OSIG Research & Portfolio Analysis

Capstone Group 46

What is OSIG?

- Oregon State Investment Group
- Manages approximately $3.5 million across three portfolios.
- Analysts develop workbooks, reports and presentations on a company that they pitch to the group once a term.
- Portfolio managers adjust equity weights in their portfolio, buy/sell stocks, and coordinate with management on companies they would like pitched.

Goals

- Create research and portfolio analysis software for the Oregon State Investment Group.
- This software should aim to help analysts complete their workbooks more efficiently, as well as to give portfolio managers insights on the best/worst positions in the portfolio.
- Use machine learning to provide portfolio insights.
- Create a central location for OSIG resources, including calendar/schedules, pitch voting results, important links and files, ect.

Portfolio Analysis

- Portfolio managers do research and choose some fixed assets
- Managers still need to figure how to weight assets appropriately

Given assets $A_1, A_2, ..., A_n$ choose weights such that

$$w_1, w_2, ..., w_n \in [0, 1]$$
$$\text{subject to } \sum_{i=1}^{n} w_i = 1$$

How to proceed?

- Read the literature
- Common method is setting up an optimization problem
- Decide against that
- The Sharpe Ratio?

Sharpe Ratio

$$S = \frac{E[R_p] - E[R_f]}{\sigma_p}$$

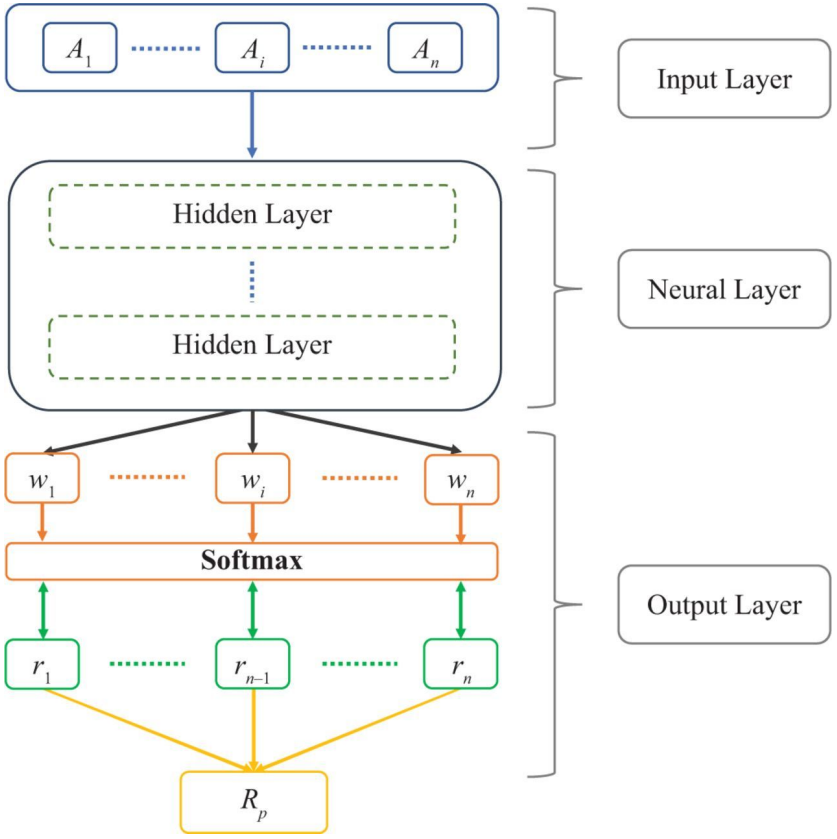
- $E[R_p]$ Expected return of portfolio
- $E[R_f]$ Expected return of risk free rate (usually fixed)
- σ_p Standard deviation of portfolio
- Tells us excess returns per unit of risk

Current Architecture - High Level Process Flow

1. Investor picks a set of assets for a new portfolio they want to create.
2. Stock prices are grabbed for the past 5 years.
3. Neural Net trains on the history of these stock prices and optimizes based on the Sharpe Ratio.
4. Recommended weights for each asset is returned once the model is finished training.

