5
0.4
0.3
0
20
40
60
80
100
Train Accuracy
Test Accuracy

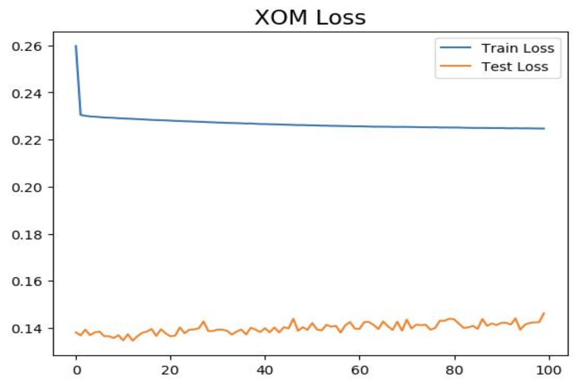

GS Accuracy
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0.30
0.25
0
20
40
60
80
100
Train Accuracy
Test Accuracy

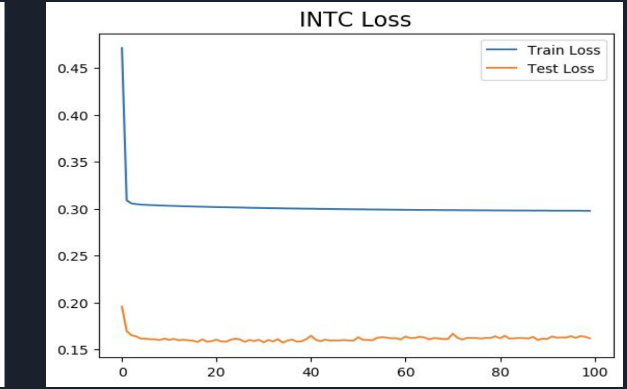

MRK Accuracy
Train Accuracy
Test Accuracy
0.6
0.5
0.4
0.3
0
20
40
60
80
100

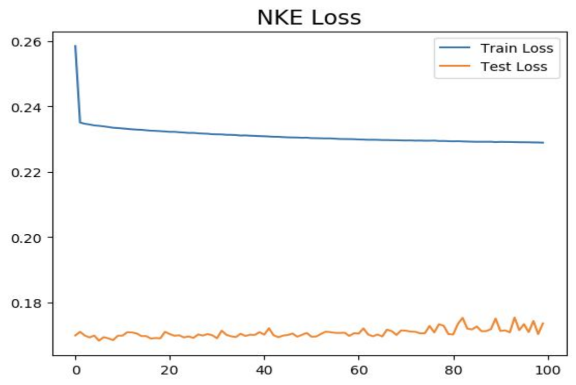
loss

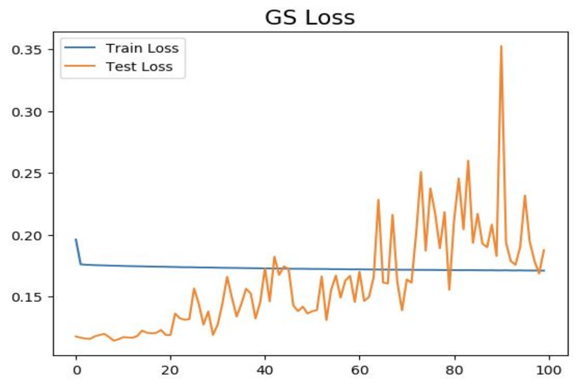
- The shapes are really strange
- They seem to have a slight curve still, a number of the models could still get better
- Unclear how much better we can make the models with the current parameters

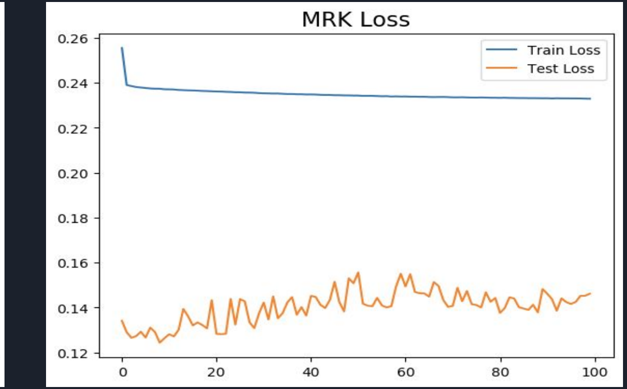

IBM Loss
Train Loss
Test Loss
0.20
0.18
0.16
0.14
0.12
0
20
40
60
80
100


XOM Loss
Train Loss
Test Loss
0.26
0.24
0.22
0.20
0.18
0.16
0.14
0
20
40
60
80
100


INTC Loss
Train Loss
Test Loss
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0
20
40
60
80
100


NKE Loss
Train Loss
Test Loss
0.26
0.24
0.22
0.20
0.18
0
20
40
60
80
100


GS Loss
Train Loss
Test Loss
0.35
0.30
0.25
0.20
0.15
0
20
40
60
80
100


MRK Loss
Train Loss
Test Loss
0.26
0.24
0.22
0.20
0.18
0.16
0.14
0.12
0
20
40
60
80
100

conclusion

- Most of the correct predictions are for no movement
    - This makes it really hard to act upon
- Without the ability to really dig into the results, it is hard to have a firm conclusion
- It seems like it isn’t accurate enough to run with money at this time
- Another training session with a slightly modified dataset could be the key

change prediction

- Predict multiple periods, instead of one
    - If prediction is flat, check next period to remake that prediction

next steps

- Different resamples/intervals
  - Combination of a few
- Lesser known and smaller stocks
- Different type of security
  - Commodities
- Use other data
  - Foreign exchange rates
  - Different volume data
  - Bid/Ask spreads and depth
- Better language processing for news and tweets
- Add weights to twitter accounts for those who either:
  - Run a company
  - Famous/respected analysts/talking heads
- Cross-evaluate companies instead of just looking at its own prices
- Try an LSTM model