Table of Contents

1. Introduction

The M6 Financial Forecasting Competition is held to gather empirical evidence that sheds light on the causes and implications of the Efficient Market Hypothesis (EMH) paradox. According to this hypothesis, financial share prices reflect all available information, making investment outperformance relative to the market impossible.

Paradoxically, there have been instances where the explanation of EMH fails to hold, such as when the market observes both the poor performance of active funds and the exceptional performance of individuals such as Warren Buffet.

In this competition, our team demonstrates that consistent forecasting performance on the financial share prices in all macro-environment is possible through data engineering and diverse modeling approaches.

2. Traditional Forecasting Framework with ML

2.1 Natural Language Processing

2.1.1 Data Source

We decided to incorporate financial news corpus as part of the data used by our Machine Learning (ML) models. The financial news corpus contains news published in Wall Street Journal.

An automated system exists to map each financial news to a specific sector to enhance the usability of the financial news corpus. With sector-specific mappings, we can compute the sentiment in each sector per time interval, which is fed into our ML models downstream.

2.1.2 Natural Language Processing Model

Our team leveraged Bidirectional Encoder Representations from Transformers (BERT) in our Natural Language Processing (NLP) tasks. BERT is a family of Transformers released by Google in 2019 and works by pretraining deep bidirectional representations from unlabelled text [1,2]. To use BERT, users have to train the final layer with the tasks intended, which spawns various BERT models like FinBERT for financial sentiment analysis and BERTopic for topic modeling [3,4].

To obtain sentiment from natural text, we leverage FinBERT, the BERT model that was further trained on TRC2-financial and Financial PhraseBank. We can retrieve a more accurate representation of the sentiment as the financial context allows FinBERT to accurately decipher the intention behind the sentences more precise than other language models [3].

2.1.3 Attention on Financial News

Similar to the ideation of transformers, we have implemented an attention layer on the financial news that was converted into features. Unfortunately, not all news is relevant to the market, which ultimately reflects the price movements of the equities. Hence, it is crucial to sieve irrelevant news to reduce noise and improve the feature relevancy to the downstream machine learning models. The attention layer is modeled by the mathematical function below:

$$\mu_{sentiment} = \min\left(\frac{e^{\frac{a_x}{a_{max}}}}{e^{0.8}}, 1\right) \quad (1)$$

Where $\mu_{sentiment}$ is the average sentiment of the equities within a specific period, a_x is the attention of the equities within the specified period, and a_{max} is the peak attention of the equities within the specific period.

2.2 Time-series Features

Timeseries data are subjected to components like trends, seasonality, and cycles. The longer the dataset, the more apparent the time-series features will exhibit.

In the financial context, we always see trends, seasonality, and cycles. There has been empirical evidence that trend-based following investment rule has generated a positive return on average across various macro environments, including high and low-interest rate regimes, high and low inflation periods, war and peace, recession, and boom [5].

We have computed 68 time-series features across 14 major categories. The 14 major categories are listed in Table 1. The full details of the time-series features used are listed in Table A1 under Appendix A.

Table 1. Major categories of time-series features

Time Series Major Categories	
ACF PACF Features	BOCP Detector
CUSUM Detector	Holt Parameters
Holt-Winters Parameters	Level Shift Features
Nowcasting	Outlier Detector
Robust Statistics Detector	Seasonality
Special AC	Statistics
STL Features	Trend Detector

2.3 Technical Indicators and Macro Trends

2.3.1 Technical Indicators

Technical analysis is a mainstream investment analysis for active traders in the current environment [6]. Technical indicators attempt to model the short-term price movements of equities based on heuristic or pattern-based calculations [7].

In this competition, we have incorporated various technical indicators as features for our ML models. The indicators we have employed are shown in Table 2.