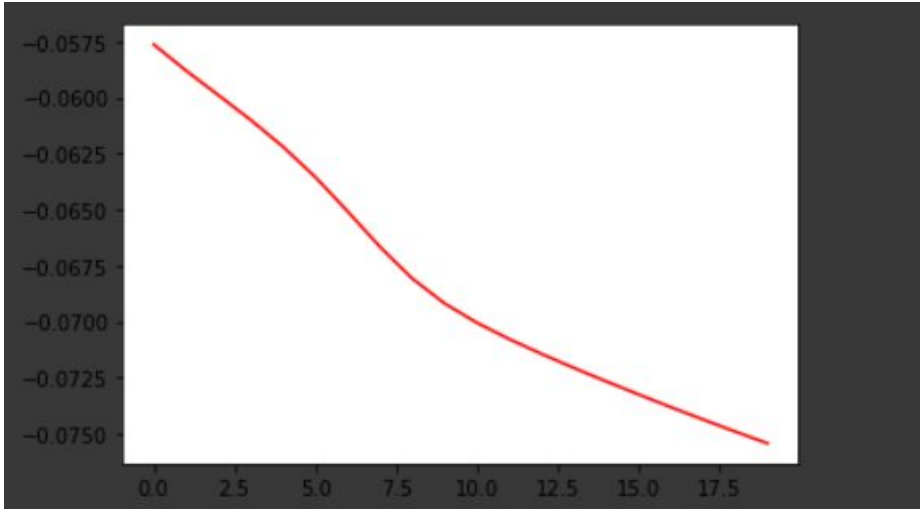
Current Architecture

- PyTorch For ML
  - Need Custom Loss Function
  - Autograd is easy
- PyEx
  - Lots of free data
  - Comes packaged in pandas



Results, Issues, and Future Improvements

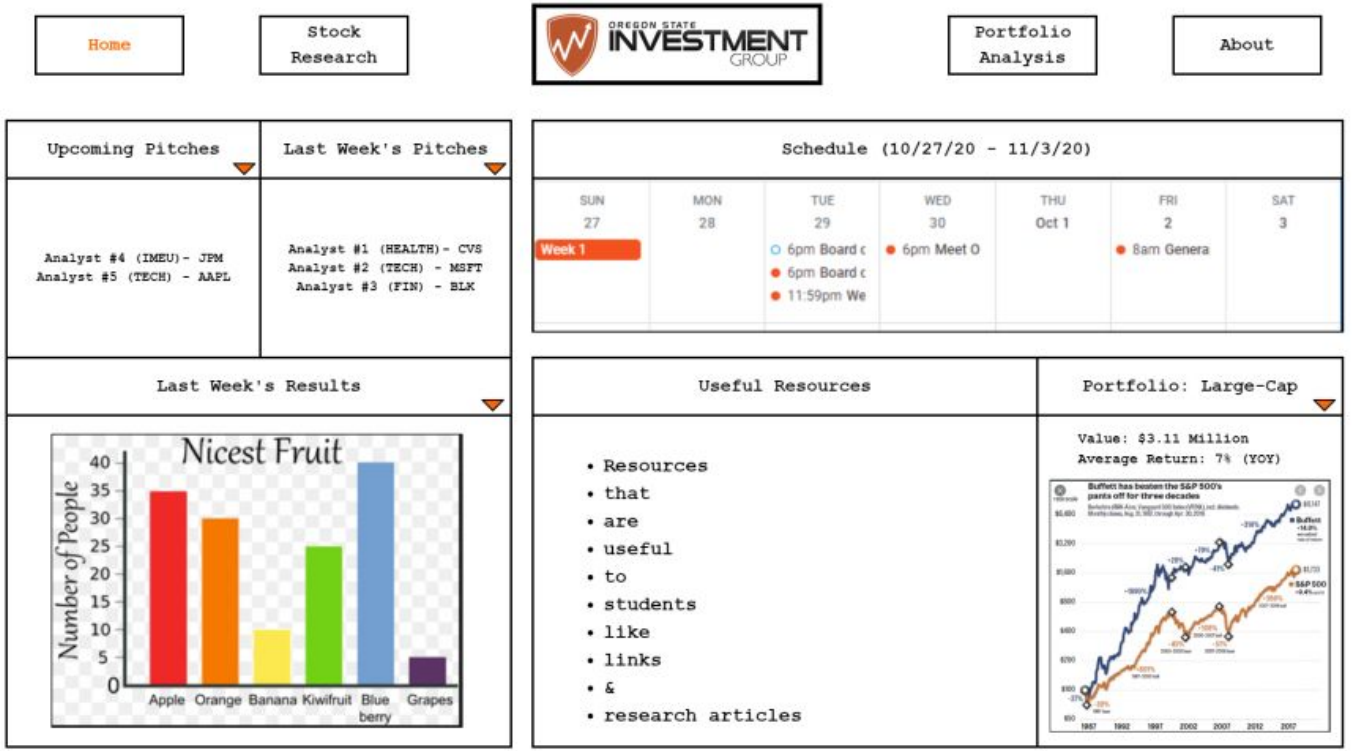
- Add visualizations so that we can track losses of the model easily
- Implement a way to measure how well our model performs testing data
- Come up with new features to add in the data so that it doesn't only include stock history
- Limit weights so that the model doesn't heavily weigh certain stocks.



