OSIG Research & Portfolio Analysis

Capstone Group 46

Project Members and Roles

Ashyan Rahavi: Machine learning development

Blake Cecil: Machine learning development

Joseph Noonan: Website development

Braeden Kuether: Project management, website design, workbook automation, project partner

What is OSIG?

- Oregon State Investment Group
- Manages approximately $3.7 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.

Project Overview

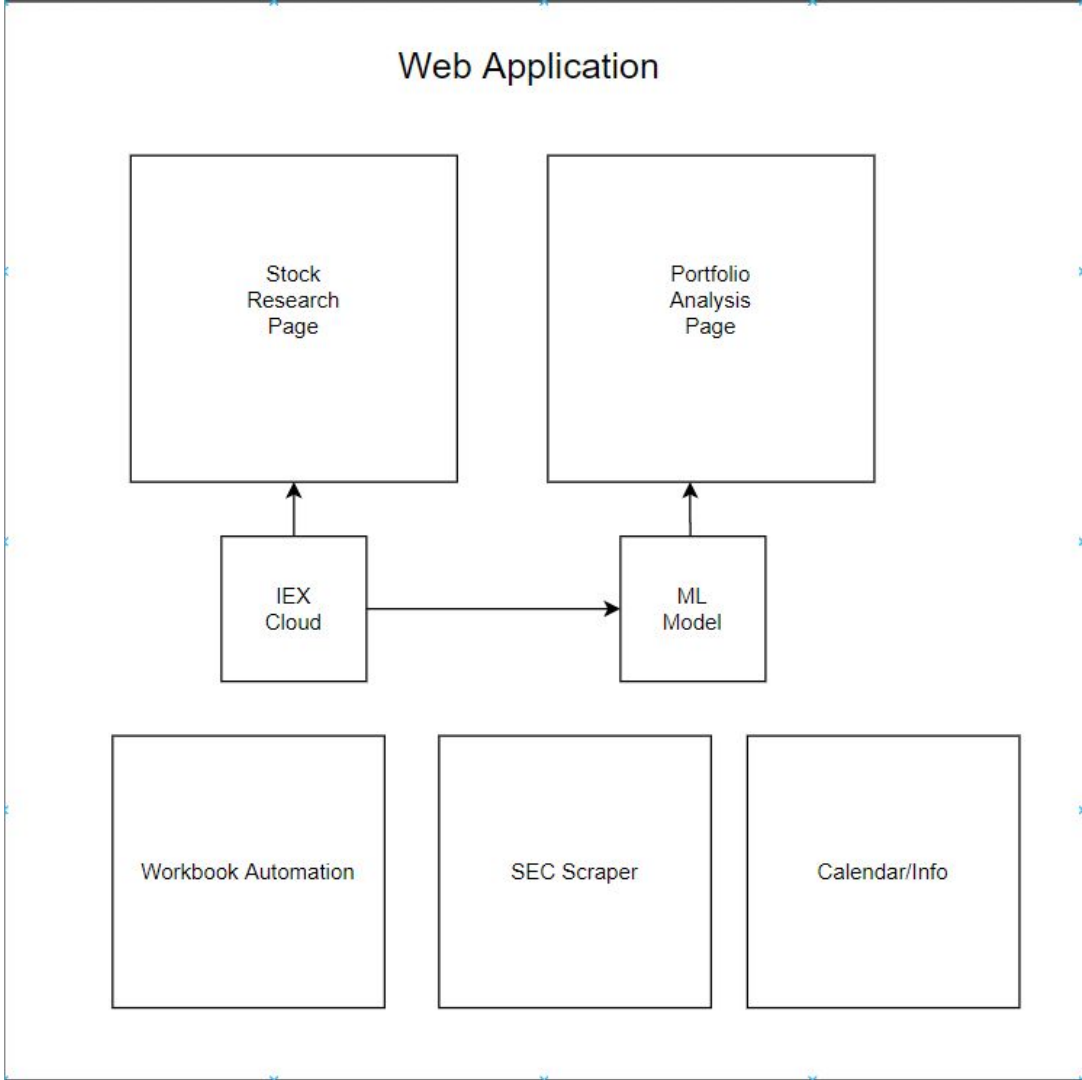
- Collect financial data from the IEX Cloud API
- Use machine learning to help manage portfolio decisions.
- Create research and portfolio analysis software for the Oregon State Investment Group.
- Create a central location for OSIG resources, including calendar/schedules, pitch voting results, important links and files, ect.
- Automate information gathering and model input in the OSIG workbook.