Table 2. Technical Indicators

Technical Indicators Employed
Welles Wilder's Directional Movement Index
Aroon Indicator
Commodity Channel Index
Chande Momentum Oscillator
MACD Oscillator
Relative Strength Index
Stochastic Oscillator / Stochastic Momentum Index
Volatility
Volume-Weighed Average Price

2.3.2 Macro Trends

Macro trends undeniably affect the price movements of equities. To model the change in the macro environment, we have introduced factors such as unemployment rates, continuing claims, OBFR, SOFR, and Money Supplies as features for the downstream ML models.

2.4 Feature Expansion and Elimination

As the relationship between each feature and the target variable may not have a fixed period difference, our team took inspiration from ACF-PCF analysis. We expanded each feature by lagging each feature to a magnitude of 9 periods.

We employed three independent feature elimination methods with a predefined threshold level to ensure that the ML models' performance is not affected by the introduction of a significant increase in features. If the features created have less significance than the threshold level, the feature will be removed from the final model training.

2.5 Stacking with ML Models

After defining our target variable (equity position band), we perform model training and calibration with the training data.

In the prediction stage, we have defined three "weak" learners to take on the prediction tasks. We employed LightGBM, XGBoost, and Distributed Random Forest. Tree-based learning models have shown remarkable performance in predicting equities position [8].

To combine the prediction results of the ML models, we employed an ensemble approach named stacking. Stacking creates a meta-learner model that takes the prediction output of the "weak" learners and provides us with a final prediction output. The workflow of stacking is illustrated in Figure 1.

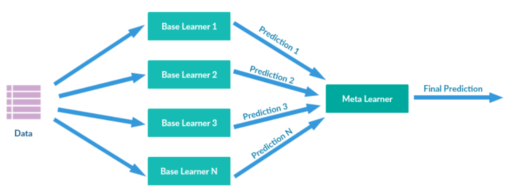

Figure 1. Stacking Workflow [9]

3. Forecasting with Covariance Matrices

Covariance is widely used in the investment world, where the investor compares the price movement of two different assets. Having assets that are not highly correlated to each other helps reduce the volatility, improving the Sharpe Ratio.

In the same vein, covariance can be used to forecast the price movement of differing assets relative to each other. The M6 Competition is focused on the relative position of equities rather than the absolute return or price movement of the equities. Hence, there are significant advantages to employing the covariance forecasting method for this competition.

Covariance forecasting has also recently picked up momentum in the investment world, where new covariance methods and models have been proposed to forecast uncertainties in the equity world [10].

The covariance of two equities is computed below:

$$cov_{A,B} = \frac{\sum(R_A - \overline{R_A})\,(R_B - \overline{R_B})}{N-1} \tag{2}$$

Where R is the returns of the equities within the specified period.

To compute the covariance matrix, we have to calculate the covariance of all 100 equities in the competition, which is C(100,2) combinations.

Once we have the covariance matrix, we can predict the relative ranking through modeling uncertainties.

The two uncertainties we have defined are:
(1) The uncertainties of covariance factor
(2) The uncertainties of returns

We have employed Monte Carlo simulation to assist us in modeling the two uncertainties we have defined. And through the simulations, we average the rank probabilities that each of the 100 equities lands.

4. Forecasting with Conformal Quantile Regression

4.1 Quantile Regression

Differing from most mean-based predictors, Quantile Regression (QR) is a statistical method that allows users to understand the non-linear relationship of target variables to the feature variables.

The formula for QR is given in Eq. 3 below:

$$Q_\tau(y) = B_0(\tau) + B_1x_1(\tau) + \cdots + B_1x_1(\tau) \qquad (3)$$

Where τ is the quantile specified, B_n is the weight for each variable, and x_n is the feature variables [11]. It is also governed by a loss function as given in Eq. 4.

$$L_\tau = max\begin{cases} y_\tau - f(x)_\tau \\ y_{1-\tau} - f(x)_{1-\tau} \end{cases} \qquad (4)$$

Where L is loss is the specified quantile [12]. With QR, we can model the uncertainties in our models' prediction and adjust our decisions based on the volatility the models' exhibit.

4.2 Conformal Prediction

Unlike the majority of popular forecasting approaches, Conformal Prediction (CP) allows us to construct distribution-free predictions within a finite sample [13]. Hence, it guarantees to contain the ground truth within a specified probability, as shown in Figure 2 [14].