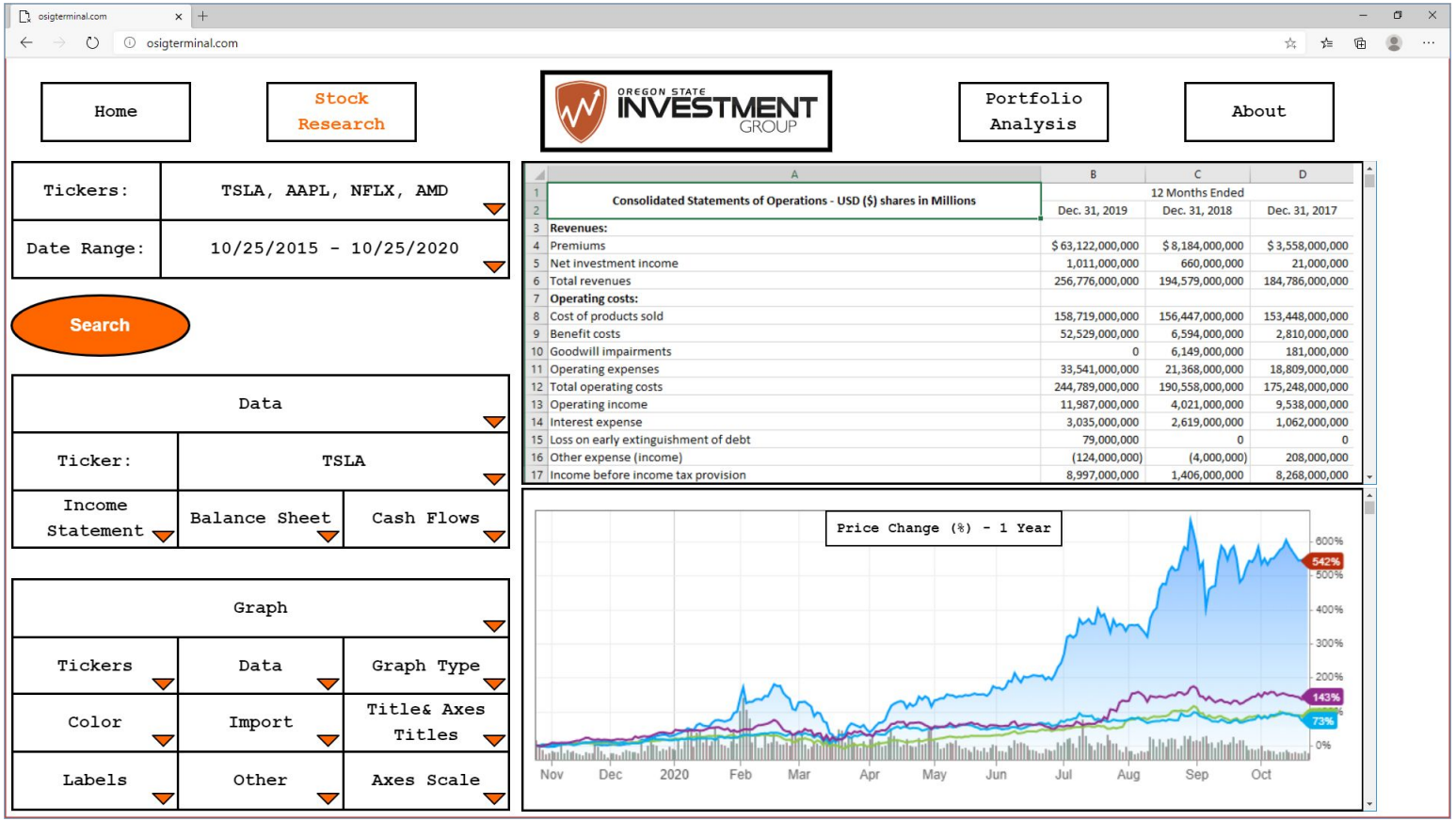
Web Application Design

- Django and SQL will be used for the backend development
- React, Bootstrap will be used for the frontend development