Overview Continued

- Web Application using react and django
- No build required on user end
- Website is limited for non OSIG members

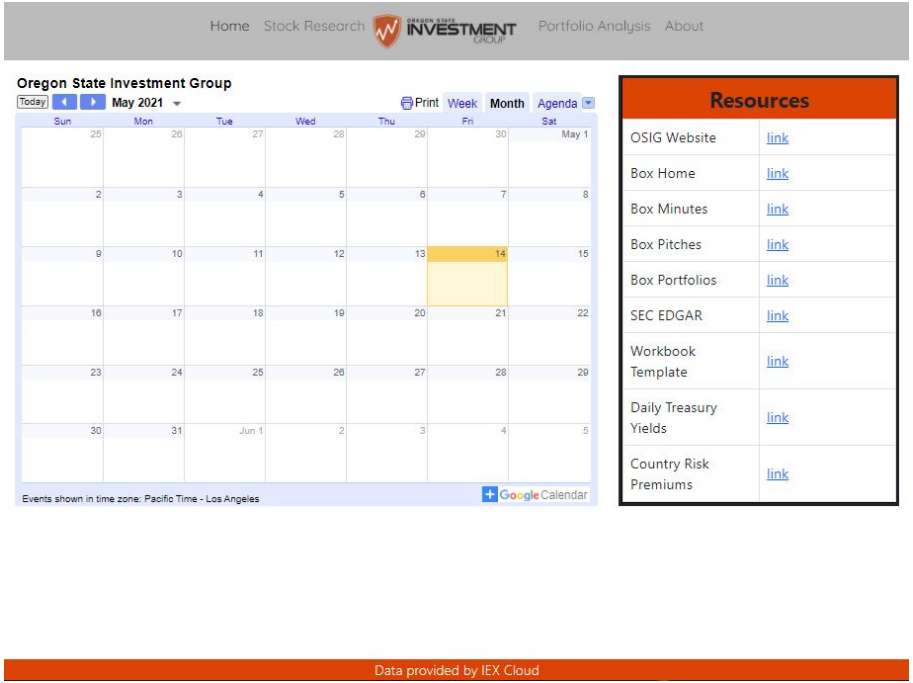
Functional Diagrams

- Stock research page
- Workbook Automation
- SEC Scraper
- Calendar/Group Info
- Portfolio analysis page



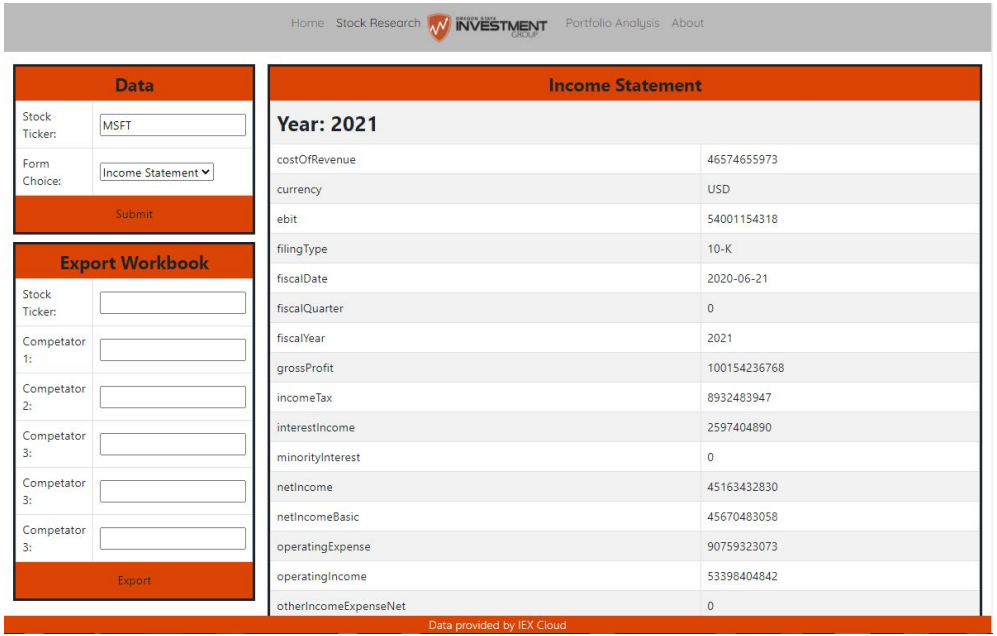
Home Page

- OSIG management will handle the updating the results of the pitches on the calendar
- OSIG will also handle putting the upcoming and past pitches on the calendar on the home page
- The calendar is currently functioning as intended
- Resources are used to help complete the workbook, such as the SEC EDGAR website, the workbook template, country risk premiums, and daily treasury yields
- Resources also contain links to the group Box and website