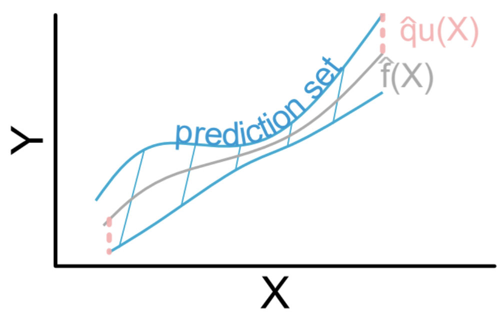

Figure 2. Conformal Regression Prediction [14]

The function of CP is modeled in Eq. 5 below:

$$1 - \alpha \leq P\{y\epsilon\tau(X_{n+1})\} \leq C \qquad (5)$$

Where α refers to the confidence level, P refers to the prediction set, and C refers to the calibration set.

4.3 Conformal Quantile Regression

Despite promising theories behind QR and CP, they have inherent disadvantages to be utilized compared to other forecasting methods. QR suffers inaccuracy of its extreme quantiles prediction on new data, and CP tends to provide static bands throughout the whole interval.

Conformal Quantile Regression (CQR) is a technique that combines statistical efficiency in QR while leveraging on the distribution-free coverage in CP proposed by a group of Stanford scholars [15].

CQR is adaptative, indicating that the width of the prediction band is highly correlated to the confidence of the prediction. This would reduce the disadvantages of QR and CP if they were to be used individually.

CQR requires three sets of data, the training data, the calibration data, and the testing data. To demonstrate CQR, we first train any QR on the training data. Then, we use QR to conduct interval predictions on the calibration dataset, which we could leverage on the results to compute the conformity score based on the principles of CP. Lastly, we could combine these two approaches by performing the prediction on the testing data and subtracting one end of the QR extremities band from the conformity score.

The formula for the conformity score is given in Eq. 6:

$$CS_i = \max(|\, y_i - f(x_i)|) \qquad (6)$$

Where CS is the conformity score, y is the target variable, and f(x) is the prediction on the calibration set.

In the investment world, knowing the probability of price return within the regression band that guarantees ground truth within the user-specified confidence level allows investors to perform data-driven decisions with an appropriate risk management framework.

5. Combining Forecasting Systems

Ensembling different forecasting systems (FS) often utilize the weightage system, modeled by Eq. 7 below:

$$FS = W_1 FS_1 + \cdots + W_N FS_N, \quad s.t. \sum_{i=0}^{N} W_i = 1 \tag{7}$$

Where W is the weight of each FS, and N is the number of FS.

5.1 Equally Weighted Forecasts

As its name suggests, an equally weighted forecast system (EWFS) assigns a static $W_{\frac{1}{N}}$ to Eq 7, which assumes equal weightage of all FS. This approach is often used by researchers if there is no additional information that could be gained to enhance the performance of the FS.

In theory, the performance of EWFS is subjected to Eq. 8:

$$P_{Worst} \geq P_{EWFS} \geq P_{Best} \tag{8}$$

Where the performance of the EWFS is better than the worst FS and worse than the best FS ensembled.

5.2 Optimization

The optimization approach attempts to calculate the optimal weightage of each FS to produce an ensembled FS that grants superior performance compared to all FS ensembled based on past performance. Despite showing tremendous success in functions like logistics or demand forecasting, it has yet to acquire significant success in equities forecasting [16].

In equities investment, optimizing FS produces an ensembled FS that may be better or worse than each FS utilized [16].

5.3 Boot-strapping

To produce an FS that is better than EWFS yet limits the downside of fully optimized FS, we leveraged the concept of boot-strapping, which combines EWFS and optimized FS.

