Numerai Signals

@parmarsuraj99

Slides: https://tinyurl.com/signals-toronto

Code: Colab notebook

What's a Signal?

- Numerical data about stocks
- Relative ranking of stocks

ticker	signal
AAPL US	0.5488135039273250
TSLA US	0.7151893663724200
EMG LN	0.6027633760716440
2638 HK	0.5448831829968970
MOH GA	0.4236547993389050

https://docs.numer.ai/numerai-signals/signals-overview

Why “Signals”? - *Monopolize data*

- Phase 2 of master plan
  - Monopolize intelligence
  - **Monopolize data**
  - Monopolize money
  - Decentralize the monopoly
- Crowdsourcing features
  - Features in classic are fixed
  - Signals open up for new/unique features
- Incentivize original signals

https://medium.com/numerai/numerais-master-plan-1a00f133dba9

Similarities with Classic

- Scoring function
  - Spearman’s Correlation
- Neutralization
  - Can be used here as well
- Targets
  - Neutralized 20 day returns
  - Relative ranking of stocks
  - Quantiled form
- Data splits
  - Training
  - Validation
  - Live
- Metrics
  - CORR
  - MMC
  - TC

How it differs

- Data
  a. Download Universe
  b. Bring your own data
  c. Generate your features
- Submission
  a. Minimum 10 stocks for live data

https://docs.numer.ai/numerai-signals/signals-overview

Universe

- A list of stock tickers
- Numerai wants predictions for
- Updated in each round

```
1  eligible_tickers
```

```
0        SVW AU
1        GEM AU
2        AZJ AU
3        NXT AU
4        TWE AU
         ...
5336    HYFM US
5337      NG US
5338    IMAX US
5339    LULU US
5340     TNK US
Name: bloomberg_ticker, Length: 5341,
```

What are the targets?

- Relative ranking of stocks in a universe
- Long:
    - Buy
    - Numbers go up
    - Closer to 1
- Short:
    - Sell
    - Numbers go down
    - Closer to 0

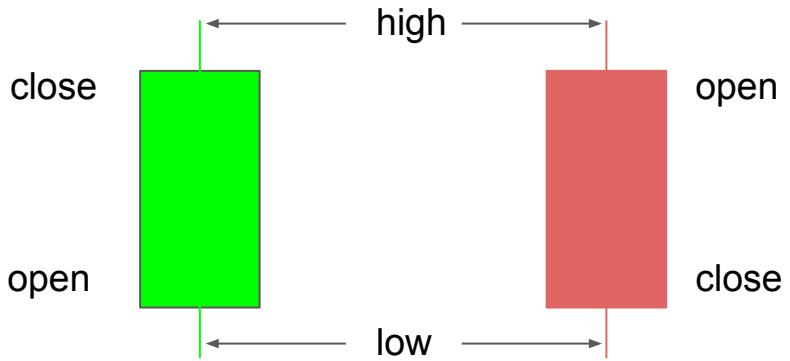
```
1  targets[targets["date"]=="2021-12-31"]
```

	bloomberg_ticker	target_20d	date
4509625	000060 KS	1.00	2021-12-31
4509626	000080 KS	0.50	2021-12-31
4509627	000100 KS	0.50	2021-12-31
4509628	000120 KS	0.50	2021-12-31
4509629	000210 KS	0.50	2021-12-31
...	...	...	...
4514995	ZUO US	0.75	2021-12-31
4514996	ZURN SW	0.75	2021-12-31
4514997	ZWS US	0.50	2021-12-31
4514998	ZYXI US	0.50	2021-12-31
4514999	ZZZ CN	0.50	2021-12-31

5375 rows × 3 columns

Data

- OHLCV
- Textual data
- Fundamental data
- Options/Futures data
- Any quantifiable data



“$$$ to the moon soon”

