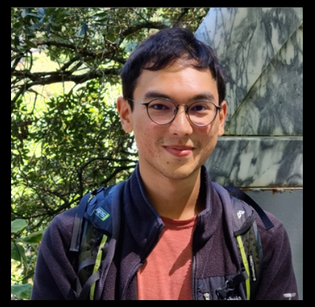
CAPITAL COMPASS

Developing quantitative tools to assist trading decisions in US equity market.








Currently live at http://3.104.120.14/

https://www.canva.com/design/DAG2xPOZbuw/ZNmGXpJEDM78LFxVgC8hAg/edit

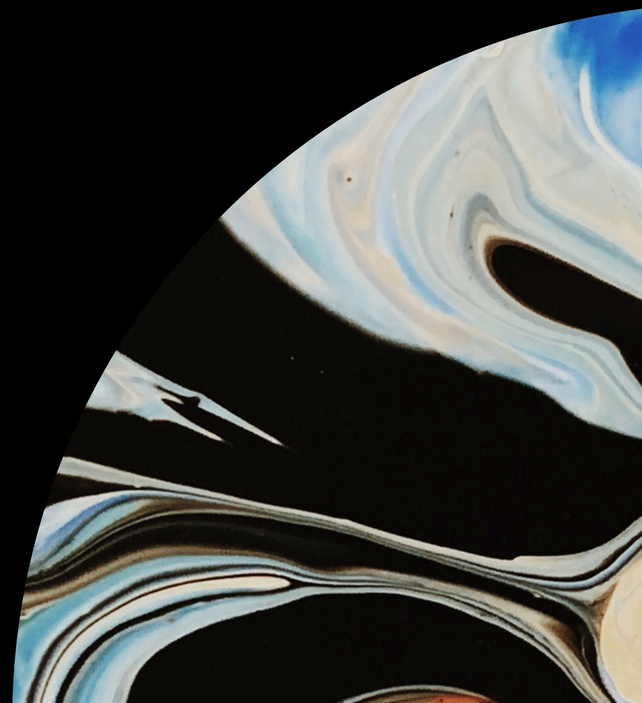
Team 11 - Bobbers

Charles, Hoang, Shawna, Phawat, Henri, Trey

(left to right, top to bottom)

Presentation Outline

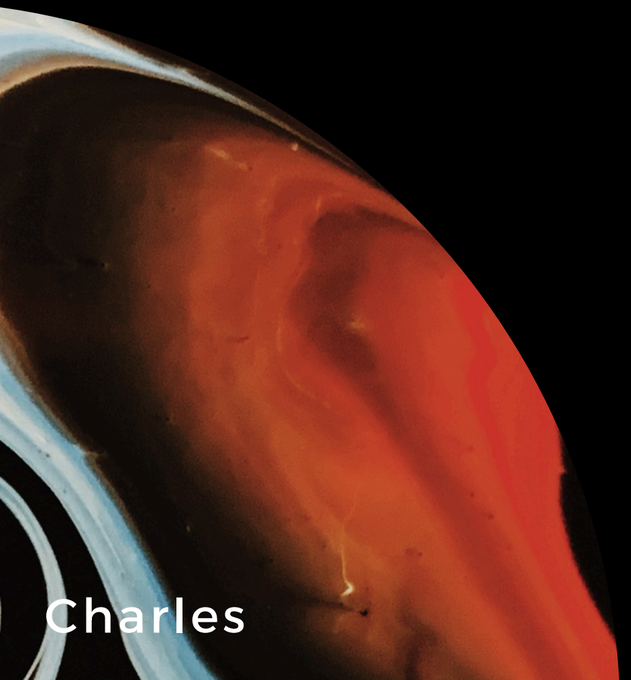
1. Introduction & Motivation
2. Demo
3. Solution Overview
4. Implementation Details
5. Challenges, Lessons Learned & Future Work
6. Project Goal Evaluation
7. Conclusion & Thanks



Charles

INTRODUCTION

Even though fund managers focus on investing in the long term, it is still important to optimise the short-term timing of their buy/sell decisions.

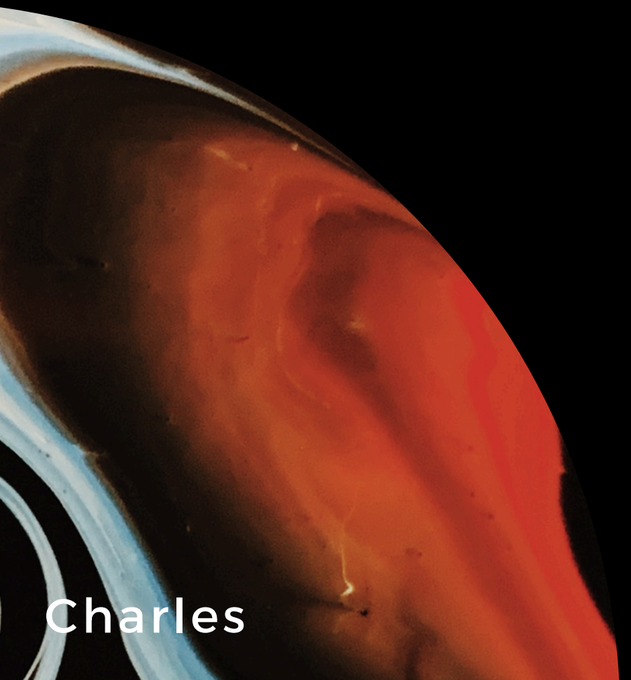


Charles

INTRODUCTION

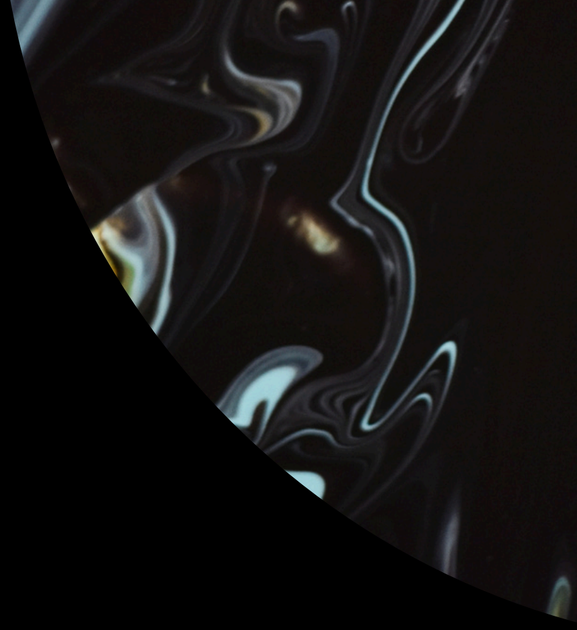
Even though fund managers focus on investing in the long term, it is still important to optimise the short-term timing of their buy/sell decisions.

Our work analyses the short-term trending/mean reverting behaviour of stocks, to help guide investors in fine-tuning the timing of their buy/sell decisions instead of doing so arbitrarily and risking value loss.



