Trading Strategy Based on Unsupervised Learning

TCDS 13

Yuval Weinberger

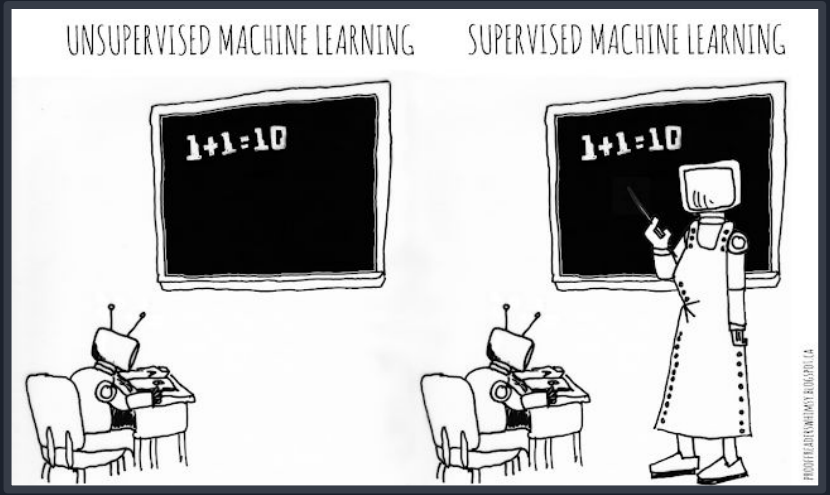
In The Beginning...

- Trying to use classifiers to classify “Bull” days and “Bear” days.
- Used SVM, RF, Logistic Regression, numerous different features and PCA, with disappointing results...

	precision	recall	f1-score	support
-1	0.68	0.38	0.49	981
1	0.28	0.58	0.38	419
accuracy			0.44	1400
macro avg	0.48	0.48	0.43	1400
weighted avg	0.56	0.44	0.46	1400

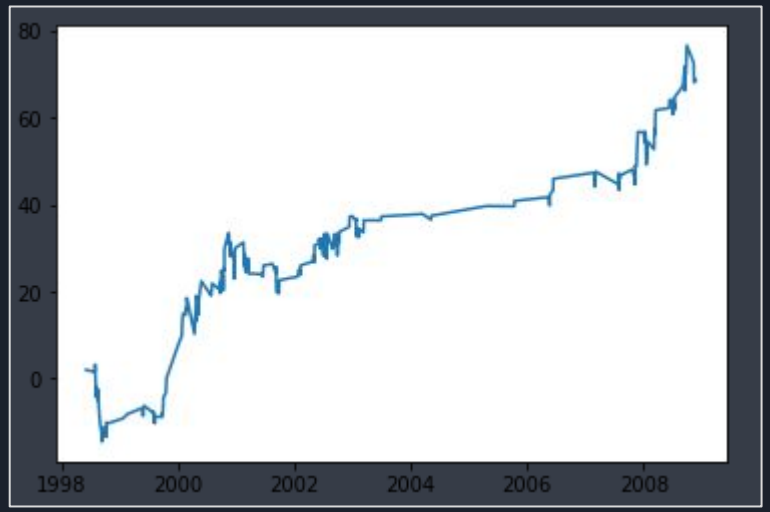
From Supervised to Unsupervised

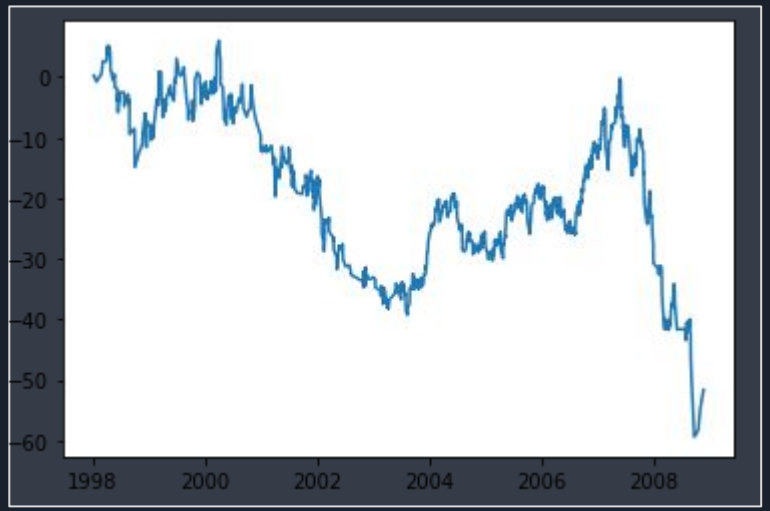
- So I wasn't able to "teach" the machine to identify "Bull" days and "Bear" day.
- But what if the machine could understand "Bull" days and "Bear" days patterns without a defined target?