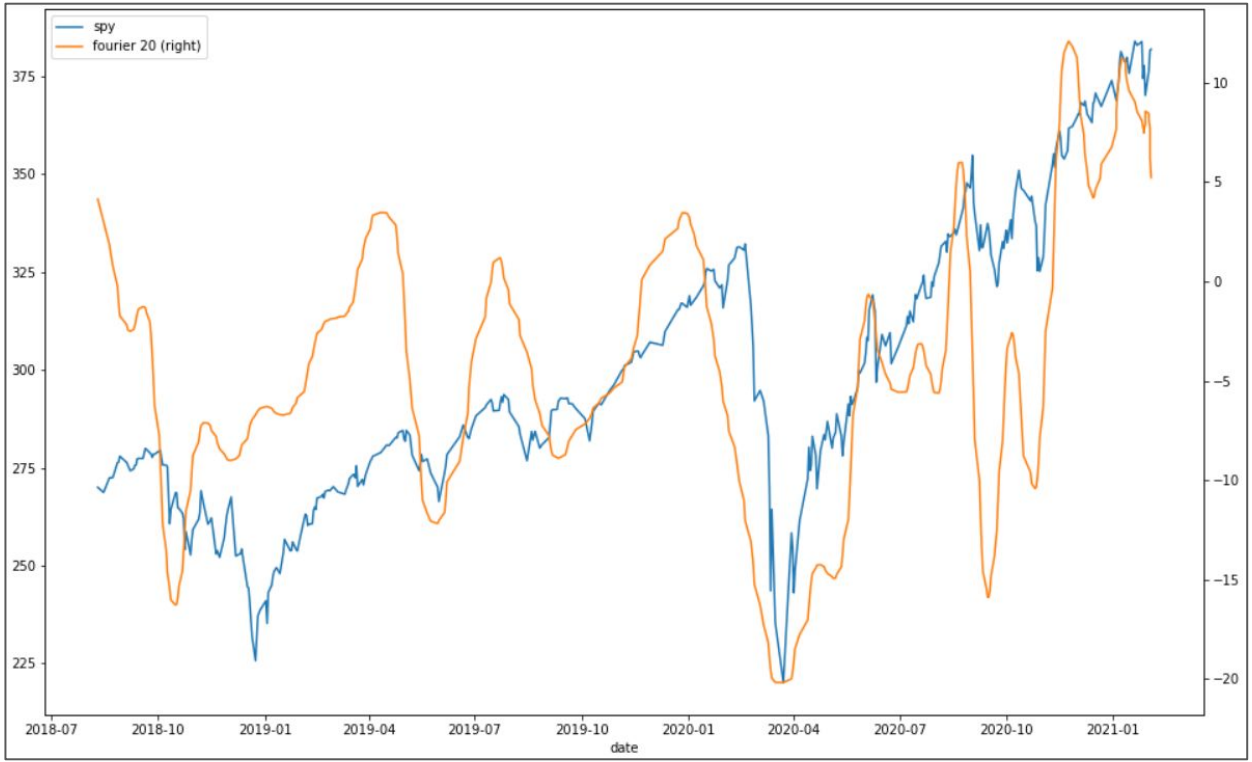
“$$$ reported worst quarter in 3 years”

“stonks only go up. not a financial advice”

Data - Textual Data

1. Sources
    a. News
    b. Tweets
    c. r/WallStreetBets
    d. …
2. Words to Numbers
    a. Use a pretrained sentiment model
    b. Or train your own
3. Numbers to ranks
    a. Sentiment scores for universe
    b. Rank based on the scores
4. Submit

