

**FLORIDA**

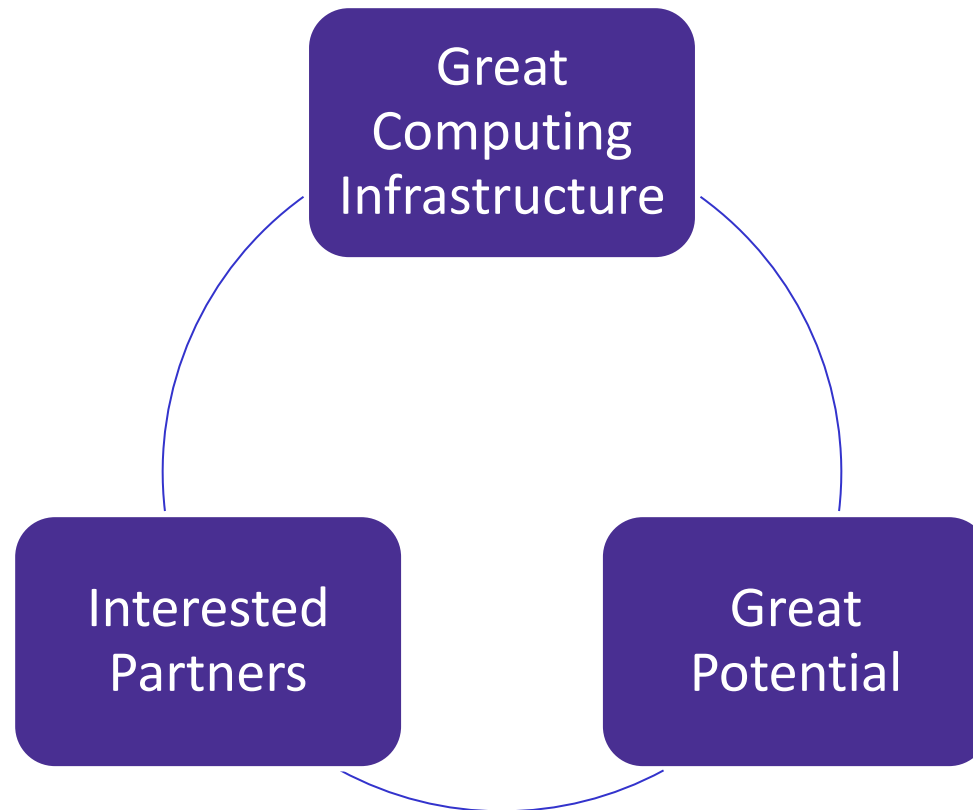
---

**POLYTECHNIC**  
**UNIVERSITY**

**Big Data and Health Informatics Lab**  
**Hedge fund Project**

**Thanos Gentimis**  
**Jay Stannard**  
**Jean-Louis Jean-Patrice**

# What we have



# What we didn't have

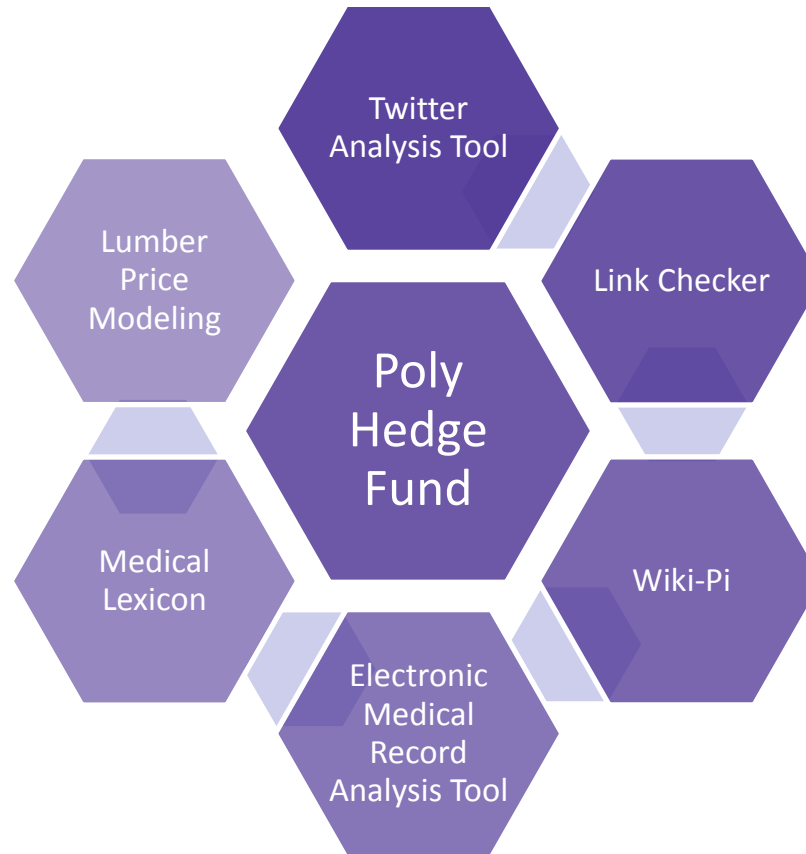
Clearly  
Defined  
Project



Health  
Informatics  
Data

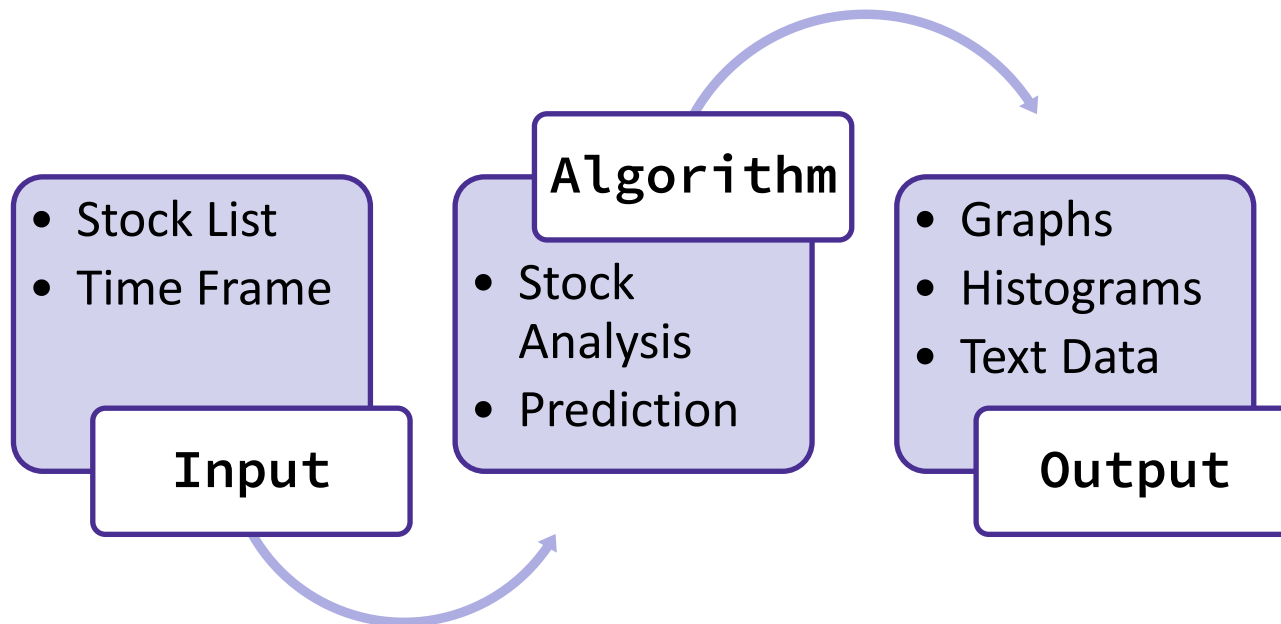
Expensive  
Software

# List of Projects



25 Active Students in the Lab

# Poly Hedge Fund



# Poly Hedge Fund – Python

The screenshot displays the PyCharm Community Edition 5.0.1 interface. The main editor window shows the file `StockAnalysis.py` with the following Python code:

```

from yahoo_finance import Share #Imports Yahoo Finance library used for getting stock data
import json, datetime #json library used for formatting YHOQ data and datetime for time
import pandas as pd #Powerful data analytics library used for time-series analysis and RMA calculations
import matplotlib.pyplot as plt #generates plots for all stocks that are candidates
import statsmodels.formula.api as sm
import os #Necessary for save function

def stockdates(): #datetime function generator that parses the all dates as the same format

    date = ('start': '0001-01-01', 'end': '9999-01-01') #formats the start and end date
    defaultDate = input("Would you like to use today's date and go back two years? ") #boolean value for analysis range
    defDate = datetime.datetime.now() #current date fetcher
    if defaultDate.capitalize() == "Yes": #singularizes user input and runs loop for boolean defaultDate input