Charles

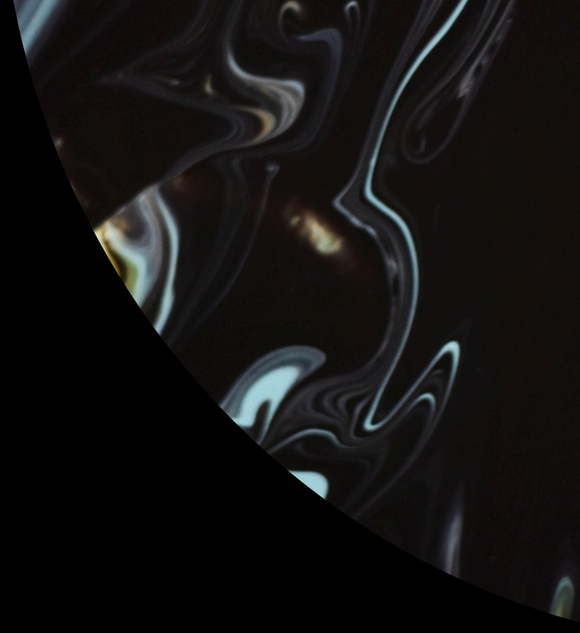
Novelty of our product



Pooled market-wide learning

Charles

Novelty of our product

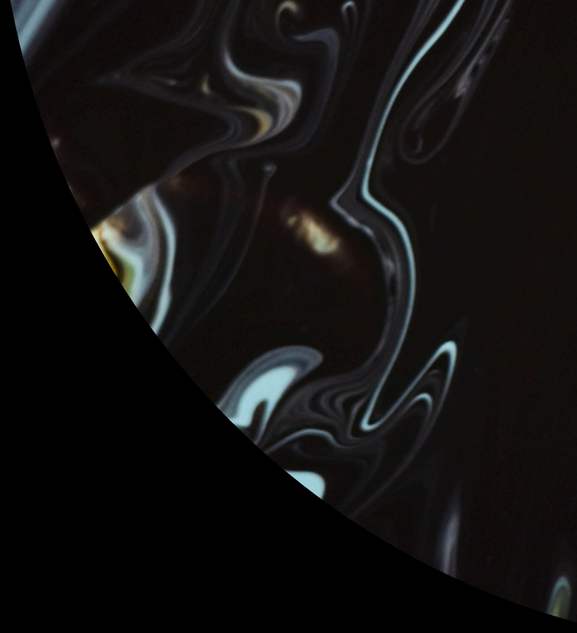


Pooled market-wide learning

Convenient and streamlined web-app

Charles

Novelty of our product



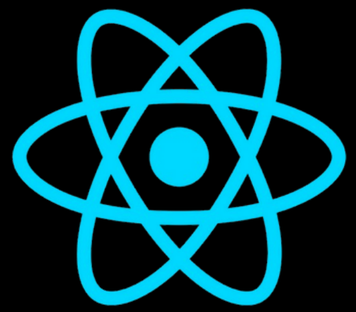
Pooled market-wide learning

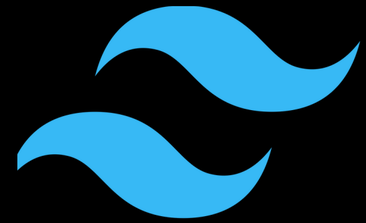
Convenient and streamlined web-app

Robust & flexible to incomplete data etc

LIVE DEMO

FRONTEND TECHNOLOGIES



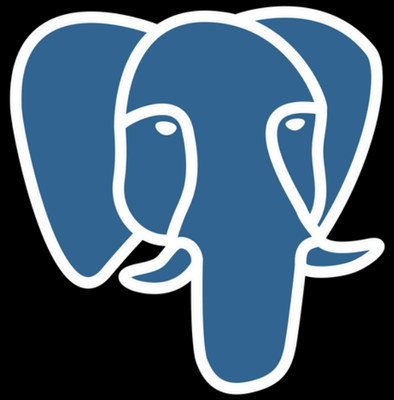


React Next.js Tailwind CSS

Phawat

BACKEND TECHNOLOGIES

Django REST

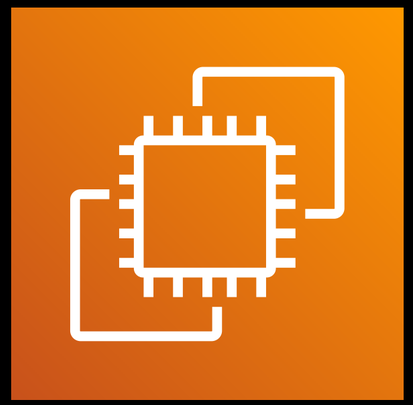


PostgreSQL Database Engine

Python 3.12+

Trey

DEPLOYMENT TECHNOLOGIES



4 CPU Cores

16GB RAM

Ubuntu Linux

USD$ 0.21 p/h

```
ubuntu@ip-172-31-22-224:~$ cat ascii.txt
 _____ __  __ _   ___  __
|_   _|  \/  | | | \ \/ /
  | | | |\/| | | | |\  /
  | | | |  | | |_| |/  \
  |_| |_|  |_|\___//_/\_\
ubuntu@ip-172-31-22-224:~$ echo 'we use tmux to handle backend \
> and frontend processes.'
we use tmux to handle backend \
and frontend processes.
ubuntu@ip-172-31-22-224:~$ apt list tmux
Listing... Done
tmux/noble-updates,now 3.4-1ubuntu0.1 amd64 [installed]
N: There is 1 additional version. Please use the '-a' switch to see it
ubuntu@ip-172-31-22-224:~$
```