The Goal

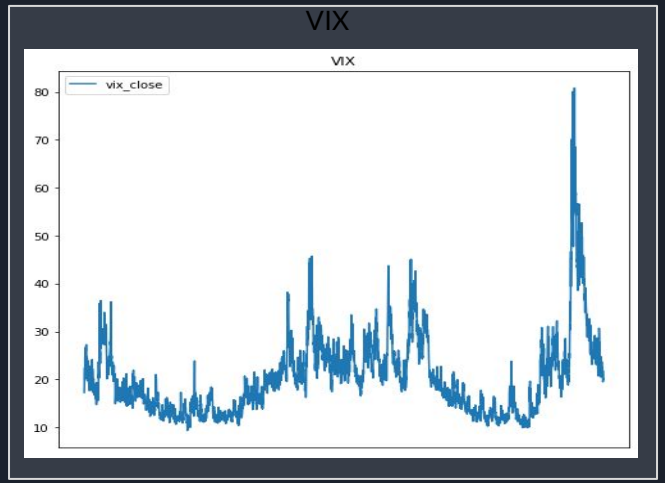
- The goal is to use K-Means to find clusters with stable gain curves.
- A cluster that shows higher return rate – will be used on testing data.





The Data

- The SPY ETF daily data from 1993 to 2020.
- The VIX close rate of daily data from 1993 to 2020.



Feature Extraction

- Candle size difference.
- VIX close rate.

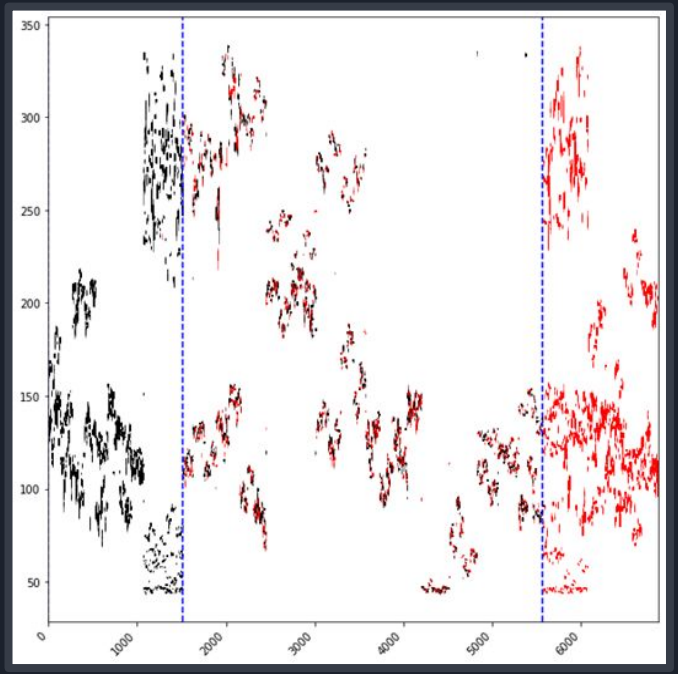
```
df["hl_range"] = (df["High"] - df["Low"])
df["oc_range"] = (df["Open"] - df["Close"])
df["oh_range"] = (df["Open"] - df["High"])
df["ol_range"] = (df["Open"] - df["Low"])
df["cl_range"] = (df["Close"] - df["Low"])
df["ch_range"] = (df["Close"] - df["High"])

df["vix_close"] = df["vix_close"] / talib.SMA(df.vix_close, 20)

```

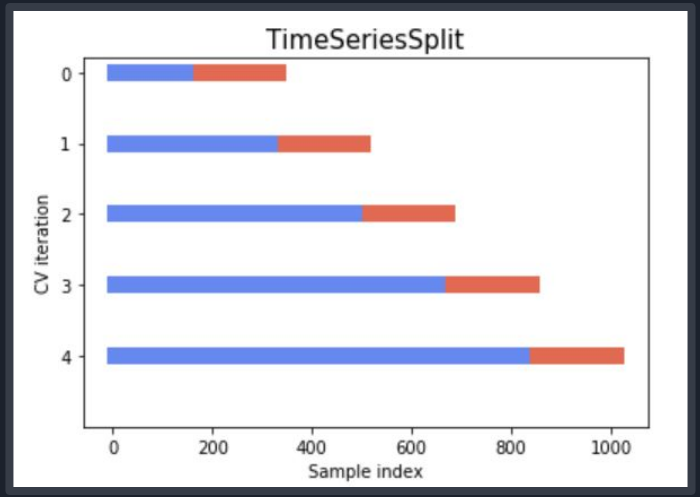
Example of Candle Classification

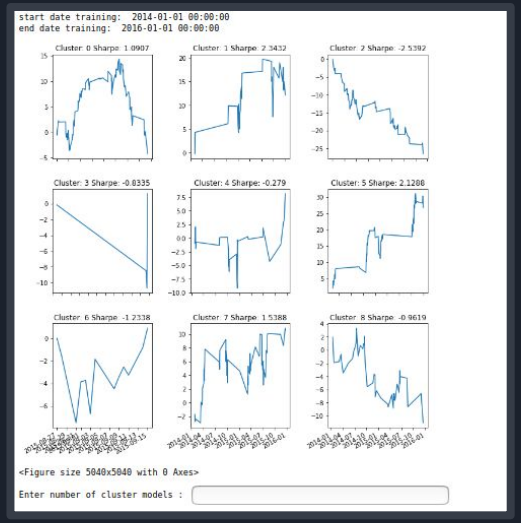
- “Bull” days and “Bear” days.
- Classified using K-Means.