Stock Research Page

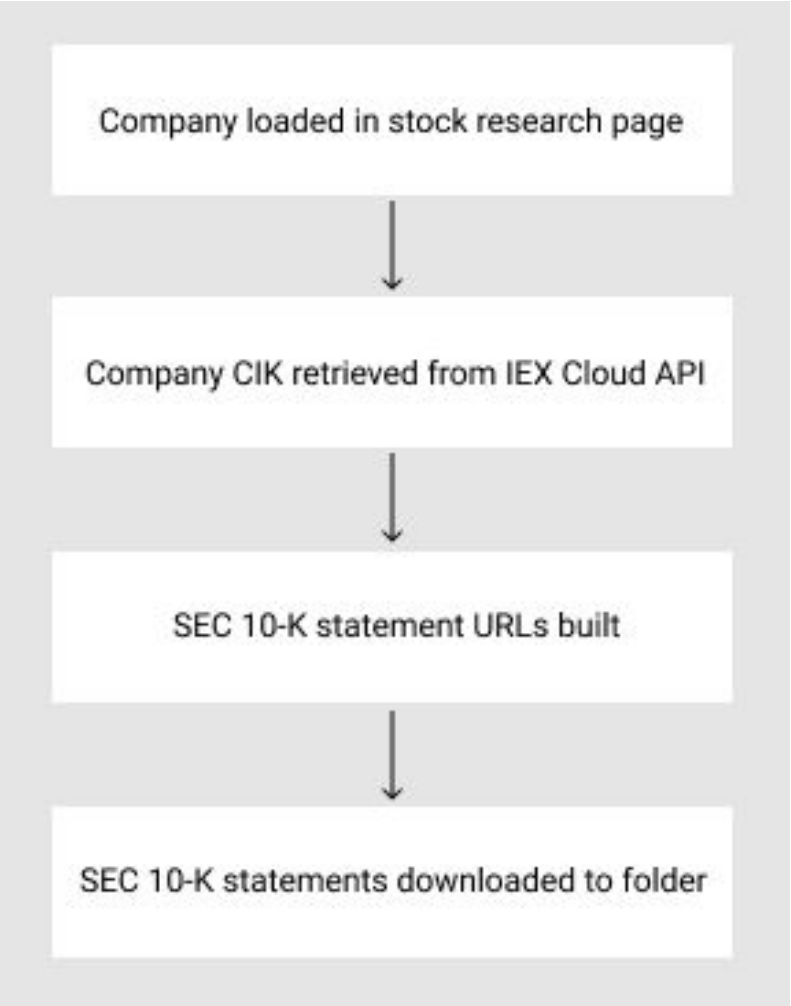
- Data provided includes income statement, balance sheet, cash flows, and summary statistics
- Any company covered in the IEX Cloud API can be displayed
- Can enter competitors and download relevant statements in a zip file
- Workbook can use the zip file to complete certain sections