AWS EC2

t3.xlarge VM

tmux utility

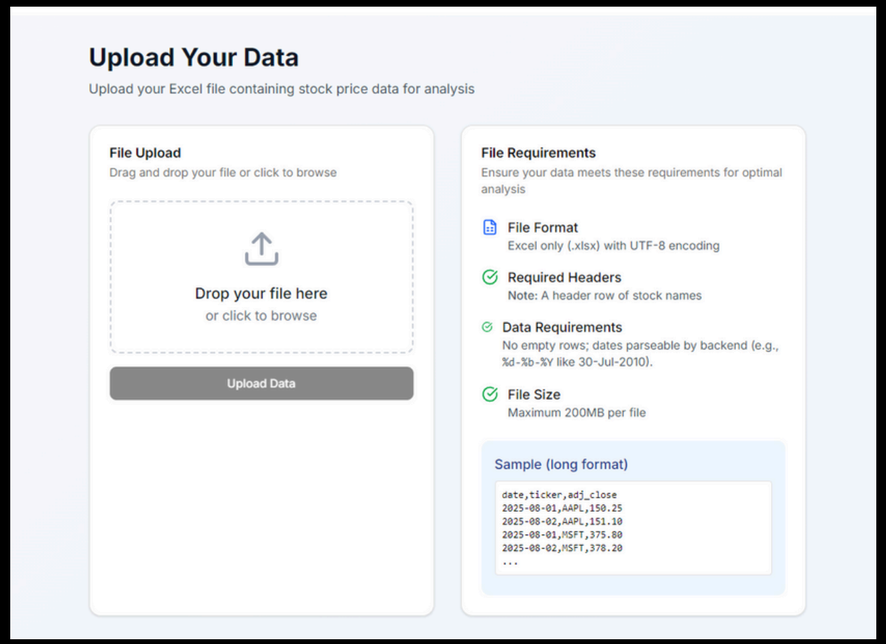
Henri

Implementation Details: Uploading

Take a pre-formatted Excel spreadsheet

Trey

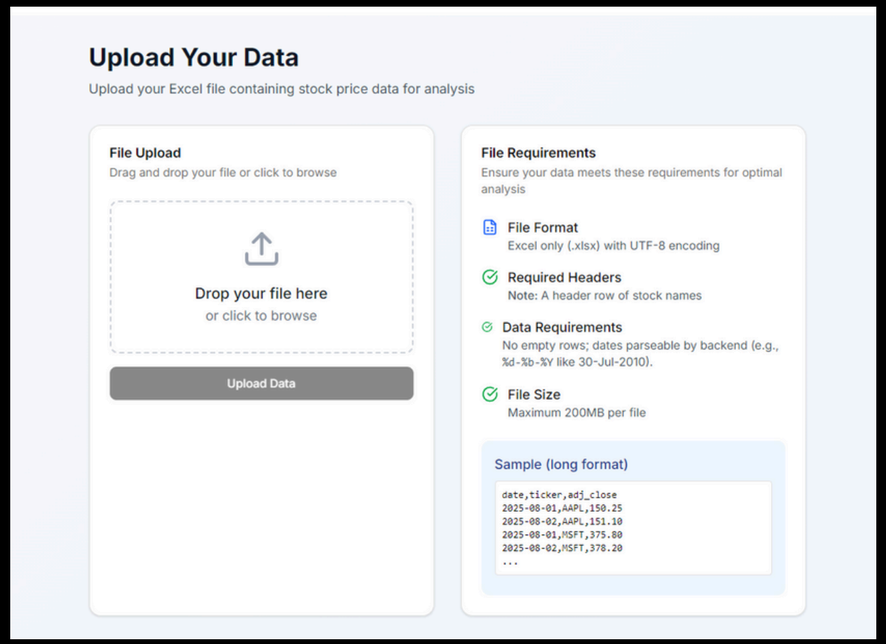
Implementation Details: Uploading

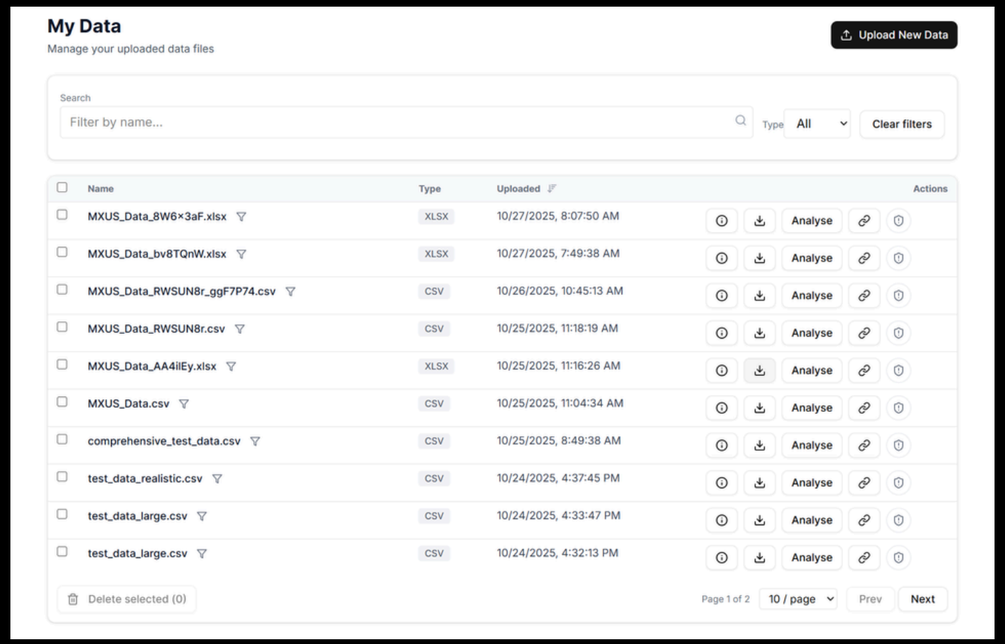


Take a pre-formatted Excel spreadsheet → Upload to the Capital Compass Webapp

Trey

Implementation Details: Uploading

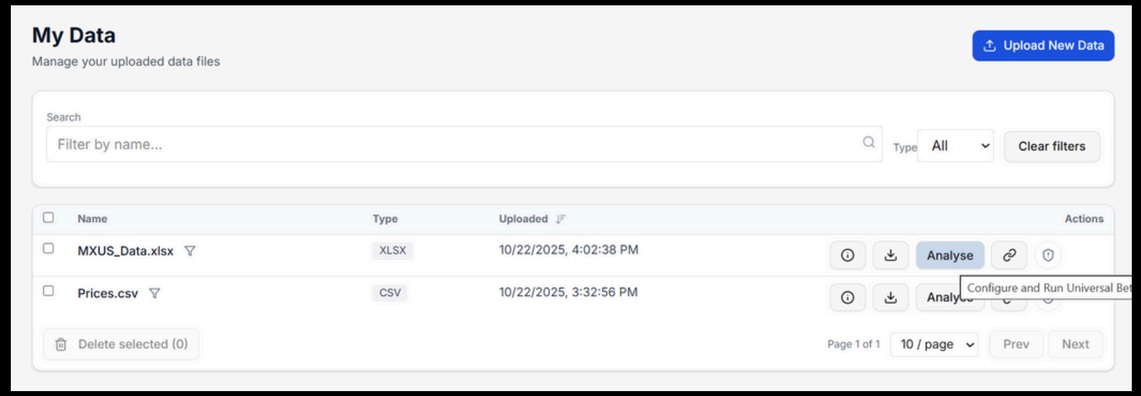




Take a pre-formatted Excel spreadsheet → Upload to the Capital Compass Webapp → View/Download your uploaded file on the ‘My Data’ page

Trey

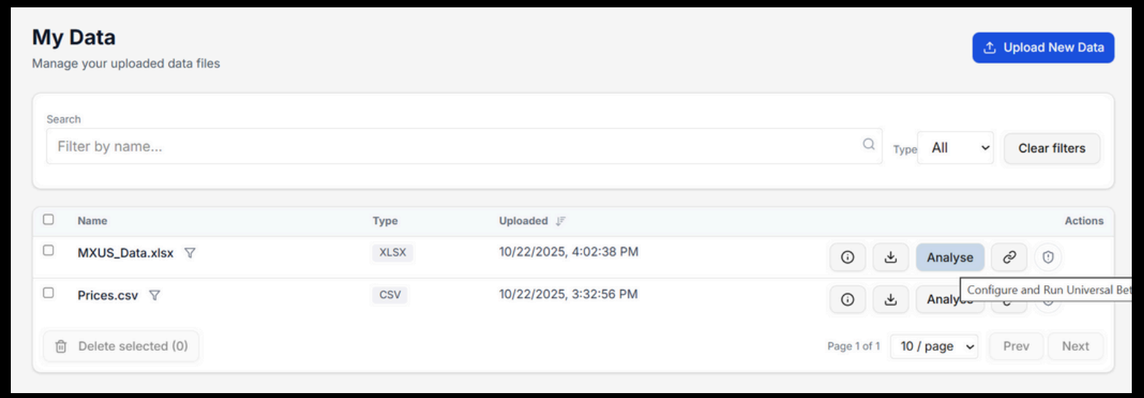
Implementation Details: Analysis

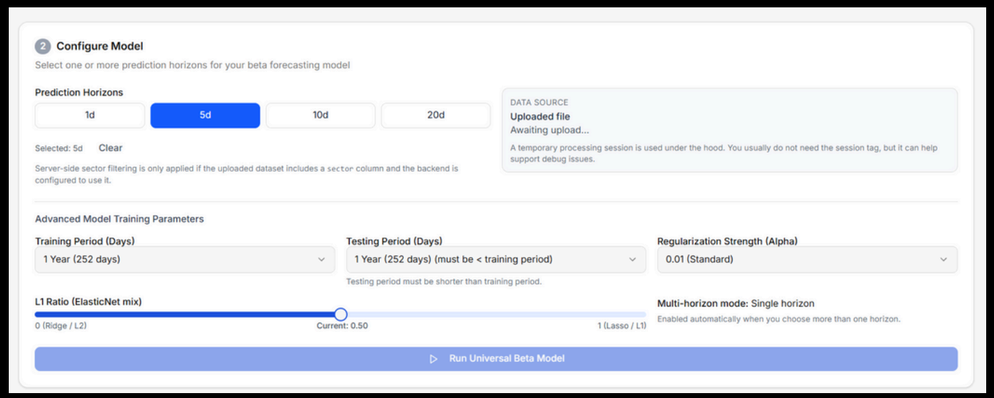


Click ‘Analyse’ on desired Upload

Phawat

Implementation Details: Analysis





Click ‘Analyse’ on desired Upload → Select Desired Model Hyper-parameters

Phawat

Implementation Details: Analysis

Ticker	Created At	β Coefficient ↓	Trend	Volatility	Train β	Lag Train β	AR(1) Corr 10d	AR(1) Corr 20d	AR(1) Corr 60d	Sign Persist 20d	Sign Persist 60d	Return 1d	Return 5d	Return 20d	Vol 63d
CVNA UN	7/25/2024, 12:00:00 PM	0.021	↗	0.054	0.125	0.125	0.244	0.053	0.156	0.368	0.441	0.006	-0.039	-0.008	0.054
MSTR UW	7/25/2024, 12:00:00 PM	0.005	↗	0.060	-0.045	-0.047	0.108	0.109	-0.013	0.737	0.525	-0.040	0.044	0.130	0.060
COIN UW	7/25/2024, 12:00:00 PM	0.004	↗	0.046	-0.036	-0.043	0.132	0.086	-0.115	0.632	0.525	-0.055	0.001	0.097	0.046
SNAP UN	7/25/2024, 12:00:00 PM	0.001	↗	0.044	0.012	0.016	-0.205	-0.141	-0.023	0.368	0.373	-0.010	-0.078	-0.210	0.044
RIVN UW	7/25/2024, 12:00:00 PM	0.001	↗	0.050	0.081	0.080	-0.408	-0.025	0.020	0.579	0.458	-0.001	-0.036	0.114	0.050
RBLX UN	7/25/2024, 12:00:00 PM	-0.002	↘	0.035	-0.027	-0.026	0.146	-0.105	0.002	0.474	0.492	0.015	0.019	0.098	0.035
SMCI UW	7/25/2024, 12:00:00 PM	-0.002	↘	0.048	-0.035	-0.039	-0.088	-0.065	-0.024	0.526	0.492	-0.022	-0.137	-0.162	0.048

Click ‘Analyse’ on desired Upload → Select Desired Model Hyper-parameters → View Results on the ‘Results’ page

Phawat

Model

The model predicts short-term stock momentum (“b”) across multiple time horizons by analysing predictive signals derived from historical return behavior.