Design Goals: Home Page

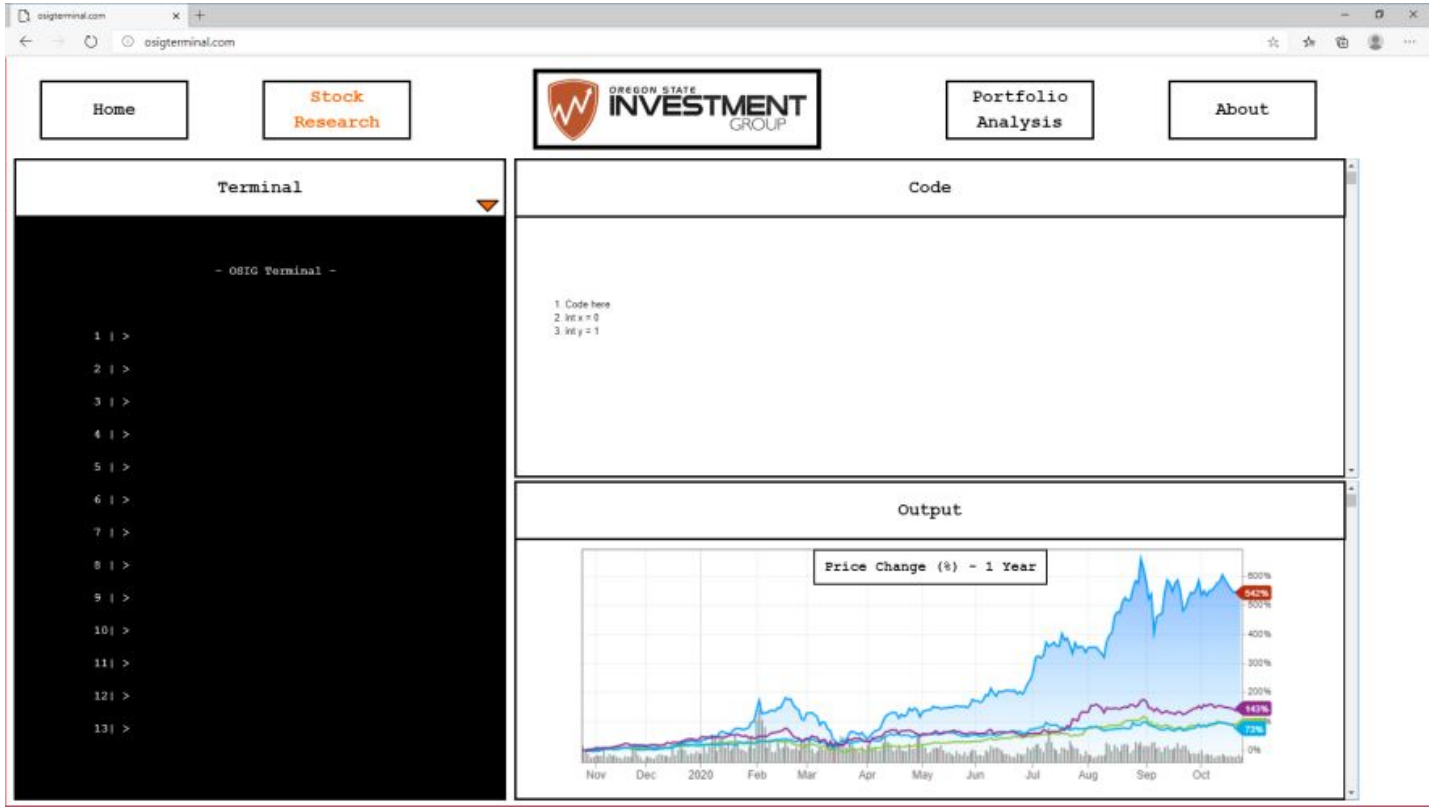


Design Goals: Stock Research Page



	A	B	C	D
1	Consolidated Statements of Operations - USD ($) shares in Millions	12 Months Ended		
2		Dec. 31, 2019	Dec. 31, 2018	Dec. 31, 2017
3	**Revenues:**			
4	Premiums	$ 63,122,000,000	$ 8,184,000,000	$ 3,558,000,000
5	Net investment income	1,011,000,000	660,000,000	21,000,000
6	Total revenues	256,776,000,000	194,579,000,000	184,786,000,000
7	**Operating costs:**			
8	Cost of products sold	158,719,000,000	156,447,000,000	153,448,000,000
9	Benefit costs	52,529,000,000	6,594,000,000	2,810,000,000
10	Goodwill impairments	0	6,149,000,000	181,000,000
11	Operating expenses	33,541,000,000	21,368,000,000	18,809,000,000
12	Total operating costs	244,789,000,000	190,558,000,000	175,248,000,000
13	Operating income	11,987,000,000	4,021,000,000	9,538,000,000
14	Interest expense	3,035,000,000	2,619,000,000	1,062,000,000
15	Loss on early extinguishment of debt	79,000,000	0	0
16	Other expense (income)	(124,000,000)	(4,000,000)	208,000,000
17	Income before income tax provision	8,997,000,000	1,406,000,000	8,268,000,000

Design Goals: Python Development Page



Progress So Far: Web Application

- Documentation has been created on how to start the server to see the current home page. This documentation is located in the Wiki on the git repo.
- Documentation on what software to use and how to install it needs to be compiled together in the wiki.
- The home page and navigation bar are the current focus of development.

- Next Goal:
  - Finish home page
  - Using the IEX Financial API to retrieve data to display to the frontend
  - Connect to an SQL database and retrieve and display data from basic table in database