Model – Train Test Process

- Stock and ETF quotes are time-series data, therefore we can't randomly shuffle the data.
- As part of the walk-forward process the user will choose the clusters to be used as a trading strategy on the testing phase.
- "Unseen" testing set: 2015-2020





Evaluation Factors

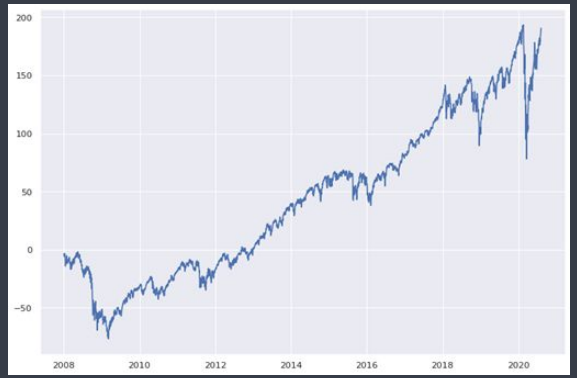
- To evaluate the model, the point gain will be used as a measure of profit. Every time a day is classified in a chosen cluster, the gain/loss in points will be added to the general gain curve.
- If a day is classified in “Buy Cluster” we buy at close and sell at the next day ‘s close.

```
self.gain_df["gain"] = self.gain_df["Close"].shift(-1) - self.gain_df["Close"]
```

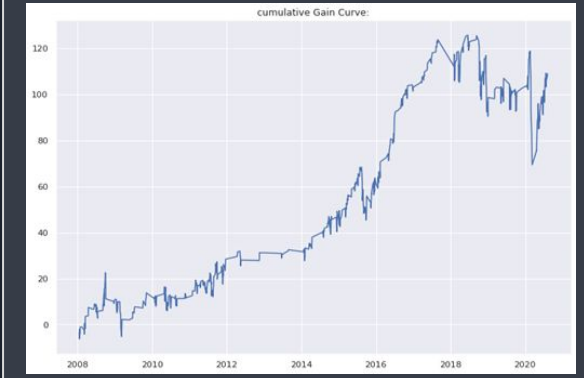
Results

- In the example, until 2018 ,this strategy outperformed the market in almost every aspect: better risk/reward, smaller draw-downs, more free liquidity and higher returns.
- Since 2018, the markets became more volatile, and this strategy stopped performing well in the new market behavior.

SPY Cumulative Gain

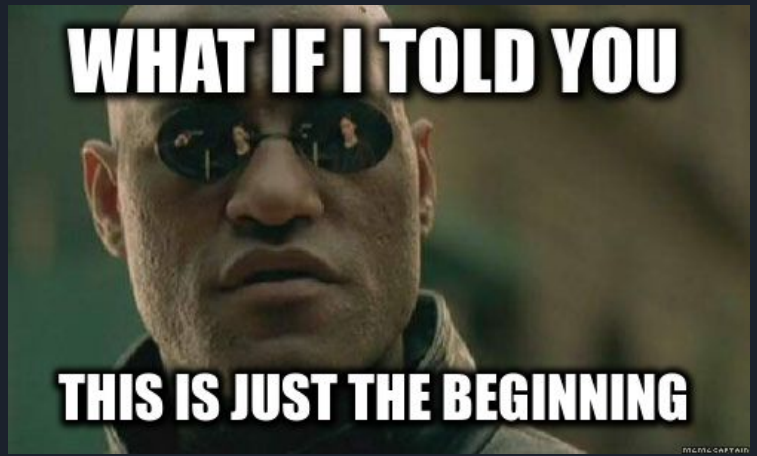


Strategy Cumulative Gain



Future Work

- Short positions can be added when there is a strong “Bear” cluster.
- Stop-loss and take-profit points can be added , to lock profits, and cut losses fast.
- Input a LSTM neural network with the clusters, and more features and let the network classify the up vs down days.



Conclusion

- Using the simplest features and position entering conditions, has shown to be a strategy that outperformed the market on set periods, yet, still unstable on other periods.
- This project can serve as a basis for future research and development on trading strategies that can combine unsupervised machine learning and technical analysis.

Bibliography

- A framework for trading system development based on k-means clustering
- K Means Clustering and Creating a Simple Trading Rule for Smoother Returns
- Stock Picks using K-Means Clustering | by Timothy Ong | uptick-blog
- K-Means Clustering of Daily OHLC Bar Data
- k-means clustering
- VIX