        date['end'] = str(defDate.year) + "-" + str(defDate.month) + "-" + str(defDate.day) #concatenates end date input
        # and parses them into a list with definitions
        date['start'] = str(defDate.year - 2) + "-" + str(defDate.month) + "-" + str(defDate.day) #concatenates start
        # date input and parses them into a list with definitions (will be used later for further analysis)
    elif defaultDate.capitalize() == "10":
        date['end'] = str(defDate.year) + "-" + str(defDate.month) + "-" + str(defDate.day)
        date['start'] = str(defDate.year - 10) + "-" + str(defDate.month) + "-" + str(defDate.day)
    else:
        date['start'] = input("Enter analytics start date (YYYY-MM-DD): ") #user input of start date
        date['end'] = input("Enter analytics end date (YYYY-MM-DD): ") #user input of end date
    return date #returns date to original function

def stocktodf(rawSymbol, date): #function that takes in two inputs: symbol and formatted date, formats into data frame

    symbol = Share(rawSymbol) #takes raw symbol and creates a yahoo-finance API object
    stockData = pd.read_json(json.dumps(symbol.get_historical(date['start'], date['end']))) #defines "stockData" read
    #from json format as start and end date from yahoo get_historical function
    stockData = stockData.iloc[::-1] #reverses stockData date
    stockData.loc[:, "RollMean100"] = pd.Series(pd.rolling_mean(stockData['Adj_Close'], 100), index=stockData.index)
    #retrieves everything in the array adjusted close list and runs 100-rolling average, parses

