**Overall Blueprint:**

*This is a very general overall outline of what needs to be done.*

Python: Grab data from APIs, clean data, store it into SQL or Mongo db. (I prefer Mongo because ORMs suck but SQL is fine too). The DB should be hosted but if not, SQLite works too.

Python/Pandas: Rescale the data.

Python or Jupyter Notebook (Keras): Feed data into AI. AI will then poop out results.

Put results into DB.

Python: Run Flask server, create API that allows access to AI results and host website on Heroku or AWS or google cloud. Honestly, if there’s time just learn to port Flask over to Django.

Javascript and HTML: Retrieve AI results and visually show the results.

**More detailed, specific instructions:**

**Data Collection:**

Find stock market data since 1980s or 1990s (CSV, API, whatever) for the big Indexes (Dow Jones, and S&P 500). Daily data would be the best. Daily would be the one we use for this project but hourly can be something we use later.

Clean the data so that we can have at least have the following information for each day (Using numbers from MSFT as an example):

Open 141.01

High 141.65

Low 138.25

Mkt cap 1.06T

P/E ratio 27.54

Div yield 1.46%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Date | Open | High | Low | MktCap | P/E Ratio | Div Yield | Daily Volume |
| 01/01/2001 | 78 | 90 | 20 | 9393 | 32 | 2 | 100000 |

In the pandas, each row should look something like this, above.

Clean the data, make sure they’re all in numbers/float format, and name it

SP500\_data\_df, Nasdaq\_data\_df, or DJ\_data\_df

In addition to the raw data, there should also be moving averages and historical data.

For example:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1-Day ago | 3-Days ago | 5-Days ago | 7-Days ago | 9-days ago | 11-days ago |
| 50 | 32 | 63 | 83 | 29 | 50 |

This would be appended/merged to the first

**Data Scaling:**

This data must then be compressed into a format that the neural network can process. This means all data should be turned into some number between 0 to 1.

For example, if we are look at data from 9/3/2015 and the price was 100, we will find the maximum and minimum price from the last 365 days to determine the range of the scale.

Our current price: 100

Max price from 365d range: 150

Min price from 365d range: 50

Our **scaled price data**: 0.5

To account for outliers, do not accept data that exceeds 3 standard deviations of the 365d data set.

**Generating training data:**

IF NO RNN: Once the data has been cleaned and scaled, it’s time to generate training data. We will need to decide what is the “range” of data we will want the AI to consider. Should the AI look at what happened the last day, last week or last month?

Create training data in the following format:

Decision Date (MetaData), 52week high, 4 week high, Today’s vol, etc etc etc

The last column, aka the test column, should say YES or NO. Aka should the AI have bought or sold the stock on that date?

*This must be decided on how much time we’re giving the AI to think. The idea is “Would it be a good idea to buy or sell right now if the next opportunity we have to buy or sell will be a day, week, or month from now?”*

**AI:**

The AI NN will be a standard feed forward NN that will be fed data from a specific P.

The NN will receive all the data from P, which will all be scaled to some value from 0 to 1 or -1 to 1. The NN will then decide if it’s time to buy or sell.

The AI will output its answer and it’ll be added to a row that will show all the information it was given (before it was scaled), the date of its decision, what its decision was, and how much gains/loss the decision would’ve resulted in.

**Front end:**

What was the average % gains or loss by the AI?

How much money would you end up with if you have given an AI $10,000 since 1980?

Show the performance differences between each AI in some kind of visual comparison.