Data - Textual Data - r/WallStreetBets



- Building on top of r/WSB users' due diligence

From r/WSB to Numerai Signals - Suraj Parmar

Getting Started with Numerai Signals - Carlo Lepelaars

Data - OHLCV

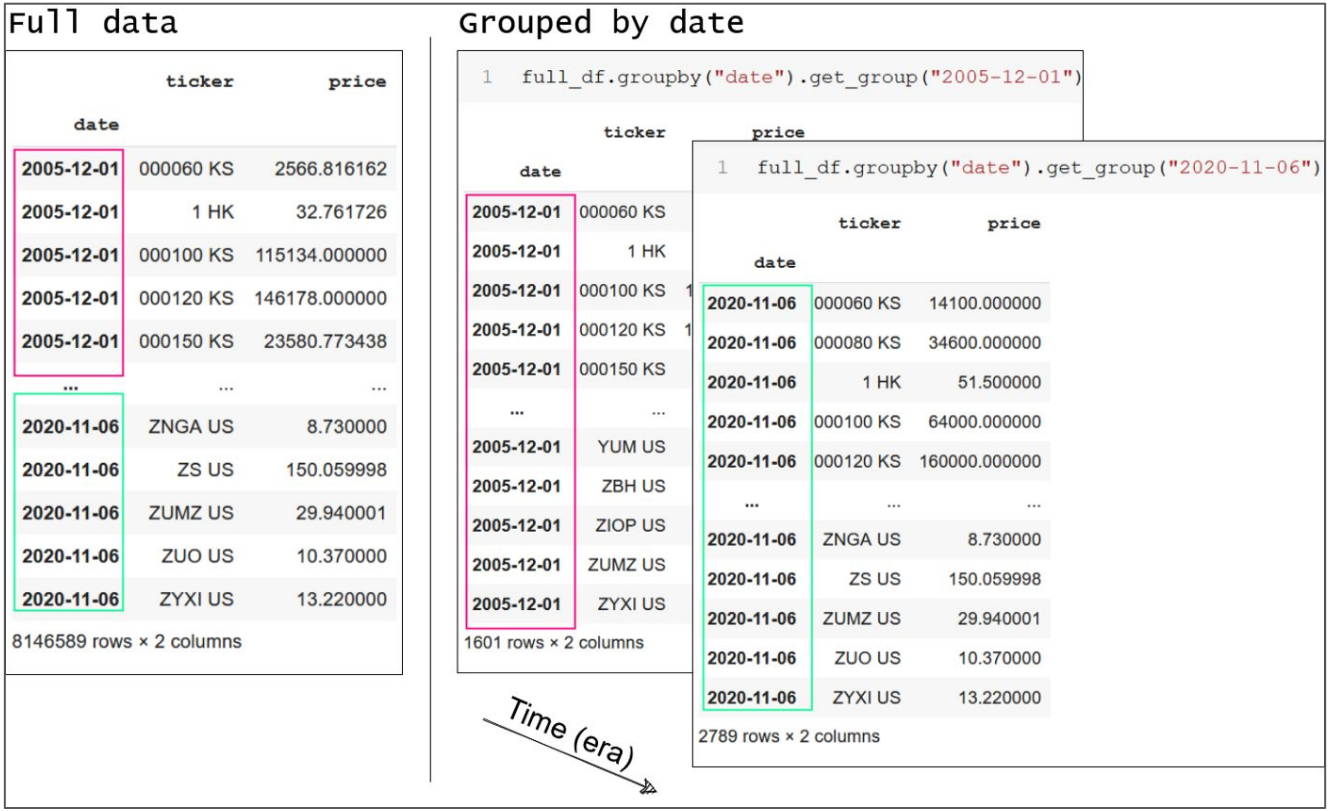
- OHLCV
  - Open
  - High
  - Low
  - Close
  - Volume

Data - OHLCV

- OHLCV
  - Open
  - High
  - Low
  - Close
  - Volume

- **Idea**
  - Get quality data
  - Generate features
  - Transform to classic’s format
  - Use model Scripts from classic

Data - Structure



date	ticker	price
2005-12-01	000060 KS	2566.816162
2005-12-01	1 HK	32.761726
2005-12-01	000100 KS	115134.000000
2005-12-01	000120 KS	146178.000000
2005-12-01	000150 KS	23580.773438
...	...	...
2020-11-06	ZNGA US	8.730000
2020-11-06	ZS US	150.059998
2020-11-06	ZUMZ US	29.940001
2020-11-06	ZUO US	10.370000
2020-11-06	ZYXI US	13.220000

8146589 rows × 2 columns

date	ticker	price
2005-12-01	000060 KS	
2005-12-01	1 HK	
2005-12-01	000100 KS	
2005-12-01	000120 KS	
2005-12-01	000150 KS	
...	...	
2005-12-01	YUM US	
2005-12-01	ZBH US	
2005-12-01	ZIOP US	
2005-12-01	ZUMZ US	
2005-12-01	ZYXI US	