Workbook Automation

- OSIG uses SEC 10-K statements to build workbooks

- This simple script retrieves those statements in excel format

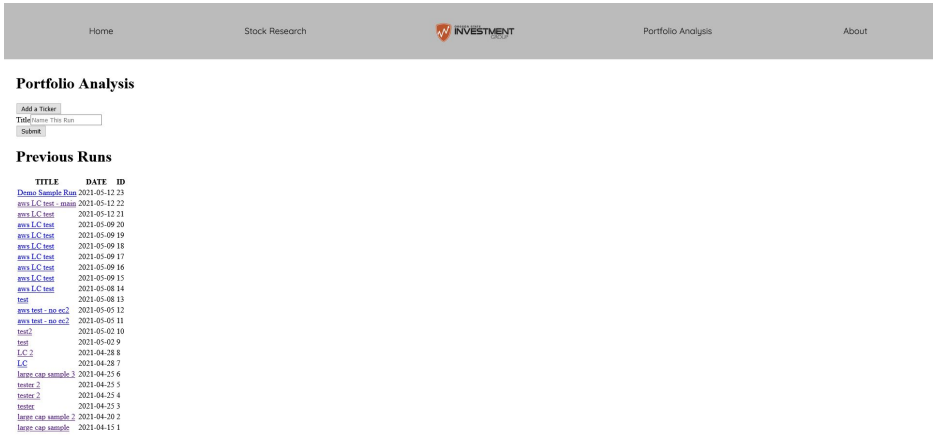


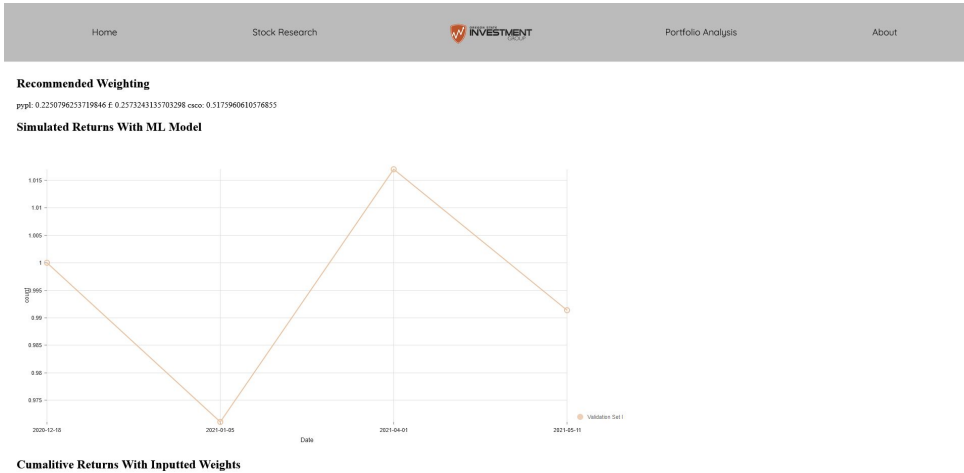
Current Architecture - High Level Process Flow

1. Investor picks a set of assets for a new portfolio they want to create.
2. Stock Features 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.

Portfolio Analysis Page

- Create portfolios with tickers from IEX cloud
- Analyze them on the back end and send recommendations, stats visualizations etc





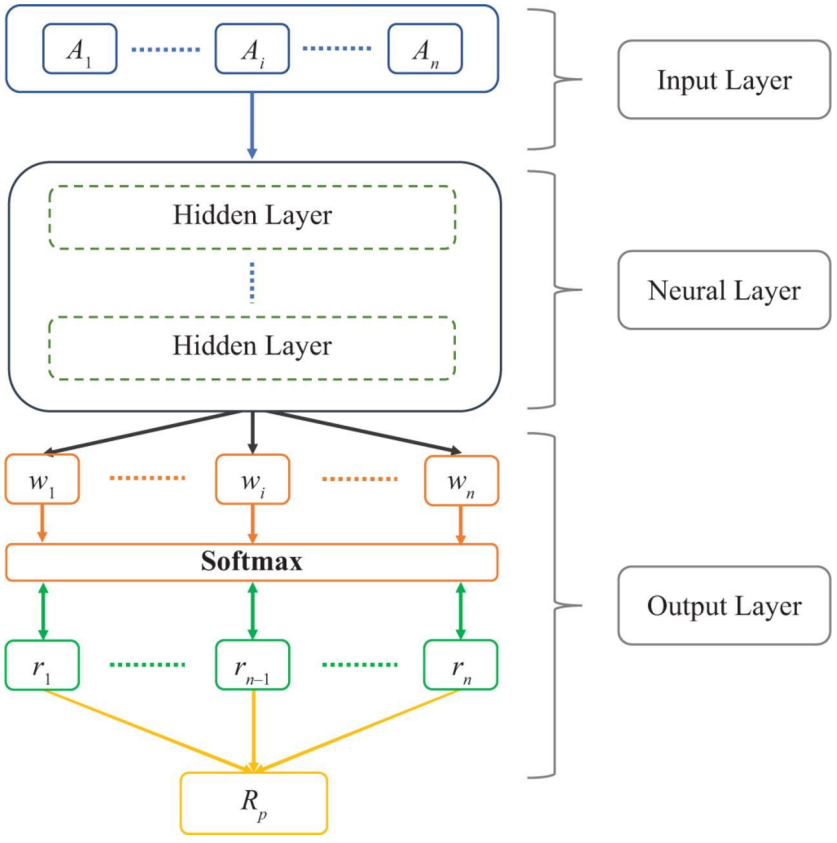
Functional Diagram

- LSTM network.
- Sharpe ratio as loss function.

$$R_{p,t} = \sum_{i=1}^{n} w_{i,t-1} r_{i,t}$$

$$E(R_{p,t}) = \frac{1}{T} \sum_{i=1}^{T} R_{p,t}$$

$$L = \frac{E(R_{p,t})}{\sqrt{\mathrm{var}(E(R_{p,t}))}}$$



ML Model Details

```python
def sharpe_loss(weights, returns):
  # weights batch * time * assets
  # returns batch * time * assets
  #print(weights.shape, returns.shape)

  #row wise dot product
  R = torch.sum(weights*returns,dim=-1)
  ER = torch.mean(R,1)
  STD = torch.std(R,1)
  ratio = torch.sum(ER/(STD+1e-6))
  return -ratio
```

```python
class Net(nn.Module):
    def __init__(self,NUM_FEATURES,NUM_ASSETS,TIME_PERIOD_LENGTH):
        super(Net, self).__init__()
        self.time = TIME_PERIOD_LENGTH
        self.input = nn.LSTM(NUM_FEATURES, 64, 1, batch_first = True)
        self.lin = nn.Linear(64,NUM_ASSETS)
        self.soft_out = nn.Softmax(dim=2)
```

Future Work

- Design more robust ML model
- More analysis tools (price prediction, asset selection, etc)
- More data collection
- Automate more tasks