Hoang

Model

The model predicts short-term stock momentum (“b”) across multiple time horizons by analysing predictive signals derived from historical return behavior.

Possible time horizons:

- 1 day (daily)
- 5 days (weekly)
- 10 days (fortnightly)
- 20 days (monthly)

Hoang

MEAN REVERSION VS TRENDING

Positive Momentum / Mean Reversion

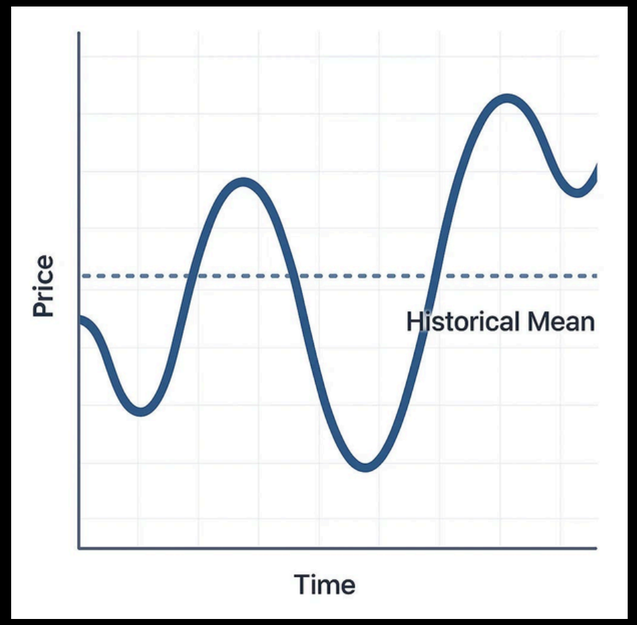


Image:
https://medium.com/@jlabs/hurst-mean-reversion-of-time-series-data-08b608479656

Negative Momentum / Trending

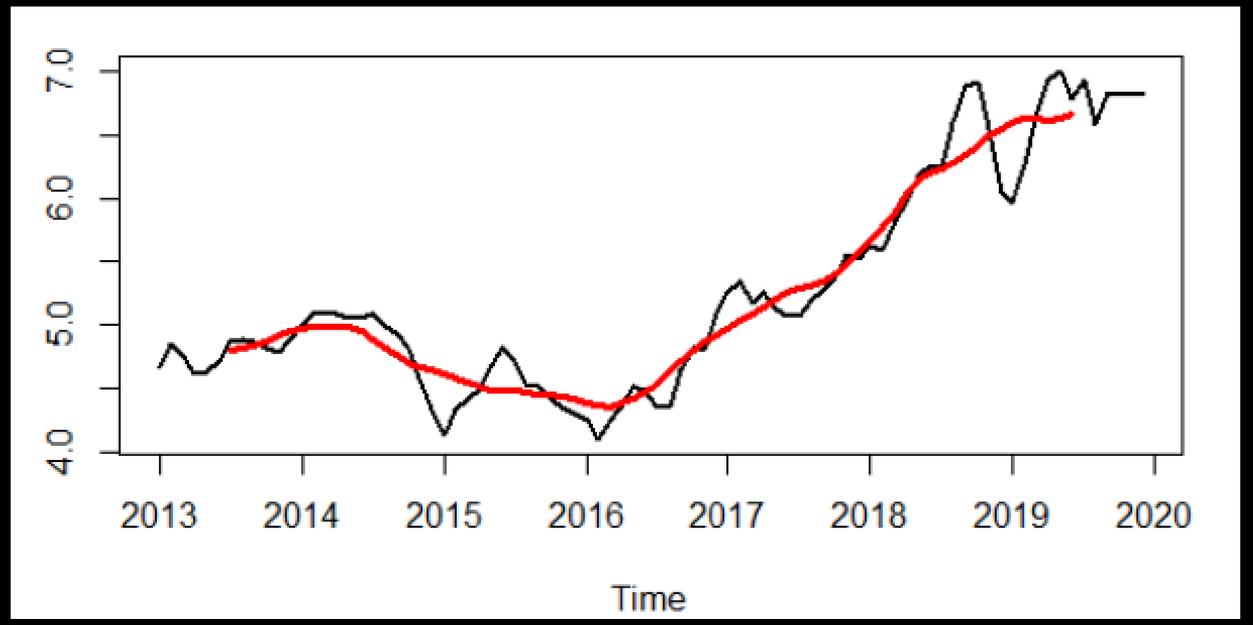
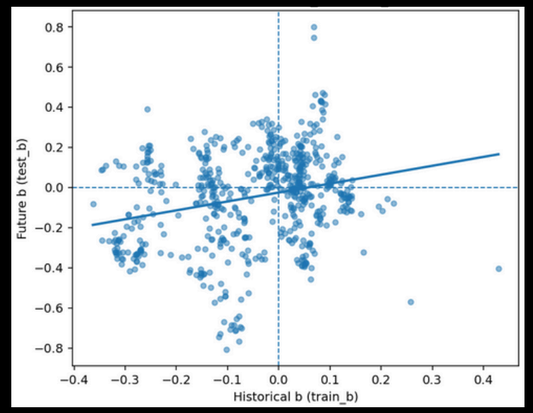


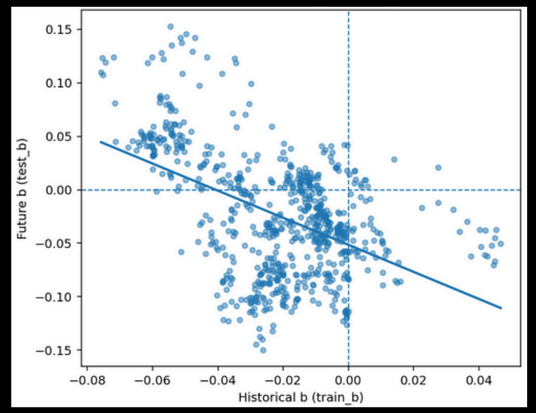
Image:
https://www.r-bloggers.com/2020/03/trend-forecasting-models-and-seasonality-with-time-series/