1601 rows × 2 columns

date	ticker	price
2020-11-06	000060 KS	14100.000000
2020-11-06	000080 KS	34600.000000
2020-11-06	1 HK	51.500000
2020-11-06	000100 KS	64000.000000
2020-11-06	000120 KS	160000.000000
...	...	...
2020-11-06	ZNGA US	8.730000
2020-11-06	ZS US	150.059998
2020-11-06	ZUMZ US	29.940001
2020-11-06	ZUO US	10.370000
2020-11-06	ZYXI US	13.220000

2789 rows × 2 columns

Data - OHLCV - Feature generation

- Technical Features
    - Simple Moving Average
    - Exponential Moving Average
    - Relative Strength Index
    - Money Flow Index
    - MACD
    - Crossovers (SMA_7/SMA_14)
    - Volatility
    - …

https://mrjbq7.github.io/ta-lib/

Data - OHLCV - Features

TradingView

Data - OHLCV - Features

```
1  full_df
```

date	mfi_14	mfi_21	rsi_14	rsi_21	sma_7	sma_21	sma_50	ema_7	ema_21	ema_50	bloomberg_ticker	adj_close
2003-02-07	40.162048	49.134438	36.559124	34.999985	462.837189	484.687042	526.277771	464.900269	480.193756	502.519775	000060 KS	462.837128
2003-02-10	46.967247	49.286930	39.534889	34.426228	465.687164	481.077057	523.405029	463.387024	478.238007	500.551208	000060 KS	458.847229
2003-02-11	50.393333	50.746490	45.348850	34.426228	469.962128	477.467072	520.492310	464.745789	477.375946	499.129364	000060 KS	468.822144
2003-02-12	52.156631	46.301655	51.648354	37.499981	471.387115	474.427094	518.656921	469.754852	478.053558	498.489990	000060 KS	484.781982
2003-02-13	53.012657	52.408001	63.529423	40.740734	473.097107	472.052124	517.180664	477.002869	479.945801	498.501373	000060 KS	498.746948
...	...	...	...	...	...	...	...	...	...	...	...	...
2022-02-17	52.222534	51.093681	21.897392	22.762276	32.813084	34.687000	36.122387	32.965977	34.397774	35.458832	ZZZ CN	32.110001
2022-02-18	48.269386	49.752239	22.463902	25.643637	32.560307	34.454105	35.999519	32.689480	34.167068	35.317703	ZZZ CN	31.860001
2022-02-22	46.931225	48.941280	14.553805	23.638109	32.236965	34.180645	35.859219	32.192112	33.851879	35.136616	ZZZ CN	30.700001
2022-02-23	58.065453	52.377228	13.806841	22.961990	31.955853	33.891914	35.711708	31.709084	33.525345	34.945377	ZZZ CN	30.260000
2022-02-24	61.257698	56.830391	14.661114	23.310106	31.563753	33.611156	35.568344	31.281813	33.204861	34.751438	ZZZ CN	30.000000

18826443 rows × 12 columns

Data - OHLCV - Feature engineering

- Normalization
  - Min-max scaling
  - Z-score?
  - Peaking into future
  - **Percentile transformation**

Data - OHLCV - Feature engineering

- Normalization
  - Min-max scaling
  - Z-score?
  - Peaking into future
  - **Percentile transformation**

- Transform to Classic's format
  - **Quantile-based discretization**
  - **Brings stationarity**