Our boot-strapped forecasting system (BSFS) computes the optimized weightage for each FS in a fixed period N. With the optimized weightage in each period N, we combine those weights through a factor of M, which is the number of periods within the predictive horizon. BSFS is modeled by Eq. 9 below:

$$BSFS = \frac{1}{M} \sum_{i=0}^{M} W_1 FS_1 + \cdots + W_N FS_N \tag{9}$$

Through boot-strapping, we can increase the performance of our ensemble FS while preventing over-optimization. The less volatility the optimized weight of each FS in each period, the closer the BSFS models optimized FS.

6. Conclusion

A combination of diverse modeling approaches and accurately extracting value from data allows the team to perform relatively well in the financial competition.

To enhance the performance of the FS in the M6 Competition, emphasis can be placed on acquiring alternative data and a more rigorous data engineering process to aid the performance of Traditional Framework with ML and CQR.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," Jun. 2017.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding," 2019.

[3] D. Aract, "FinBERT: Financial Sentiment Analysis with Pre-trained Language Models," Jun. 2019.

[4] M. Grootendorst, "Topic Modeling with BERT," *Medium*, 06-Oct-2020. [Online]. Available: https://towardsdatascience.com/topic-modeling-with-bert-779f7db187e6.

[5] B. Hurst, Y. H. Ooi, and L. H. Pedersen, "A century of evidence on trend-following investing," *The Journal of Portfolio Management*, vol. 44, no. 1, pp. 15–29, 2017.

[6] L. Yin and Q. Yang, "Predicting the oil prices: Do technical indicators help?," *Energy Economics*, vol. 56, pp. 338–350, 2016.

[7] J. Chen, "Technical indicator definition," *Investopedia*, 29-Jun-2022. [Online]. Available: https://www.investopedia.com/terms/t/technicalindicator.asp.

[8] E. K. Ampomah, Z. Qin, and G. Nyame, "Evaluation of tree-based ensemble machine learning models in predicting stock price direction of movement," *Information*, vol. 11, no. 6, p. 332, 2020.

[9] S. Setunga, "Stacking in Machine Learning," *Medium*, 02-Jan-2020. [Online]. Available: https://supunsetunga.medium.com/stacking-in-machine-learning-357db1cfc3a.

[10] E. Symitsi, L. Symeonidis, A. Kourtis, and R. N. Markellos, "Covariance forecasting in equity markets," *SSRN Electronic Journal*, 2018.

[11] S. Tadakaluru, "What is quantile regression? Introduction to quantile regression," 22-Mar-2022. [Online]. Available: https://www.mygreatlearning.com/blog/what-is-quantile-regression/.

[12] C. Garcia, "Quantile Regression," *GitHub*, 06-Jul-2021. [Online]. Available: https://github.com/cgarciae/quantile-regression.

[13] S. Tadakaluru, "What is quantile regression? Introduction to quantile regression," 22-Mar-2022. [Online]. Available: https://www.mygreatlearning.com/blog/what-is-quantile-regression/.

[14] A. Angelopoulos and S. Bates, "A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification," Jul. 2021.

[15] Y. Romano, E. Patterson, and E. Candes, "Conformalized Quantile Regression," 2019.

[16] R. Carver, *Systematic trading: A unique new method for designing trading and investing systems*. Petersfield, Hampshire: Harriman House Ltd., 2015.

Appendix A. Time Series Features

Table A1.1 Time Series Features Employed

Category	Variable	Definition
ACF PACF Features	y_acf1	First ACF value of the original series
	y_acf5	Sum of squares of first 5 ACF values of original series
	diff1y_acf1	First ACF value of the differenced series
	diff1y_acf5	Sum of squares of first 5 ACF values of differenced series
	diff2y_acf1	First ACF value of the twice-differenced series
	diff2y_acf5	Sum of squares of first 5 ACF values of twice-differenced series
	y_pacf5	Sum of squares of first 5 PACF values of original series
	diff1y_pacf5	Sum of squares of first 5 PACF values of differenced series
	diff2y_pacf5	Sum of squares of first 5 PACF values of twice-differenced series
	seas_acf1	Autocorrelation coefficient at the first seasonal lag
	seas_pacf1	Patial Autocorrelation coefficient at the first seasonal lag
BOCP Detector	bocp_num	Number of changepoints detected by BOCP Detector
	bocp_conf_max	Max value of the confidence of the changepoints detected
	bocp_conf_mean	Mean value of the confidence of the changepoints detected
CUSUM Detector	cusum_num	Number of changepoints, either 0 or 1
	cusum_conf	Confidence of the changepoint detected, 0 if not changepoint
	cusum_cp_index	Index or position of the changepoint detected within the time series
	cusum_delta	Delta of the mean levels before and after the changepoint
	cusum_llr	Log-likelihood ratio of changepoint
	cusum_regression_detected	Boolean - whether regression is detected by CUSUM
	cusum_stable_changepoint	Boolean - whether changepoint is stable
	cusum_p_value	P-value of changepoint