```

The Run console at the bottom shows the following output:

```

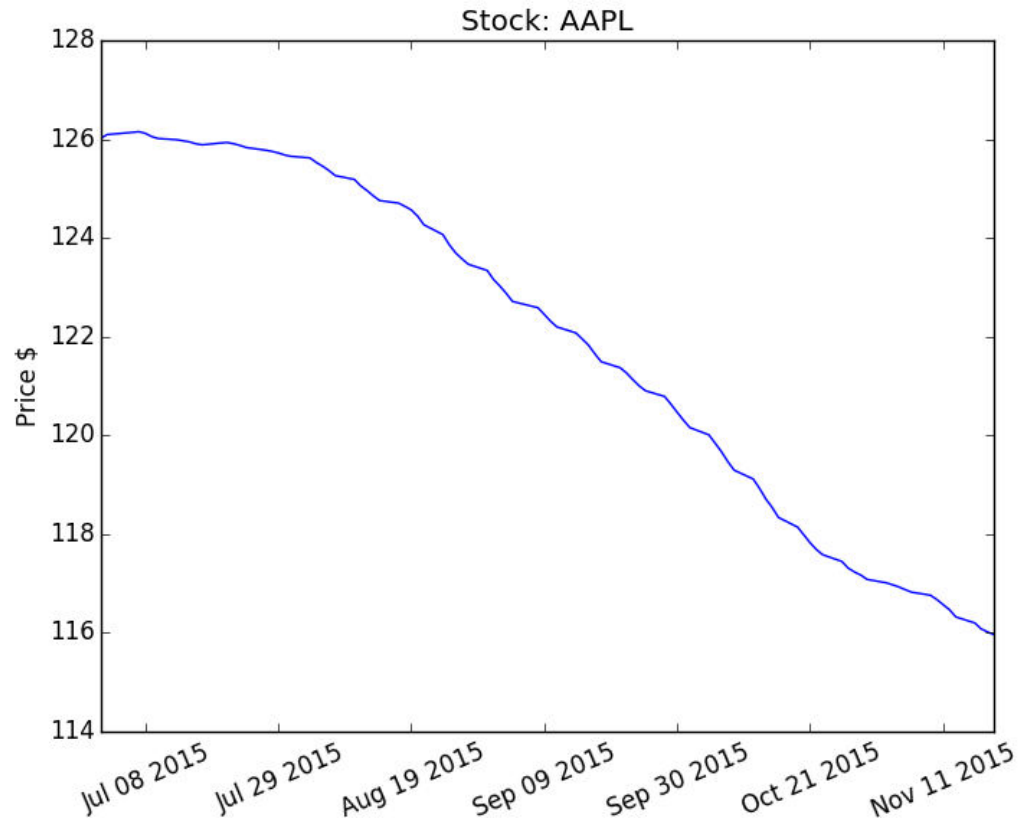
The comma separated value file for NKE has been saved
The plot for NKE has now been saved
NKE is a candidate for purchase
The histogram for WMT has now been saved
The comma separated value file for WMT has been saved
The plot for WMT has now been saved
WMT is not a candidate for purchase
The histogram for HOG has now been saved
The comma separated value file for HOG has been saved
The plot for HOG has now been saved
HOG is not a candidate for purchase
Purchase Stock GE the hundred day linear regression is 0.09 and three-hundred day is 0.00
Purchase Stock MCD the hundred day linear regression is 0.05 and three-hundred day is 0.01

Process finished with exit code 0

```

The status bar at the bottom indicates: 56:1 CRLF+ UTF-8+ Git: master+.