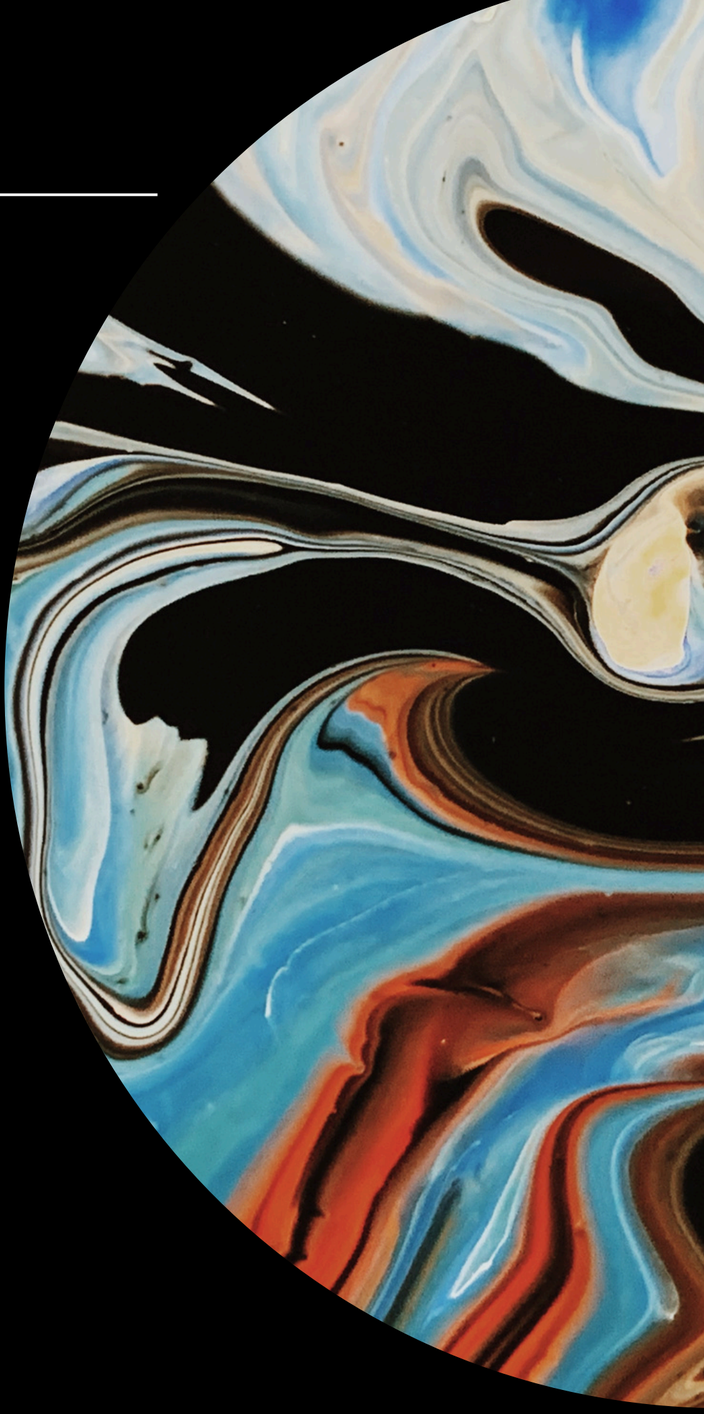
Hoang

TWO PARTS OF THE MODEL

1. Find historical b



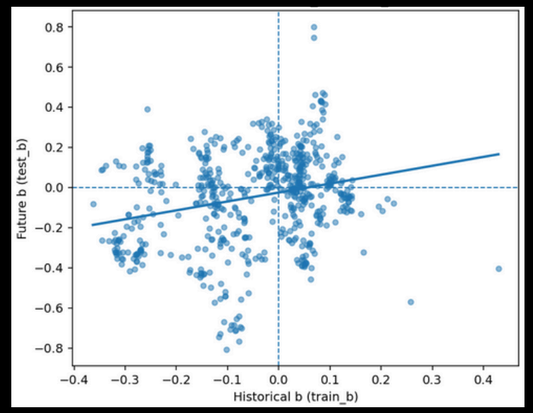


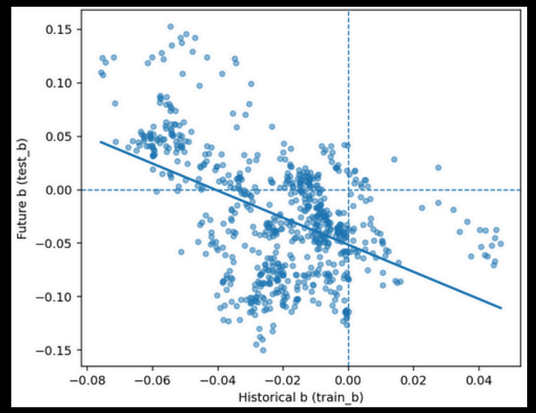


Shawna

TWO PARTS OF THE MODEL

1. Find historical b

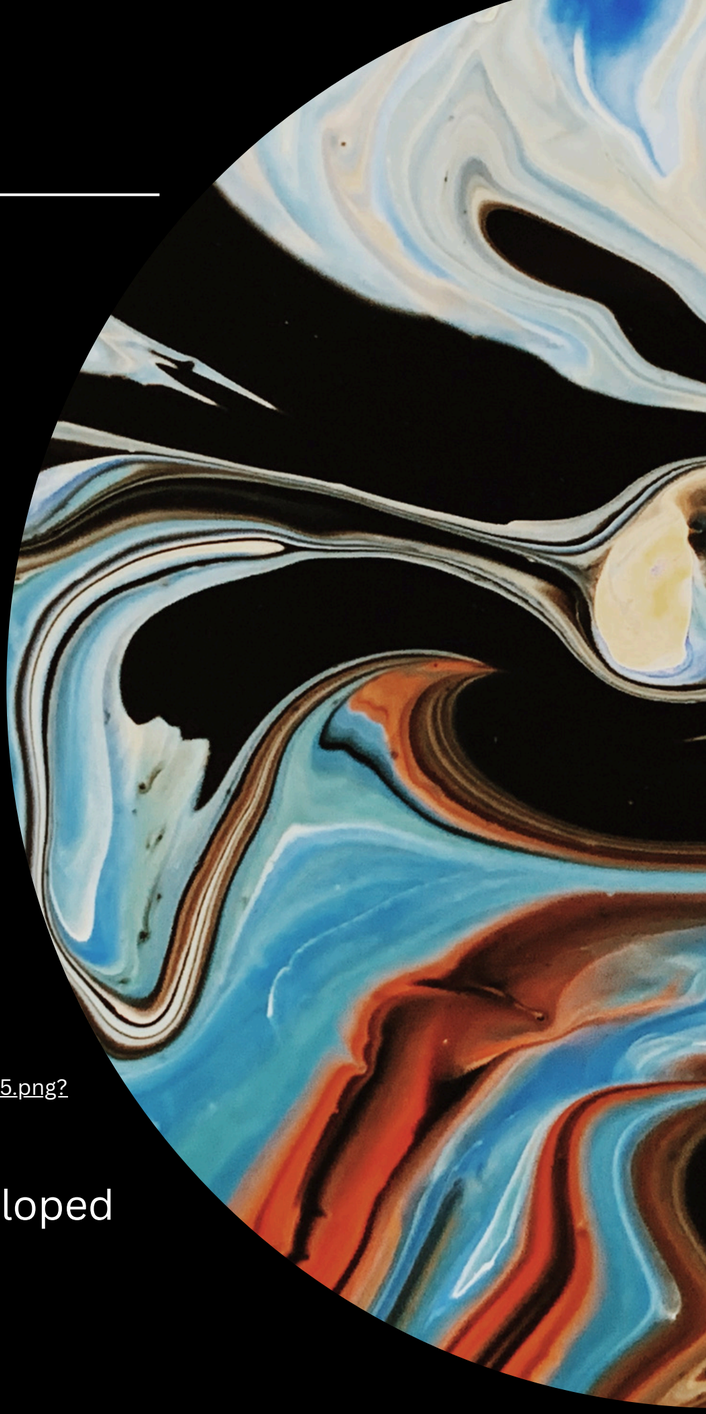




2. Predict future b value

Image: https://cdn-icons-png.freepik.com/256/12492/12492225.png?semt=ais_white_label

We have used Python packages such as Numpy, Pandas, StatsModels, and our self-developed algorithms.