Table A1.2 Time Series Features Employed (cont.)

Category	Variable	Definition
Holt Parameters	holt_alpha	Level parameter of the Holt model
	holt_beta	Trend parameter of the Holt model
Holt-Winters Parameters	hw_alpha	Level parameter of a fitted Holt-Winter's model
	hw_beta	Trend parameter of a fitted Holt-Winter's model
	hw_gamma	Seasonal parameter of a fitted Holt-Winter's model
Level Shift Features	level_shift_idx	Location of the maximum mean value difference, between two consecutive sliding windows
	level_shift_size	Size of the maximum mean value difference, between two consecutive sliding windows
Nowcasting	nowcast_roc	Indicating return comparing to step n back
	nowcast_ma	Indicating moving average in the past n steps
	nowcast_mom	Indicating momentum
	nowcast_lag	Indicating lagged value at the past n steps
	nowcast_macd	Moving average convergence divergence is a trend-following momentum indicator that shows the relationship between two moving averages of a security's price
	nowcast_macdsign	MACD combined with EWMA with a decay window of 9 and minimum periods of 8
	nowcast_macddiff	MACD minus MACDsign
Outlier Detector	outlier_num	Number of outliers
Robust Statistics Detector	robust_num	Number changepoints detected by the Robust Stat Detector
	robust_metric_mean	Mean of the Metric values from the Robust Stat Detector
Seasonality	seasonal_period	Detected seasonality period
	trend_mag	Slope acquired via fitting simple linear regression model on the trend component as trend magnitude
	seasonality_mag	Difference between the 95 percentile and 5 percentile of the seasonal component as the seasonality magnitude
	residual_std	Standard deviation of the residual component
Special AC	firstmin_ac	The time of first minimum in the autocorrelation function
	firstzero_ac	The time of first zero crossing the autocorrelation function

Table A1.3 Time Series Features Employed (cont.)

Category	Variable	Definition
Statistics	length	Length of the time series array
	mean	Average of the time series array
	var	Variance of the time series array
	entropy	Getting normalized Shannon entropy of power spectral density
	lumpiness	Calculating the variance of the chunk-wise variances of time series
	stability	Calculating the variance of chunk-wise means of time series
	flat_spots	Maximum run-lengths across equally-sized segments of time series
	hurst	Hurst exponent measures the amount that a series deviates from a random walk
	std1st_der	Standard deviation of the first derivative
	crossing_points	The number of times a time series crosses the median line
	binarize_mean	Convert time series array into a binarized version
	unitroot_kpss	Test a null hypothesis that an observable time series is stationary around a deterministic trend using KPSS
	heterogeneity	Engle's Lagrange Multiplier test to check for the existence of Autoregressive Conditional Heteroscedasticity (ARCH)
	histogram_mode	Measures the mode of the data vector using histograms with a given number of bins
	linearity	R square from a fitted linear regression
STL Features	trend_strength	Strength of trend
	seasonality_strength	Strength of seasonality
	spikiness	Variance of the leave-one-out variances of the remaining component
	peak	Location of peak
	trough	Location of trough
Trend Detector	trend_num	Number of trends detected by the Kats Trend Detector
	trend_num_increasing	Number of increasing trends
	trend_avg_abs_tau	Mean of the absolute values of Taus of the trends detected