# Poly Hedge Fund – Graph

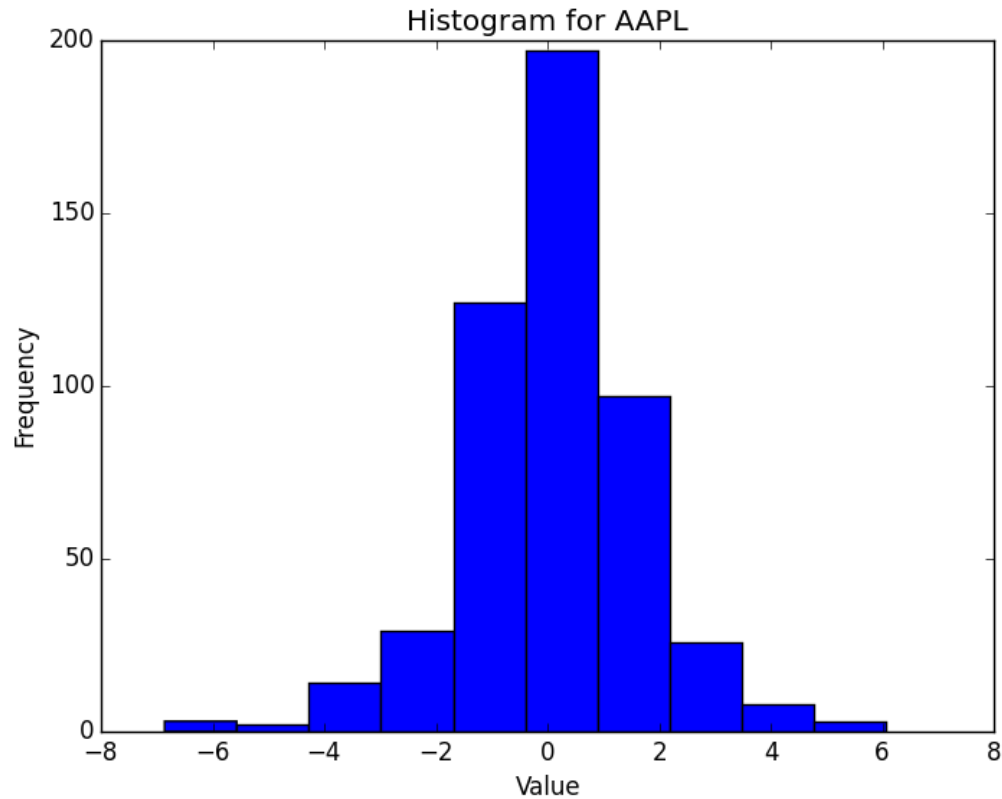


# Poly Hedge Fund – Text

Adj_Close	Close	Date	High	Low	Open	Symbol	Volume	RollMean: Ro
70.84792	515	11/20/2013	520.42	514.33	519.23	AAPL	48479200	
71.69259	521.14	11/21/2013	521.21	513.67	517.6	AAPL	65506700	
71.50825	519.8	11/22/2013	522.16	518.53	519.52	AAPL	55931400	
72.05027	523.74	11/25/2013	525.87	521	521.02	AAPL	57327900	
73.37919	533.4	11/26/2013	536.14	524	524.12	AAPL	1E+08	
75.10705	545.96	11/27/2013	546	533.4	536.31	AAPL	90862100	
76.49787	556.07	11/29/2013	558.33	547.81	549.48	AAPL	79531900	
75.83204	551.23	12/2/2013	564.33	550.82	558	AAPL	1.18E+08	
77.90795	566.32	12/3/2013	566.38	557.68	558.3	AAPL	1.13E+08	
77.72636	565	12/4/2013	569.19	560.82	565.5	AAPL	94452400	
78.12531	567.9	12/5/2013	575.14	566.41	572.65	AAPL	1.12E+08	
77.04127	560.02	12/6/2013	566.75	559.57	565.79	AAPL	86088100	
77.92308	566.43	12/9/2013	569.58	560.9	560.9	AAPL	80123400	
77.80202	565.55	12/10/2013	567.88	561.2	563.58	AAPL	69567400	
77.22561	561.36	12/11/2013	570.97	559.69	567	AAPL	89929700	
77.1122	560.54	12/12/2013	565.24	560.02	562.14	AAPL	65570500	



# Poly Hedge Fund -- Histogram



# Tools Used

Yahoo Finance API

```
graph TD; A[Yahoo Finance API] --> B[Python (Pandas)]; B --> C[R (Forecast)];
```

Python (Pandas)

R (Forecast)

# Lab Speakers-Outreach

- **Dr. Tim Regan – Lakeland Regional Health**
- **Todd Baylis – Qgiv CEO**
- **Sean McCarty – Lakeland Electric**
- **John Fico – Rockwell Industries**
- **Theresa Sutter – Curtiss-Wright**
- **Susan LeFrancois – Big Data and Health Informatics**
- **Jennifer Staab – Wiki presentation**
- **Matt Gruenau – Chastain Skillman**
- **Gary Snyder – Senior Manager of Business Analysis  
Publix**

**Thank You**