Shawna

HISTORICAL B - MODEL

We use ARIMA(1, 0, 0)

$$R_{t+1} = C + b \cdot R_t$$

Where:

- R_{t+1} is the return of time $t + 1$
- C is a constant (representing the y intercept)
- b is the mean reversion coefficient
- R_t is the return of time t

Shawna

HISTORICAL B - MODEL

We use ARIMA(1, 0, 0)

$$R_{t+1} = C + \underline{b \cdot} R_t$$

what we are trying to find

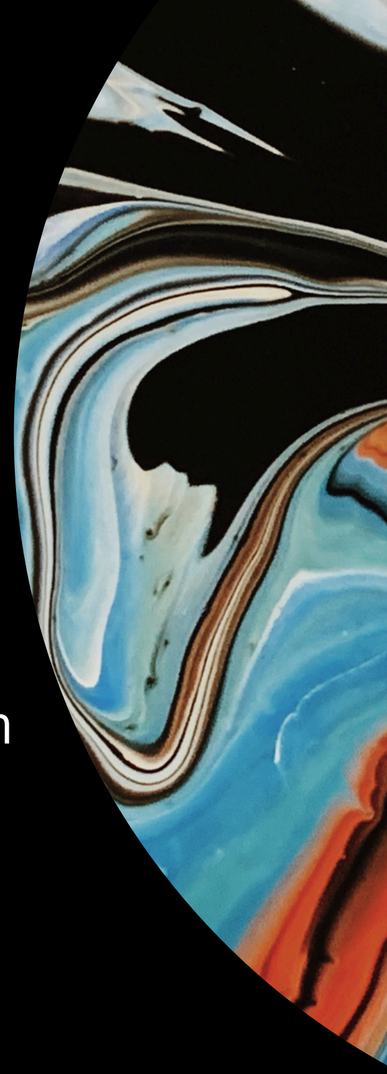
Where:

- R_{t+1} is the return of time $t + 1$
- C is a constant (representing the y intercept)
- b is the mean reversion coefficient
- R_t is the return of time t

Shawna

PREDICTING FUTURE B - PREDICTORS

- AR(1)-layer inputs
  - train_b
  - lag_train_b
- Autocorrelation
  - ar1_corr_10d
  - ar1_corr_20d
  - ar1_corr_60d
- Sign persistence
  - sign_persist_20d
  - sign_persist_60d
- Recent returns
  - ret_1d
  - ret_5d
  - ret_20d
- Volatility
  - vol_63d
  - vol_126d
  - vol_252d
  - vol_756d
- Trend / Momentum level
  - Z_ma_20d
- Cross-Sectional Rank of Volatility
  - cs_rank_vol_63d
  - cs_rank_vol_126d
  - cs_rank_vol_252d
  - cs_rank_vol_756d
- Cross-sectional ranks
  - cs_rank_ret_1d
  - cs_rank_ret_5d
  - cs_rank_ret_20d
  - cs_rank_z_ma_20d
- Cross-sectional dispersion
  - xsec_disp_1d
  - xsec_disp_30d



Shawna

ELASTIC NET

Combines two regularization methods and their strengths:

- L1 (Lasso) - Feature selection
- L2 (Ridge) - Stability

Shawna

ELASTIC NET

Combines two regularization methods and their strengths:

- L1 (Lasso) - Feature selection
- L2 (Ridge) - Stability

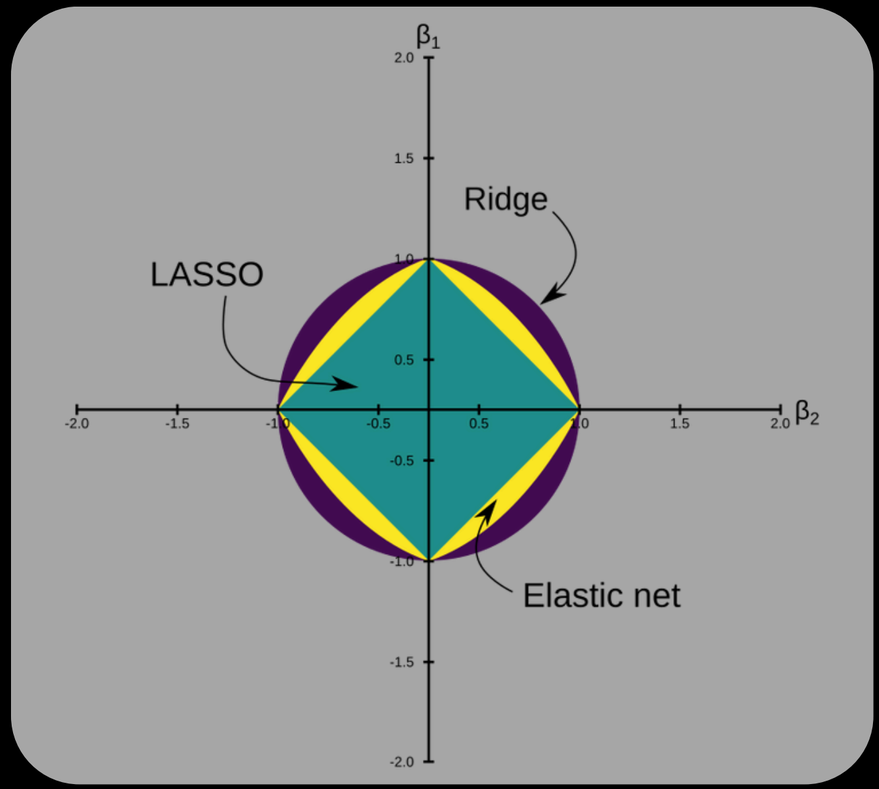


Image:
https://analyticsindiamag.com/ai-trends/hands-on-tutorial-on-elasticnet-regression/

Shawna

ELASTIC NET

Combines two regularization methods and their strengths:

- L1 (Lasso) - Feature selection
- L2 (Ridge) - Stability

Creates an accurate, flexible, and robust model.

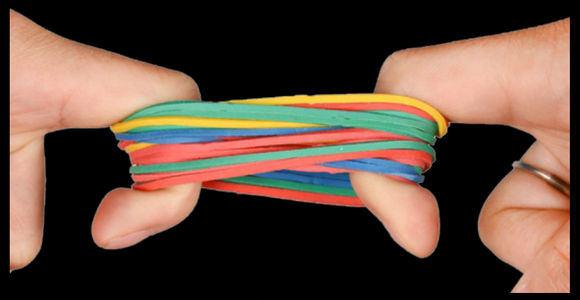


Image:
https://www.pngegg.com/en/png-bamjq/download

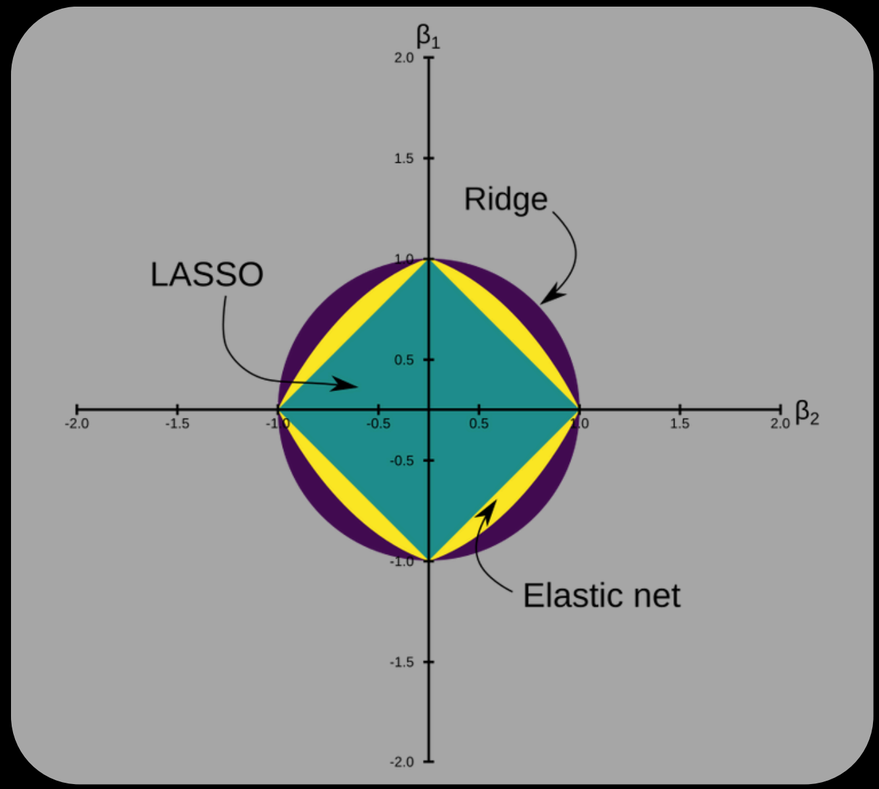


Image:
https://analyticsindiamag.com/ai-trends/hands-on-tutorial-on-elasticnet-regression/

