RiskEx + UC Berkeley DataX

Spring 2018

Background and Motivation

Basically, we are trying to solve this problem: **How do traders know which news articles/publications/news beats they should listen to and base their trading decisions on?**

1. What asset is the trader working with?
   - Get a list of past news articles on the topic from every available publication
   - Establish a listener to keep track of newly published articles

2. Which of these news articles are “relevant” to the price of the asset being traded (i.e. which of these news items directly affect prices and lead to trading activity?)
   - Regress article timing and “attributes” to prices to establish a relationship
   - Generate a “Relevance Score” based on a number scale or (1-5) or Low, Medium and High

3. How reputable are the sources from which these articles are published?
   - "Reputation Score” can be initially assigned i.e. Reuters, AP, PRN etc. can be whitelisted with a high score
   - Eventually, “Reputation Score” can be crowdsourced/algorithmically determined and trustworthiness can be established using several variables.
     - How long has the publisher been around?
     - Historically, how accurate and relevant to trading activity have their articles been?

Final Article Score will be a function of Relevance Score + Reputation Score of the publisher. This score will be assigned to future articles and displayed to the trader based on the total score.

Traders can then get **updates in real-time** with our bot predicting how impactful the news will be.

In this initial phase of the bot, we focused on **Bitcoin** as our primary asset.

Project Scope and Progress

Work Performed Thus Far:

1. **Data Scraping** - notebooks have been created that can scrape metadata and content for a given URL.
    1. Goal is to re-purpose code to scrape desired websites given some flag. (either time, user requested, or triggered by modeling demands)/
2. **News API** - notebooks have been created for accessing HTML response objects, and for extracting metadata and article content.

3. **Data set** - 30,000+ time stamped articles have been extracted and are in the process of being aggregated with granular pricing data.

Next Steps:

I. **Words sub-project**

II. **Numbers sub-project (goal is to make it flexible enough that it can be ran on an ongoing basis).**

i. Function 1 (takes list of *flags*):

Compares rolling weigted average (rwa) to the flag parameter_i.

If weighted-price_i > or < than flag

Compute and track rolling flag

Record indeces while true

Continue until < or > than rolling flag

ouput rows as dict

ii. Function 2 (takes window range, list of flags)

Compares rolling weigted average (rwa) to the given flag.

For window range (wr) decresing by some metric

If weighted-price_i > or < than flag_i

Record indeces from (-)wr to (+)wr

Expand (+)wr until flag turned off, contineu recording for (+)wr_i

Repaet n-times according to some metric that decreases window of time

Record and return dict of data(hr)

iii. Function 3 (takes dict from F1, and F2)

Iterarates over the data set, and extracts appropriated for each ordered pair (flag_i, data(hr)_j) for i in data(flag) and j in data(hr))

Call Function 4 (pass parameters after appropriate clean up)

iv. Function 4 (takes dict from Function 3)

Computes time series analysis for the given paramenters

Computes validation metrics, for each set of parameters passed

Compare metrics and find best choice for *flag* and *(+,-)hr*

Return appropriate information

General and Ongoing Steps

1. **Finalize data set** - perform cleanup and preprocessing for news and price data
2. **Exploratory Data Analysis (EDA)** - extract features, and perform basic EDA and visualizations.
3. **Feature Extraction** - use knowledge from EDA to curate set of features. Perform modeling with data (regression, clustering, networks, etc.) to gather insight into relationships and potential features.
4. **Time Series, and Classification** - model behavior and create classification model (relevant/non-relevant news) to predict price-impact of news-event.

Future Projects (ideas/recommendations):

1. **Better granularity of data** - create series of crawlers that extract the desired granular data.
    1. Examples include: pricing data, google trend data, etc.
    2. Broad spectrum crawlers should be scheduled. Event driven crawlers will have to be developed around predictive modeling and/or event detection.
2. **Create Predictive Pricing Model.** Use pricing predictions to trigger searches for relevant articles, and evaluate both pricing model and recommender system. Pricing model can also be used for arbitrage purposes, and optimization of LOB processes.
3. **Export processes to cloud.** Data set size, and dynamic processing will require greater capacity to store data, as well as faster and/or distributed computing
4. **Better Detection -** use knowledge gained from part 4 above, to create a ranking of feature importance/impact, that can then be used to prioritize and optimize crawling efforts in the future.
5. **Better Prediction** - refine and improve classification and price-prediction models.

Spring 2018 team:

Raghav Mathur: Project Manager

Manana Hakobyan: Co-technical Lead

Elias Castro Hernandez: Co-technical Lead

Ran Jiang: Market Data Lead