Data - OHLCV - Feature engineering

```
full_df[list(sma_features + ema_features+ rsi_features+ mfi_features) + ["bloomberg_ticker"]]
```

date	sma_7	sma_21	sma_50	ema_7	ema_21	ema_50	rsi_14	rsi_21	mfi_14	mfi_21	bloomberg_ticker
2003-02-06	3.0	3.0	3.0	3.0	3.0	3.0	4.0	2.0	4.0	1.0	AGL AU
2003-02-06	1.0	1.0	1.0	1.0	1.0	1.0	3.0	2.0	4.0	4.0	AIA NZ
2003-02-06	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.0	2.0	2.0	APA AU
2003-02-06	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	2.0	1.0	API AU
2003-02-06	2.0	2.0	2.0	2.0	2.0	2.0	2.0	3.0	0.0	0.0	AWC AU
...	...	...	...	...	...	...	...	...	...	...	...
2022-02-25	2.0	2.0	2.0	2.0	2.0	2.0	3.0	2.0	4.0	4.0	9962 JP
2022-02-25	4.0	4.0	4.0	4.0	4.0	4.0	2.0	0.0	0.0	0.0	9983 JP
2022-02-25	3.0	3.0	3.0	3.0	3.0	3.0	2.0	2.0	3.0	2.0	9984 JP
2022-02-25	2.0	2.0	2.0	2.0	2.0	2.0	4.0	4.0	2.0	4.0	9987 JP
2022-02-25	2.0	2.0	2.0	2.0	2.0	2.0	4.0	4.0	3.0	1.0	9989 JP

18826443 rows × 11 columns

Data - OHLCV - Merging Targets

date	sma_7	sma_21	sma_50	ema_7	ema_21	ema_50	mfi_14	mfi_21	bloomberg_ticker	data_type	target_20d
2003-03-07	4.0	4.0	4.0	4.0	4.0	4.0	0.0	2.0	000270 KS	train	0.50
2003-03-07	4.0	4.0	4.0	4.0	4.0	4.0	2.0	2.0	000810 KS	train	0.50
2003-03-07	4.0	4.0	4.0	4.0	4.0	4.0	2.0	4.0	002790 KS	train	1.00
2003-03-07	4.0	4.0	4.0	4.0	4.0	4.0	0.0	0.0	003490 KS	train	0.50
2003-03-07	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0	004170 KS	train	0.25
...	...	...	...	...	...	...	...	...	...	...	...
2022-01-07	0.0	0.0	1.0	0.0	0.0	0.0	4.0	4.0	ZUO US	validation	0.75
2022-01-07	3.0	3.0	3.0	3.0	3.0	3.0	3.0	4.0	ZURN SW	validation	0.50
2022-01-07	1.0	1.0	1.0	1.0	1.0	1.0	0.0	3.0	ZWS US	validation	0.50
2022-01-07	0.0	0.0	0.0	0.0	0.0	0.0	1.0	3.0	ZYXI US	validation	0.25
2022-01-07	1.0	1.0	1.0	1.0	1.0	1.0	4.0	0.0	ZZZ CN	validation	0.25

2656062 rows × 11 columns

Training

- Data looks similar to classic
- Classic’s model can be used here
- Target neutralization
- Prediction Neutralization

```
from xgboost import XGBRegressor

feature_names = list(new_indicators)

model = XGBRegressor(
    tree_method="gpu_hist"
)

gc.collect()
model.fit(train_data[feature_names],
          train_data[TARGET_NAME],
          eval_set=[(test_data[feature_names],
          test_data[TARGET_NAME])],
          verbose = 10)
```