Shawna

MODEL EVALUATION

Metrics:

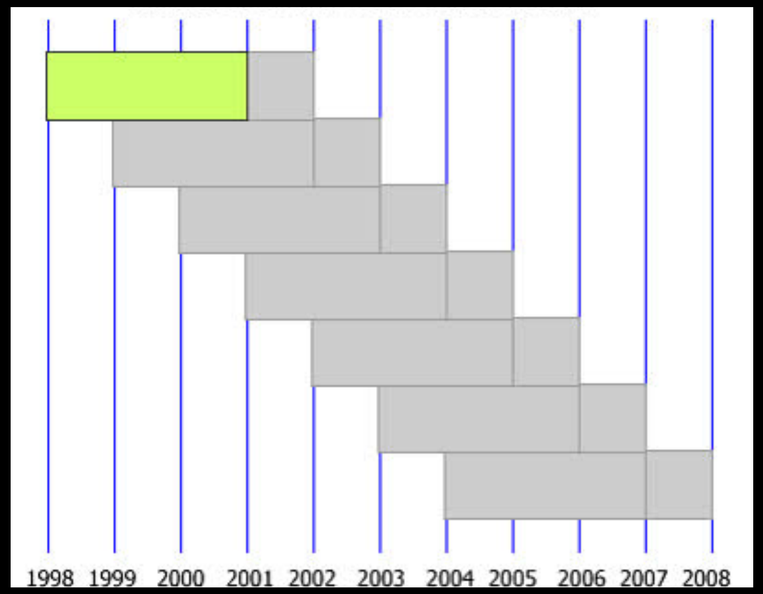
- MAE (0.06)
- RMSE (0.08)
- Directional accuracy (59.5%)
- Global R^2 (0.036)
- Mean R^2 per fold (-0.15)
- Weighted R^2 (-0.18)

Shawna

MODEL EVALUATION

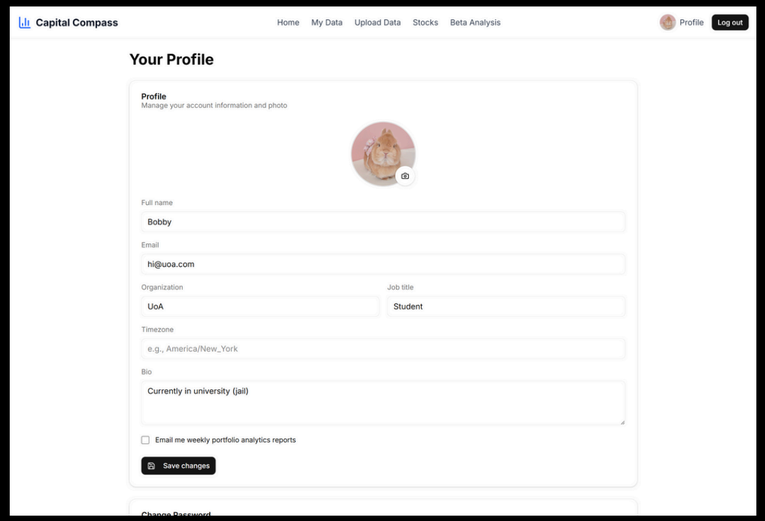
Metrics:

- MAE (0.06)
- RMSE (0.08)
- Directional accuracy (59.5%)
- Global R^2 (0.036)
- Mean R^2 per fold (-0.15)
- Weighted R^2 (-0.18)