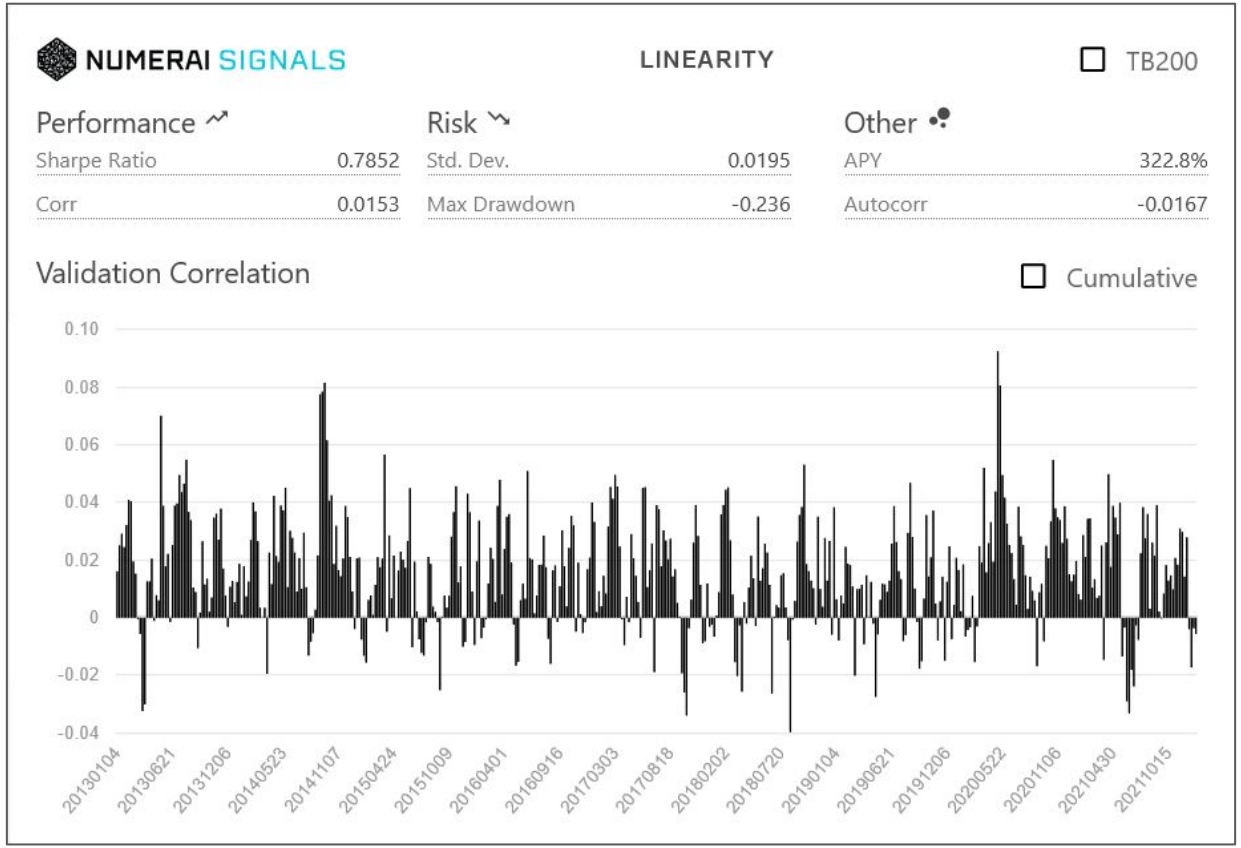
Submission

date	bloomberg_ticker	friday_date	data_type	signal
2013-01-04	000060 KS	20130104	validation	0.746343
2013-01-04	000080 KS	20130104	validation	0.076597
2013-01-04	000100 KS	20130104	validation	0.260585
2013-01-04	000120 KS	20130104	validation	0.433025
2013-01-04	000210 KS	20130104	validation	0.543880
...	...	...	...	...
2022-02-24	ZURN SW	20220225	live	0.526634
2022-02-24	ZWS US	20220225	live	0.679228
2022-02-24	ZY US	20220225	live	0.979530
2022-02-24	ZYXI US	20220225	live	0.965806
2022-02-24	ZZZ CN	20220225	live	0.576181

1620045 rows × 4 columns

Walkthrough

Diagnostics - Backtest



Notes

- Use quality data
- Survivorship bias
- Look-ahead bias
- Currency difference
- Possible to submit raw features
    - No machine learning needed
    - Rank the values for live universe
- 50 model slots
- Backtest
- Live test your predictions w/o hassle of trading

What’s next?

- Look for other indicators
- Try different modeling techniques
- Create and evaluate your own targets
- Experiment and submit with more models
- Participate on RocketChat or Forum
- Check out numerbay.ai

What’s next?

- Open Signals
- Signals example script
- Jason’s notebook
- [NumeraiSignals] Starter for beginners
- Getting Started with Numerai Signals
- From r/WSB to Numerai Signals
- Let’s talk about Signals

Thank you