Shawna

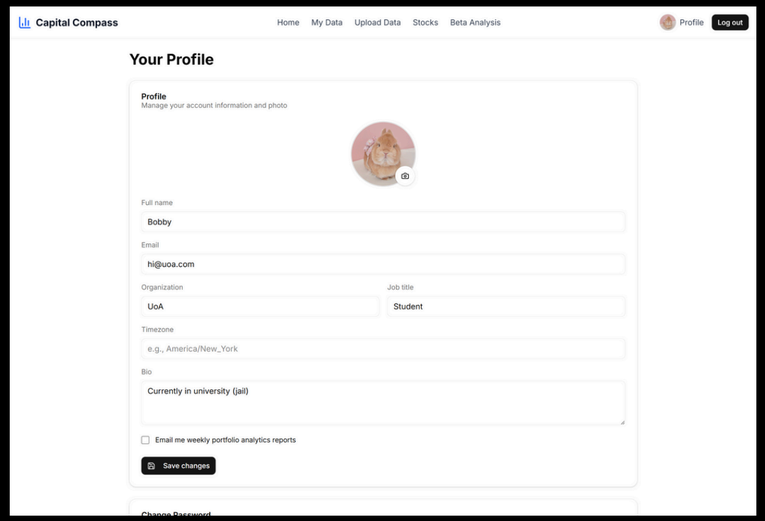
ADDITIONAL FEATURES



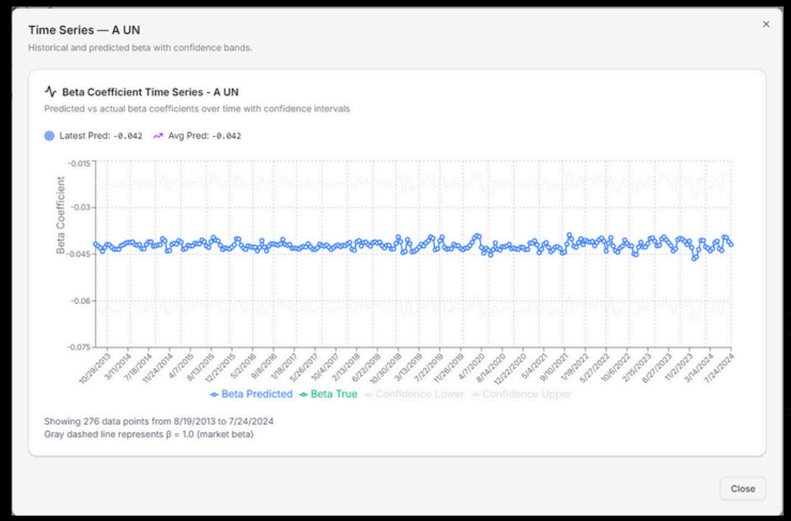
Profile Customization

Henri

ADDITIONAL FEATURES



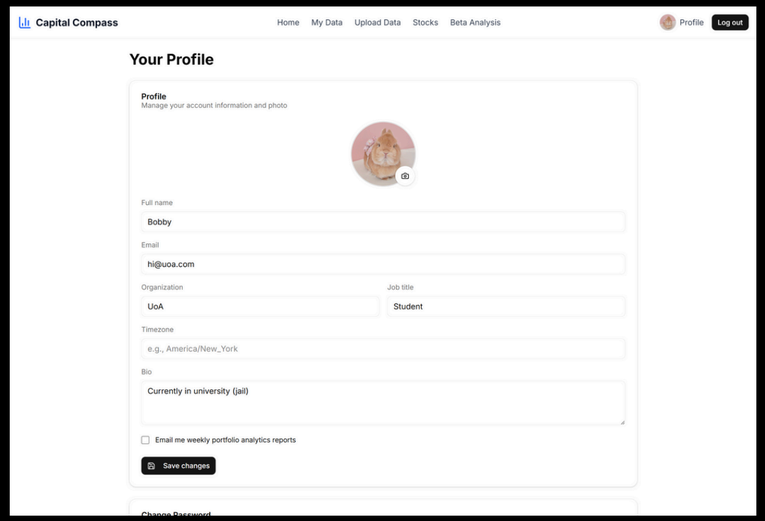
Profile Customization



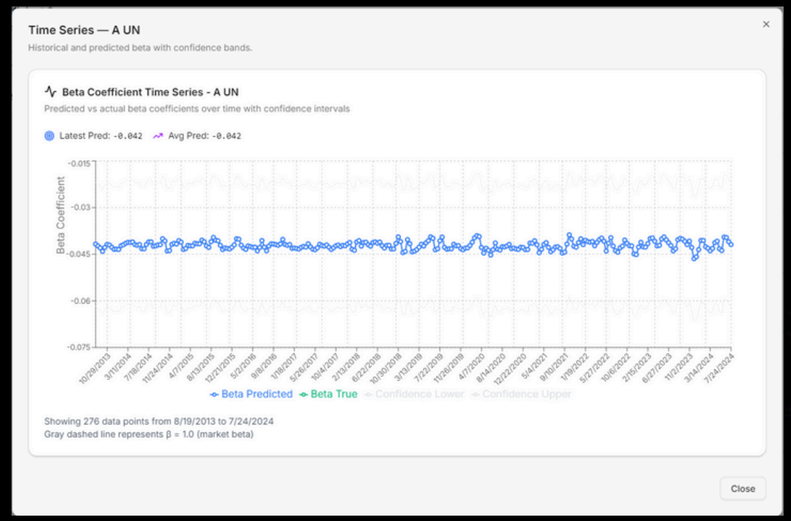
Stock Beta Time Series

Henri

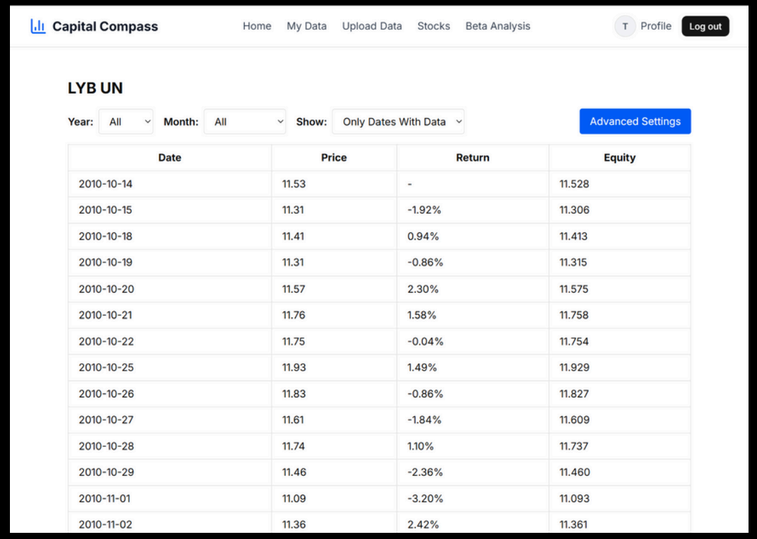
ADDITIONAL FEATURES



Profile Customization



Stock Beta Time Series



Date	Price	Return	Equity
2010-10-14	11.53	-	11.528
2010-10-15	11.31	-1.92%	11.306
2010-10-18	11.41	0.94%	11.413
2010-10-19	11.31	-0.86%	11.315
2010-10-20	11.57	2.30%	11.575
2010-10-21	11.76	1.58%	11.758
2010-10-22	11.75	-0.04%	11.754
2010-10-25	11.93	1.49%	11.929
2010-10-26	11.83	-0.86%	11.827
2010-10-27	11.61	-1.84%	11.609
2010-10-28	11.74	1.10%	11.737
2010-10-29	11.46	-2.36%	11.460
2010-11-01	11.09	-3.20%	11.093
2010-11-02	11.36	2.42%	11.361

Historical Stock Data Explorer

Henri

CHALLENGES & LESSONS LEARNED

What Worked

- Integrating Next.js ↔ Django REST ↔ Postgres
- AWS Integration with the webapp
- Handling NaN serialization and JSON errors
- Uploading of Properly formatted Excel files
- Calculation of β-coefficient, factors, etc.
- Intuitive and feature-rich frontend
- Robust & flexible model

CHALLENGES & LESSONS LEARNED

What Worked

- Integrating Next.js ↔ Django REST ↔ Postgres
- AWS Integration with the webapp
- Handling NaN serialization and JSON errors
- Uploading of Properly formatted Excel files
- Calculation of β-coefficient, factors, etc.
- Intuitive and feature-rich frontend
- Robust & flexible model

Challenges Faced

- Model Runtime
- Choosing what predictors to use
- Choosing what predictive model to use
- Prevent look-ahead bias for backtesting
- Clustering
- Clear Communication with client and team members

Hoang

CHALLENGES & LESSONS LEARNED

What Worked

- Integrating Next.js ↔ Django REST ↔ Postgres
- AWS Integration with the webapp
- Handling NaN serialization and JSON errors
- Uploading of Properly formatted Excel files
- Calculation of β-coefficient, factors, etc.
- Intuitive and feature-rich frontend
- Robust & flexible model

Lessons Learned

- Clear communications and outline from the start
- Clear documentation accelerates debugging
- Iteration > perfection for complex pipelines
- Time management

Challenges Faced

- Model Runtime
- Choosing what predictors to use
- Choosing what predictive model to use
- Prevent look-ahead bias for backtesting
- Clustering
- Clear Communication with client and team members

Charles

CHALLENGES & LESSONS LEARNED

What Worked

- Integrating Next.js ↔ Django REST ↔ Postgres
- AWS Integration with the webapp
- Handling NaN serialization and JSON errors
- Uploading of Properly formatted Excel files
- Calculation of β-coefficient, factors, etc.
- Intuitive and feature-rich frontend
- Robust & flexible model

Lessons Learned

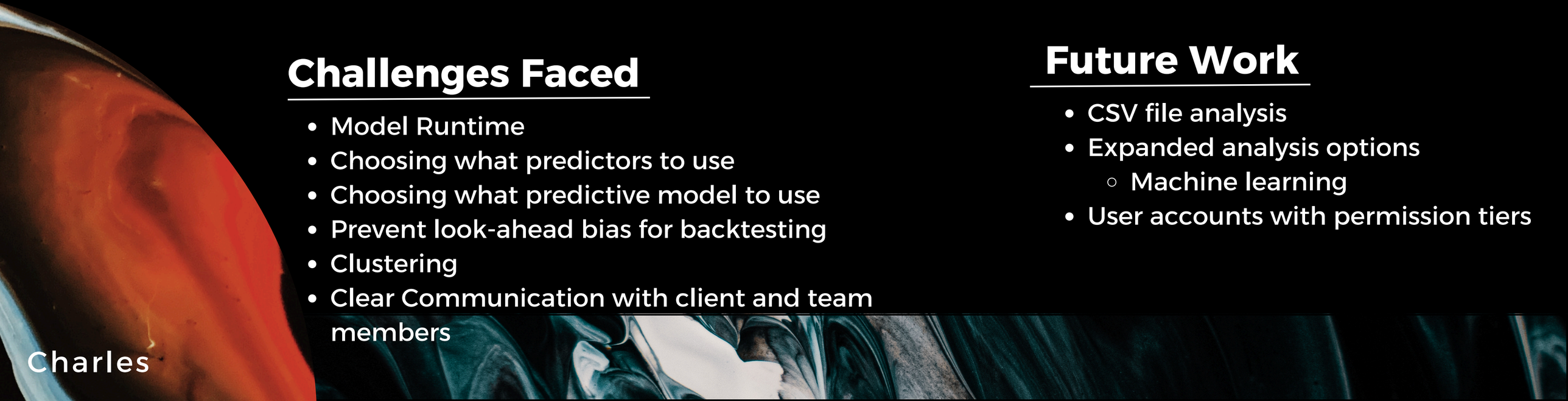
- Clear communications and outline from the start
- Clear documentation accelerates debugging
- Iteration > perfection for complex pipelines
- Time management

Challenges Faced

- Model Runtime
- Choosing what predictors to use
- Choosing what predictive model to use
- Prevent look-ahead bias for backtesting
- Clustering
- Clear Communication with client and team members

Future Work

- CSV file analysis
- Expanded analysis options
  - Machine learning
- User accounts with permission tiers



Charles

PROJECT GOAL EVALUATION

Our Goals:

- Develop a Robust Mean-Reversion Analysis Model
- Incorporate Peer and Market Context Through Clustering
- Create a Weighted Scoring System with MVP Backtesting
- Design a User-Friendly Web Platform for Analysis Delivery

=> We have achieved all of these goals except clustering. We tried to cluster stock based on given price data but could not find any significant result.

Hoang

THANK YOU!

THANK YOU!

Special thanks to

Nicholas Bagnall
Anna Trofimova
Andrew Balemi
Matt Edwards