

Momentum Strategy Enhancement with Machine Learning

Sector-based Random Forest Application

COMP4971C - Independent Work (Fall 2019)

WONG Jia Yeung, Leo

November, 2019

Abstract

In this research project, the feasibility of improving "plain" momentum strategy with machine learning method (Random Forest) in financial portfolio construction and optimisation is examined, while simultaneously testing the market efficiency of the Hong Kong stock market.

Contents

1	Introduction	1
2	Disclaimer	1
3	Data, Tools and Evaluation	2
3.1	Data	2
3.2	Tools	2
3.3	Evaluation Metrics	2
3.3.1	Sharpe Ratio	3
3.3.2	Win Rate	3
4	Methodology	3
4.1	Training/Test Sets Split	3
4.2	Feature Generation/Selection	4
4.3	Random Forest Model	4
4.4	Trading Strategy	4
5	Result*	4
6	Conclusion*	5
7	Future Work	5
8	Appendix*	6

1 Introduction

The finance industry is one of the major driving forces of Hong Kong. Portfolio construction and optimisation has always been a tedious work for discretionary asset managers and individual investors. With the increasing integration and popularity of machine learning techniques in various industries, the finance industry provides an experimental field with great incentives.

To narrow down the scope of research, a simple long-short momentum strategy is used to construct a stock-only portfolio, focusing on the Hong Kong stock market. This project examines the prediction power of random forest model and compare the findings against benchmark (without any machine learning enhancement).

2 Disclaimer

The information presented in this research is not intended as, and shall not be understood as financial advice to enter in any security transactions or to engage in any of the

investment strategies.

3 Data, Tools and Evaluation

3.1 Data

We use data of listed Hong Kong companies from Refinitiv by Blackstone Group and Thomson Reuters.

We obtain all companies that are currently (July 2019) listed or have been listed on the HKEX from Refinitiv from January 2000 to June 2019 to eliminate survivorship bias. We then obtain corresponding daily price/volume data and recompile the index to align different companies.

We report the distribution of companies (raw data) in Figure 1, split by TRBC¹ economic sector.

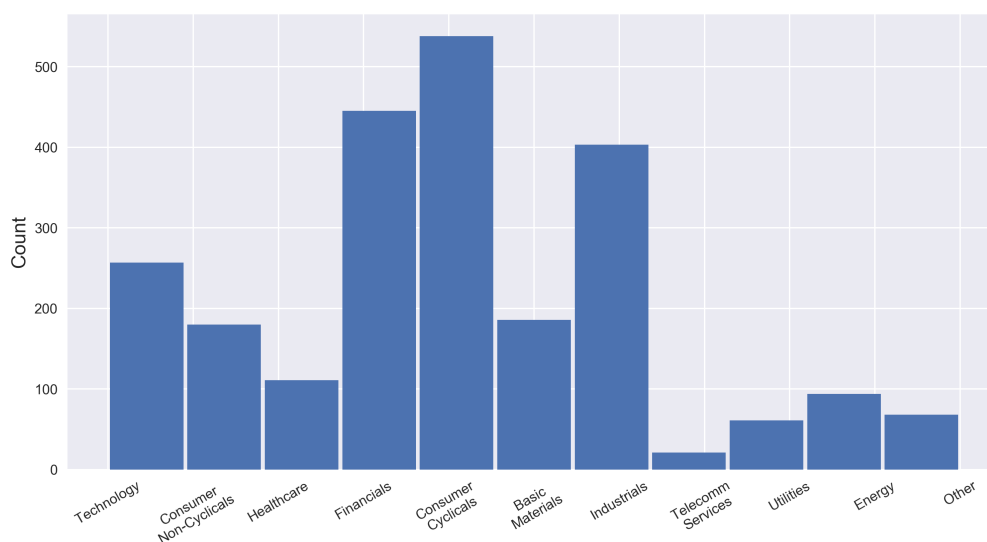


Figure 1: Distribution of TRBC Economic Sector

3.2 Tools

Backtest and data processing is conducted in Python 3.7 with Jupyter kernels using numpy and pandas. We use sci-kit learn for the Random Forest model, trained on CPUs.

3.3 Evaluation Metrics

Sharpe ratio and win rate are the two main evaluation metrics.

¹Thomson Reuters Business Classification System

3.3.1 Sharpe Ratio

Sharpe ratio was first introduced by Sharpe 1966. It measures the expected return gained per unit of risk taken for a zero investment strategy. According to the definition in Sharpe 1994, assume R_{Pt} as a t -period return series, R_{ft} as the risk-free rate series over the same period. Then the Sharpe ratio S_h from $t = 1$ to $t = T$:

$$S_h \equiv \frac{\bar{D}}{\sigma_D}$$

where $D \equiv R_{Pt} - R_{ft}$

$$\bar{D} \equiv \frac{1}{T} \sum_{t=1}^T D_t$$

$$\sigma_D \equiv \sqrt{\frac{\sum_{t=1}^T (D_t - \bar{D})^2}{T - 1}}$$

3.3.2 Win Rate

Win rate is expressed as the ratio of profiting time to the total investment timespan. Let π_t be the profit/loss of a strategy at time t , T be the total number of steps (timespan). Assume the profit/loss is non-zero at every time t , i.e. $n_{\pi=0} = 0$, then $T = n_{\pi<0} + n_{\pi>0}$. Let w be the win rate, pf be the profit factor, pr be the payoff ratio, RoR be the risk of ruin.

$$w \equiv \frac{n_{\pi>0}}{n_{\pi<0} + n_{\pi>0}}$$

4 Methodology

Our methodology consists of four stages. First, we split the data into training and test sets. Second, we discuss the feature space for training and prediction. Third, we describe the random forest model. Fourth, we apply the trading strategy.

4.1 Training/Test Sets Split

We define multiple non-overlapping train-test combination sets to avoid information leak. A train-test combination consists of a training period lasting 750 transaction days (approximately 3 years) and a test period of 250 days (approximately 1 year, Fischer and Krauss 2017). We split the dataset from 2000 to 2015 into 4 non-overlapping study periods.

4.2 Feature Generation/Selection

We currently use "momentum" based features, i.e., we adopt a "lookback" period which l -timesteps of historical price data is reviewed and used for feature computation. We take a simple return of security s during the time interval from $t - l$ to t as the input.

$$r_{t-l,t}^s = \frac{P_t^s}{P_{t-l}^s} - 1$$

Note that the input feature (return) is normalised accross the sector m , i.e.,

$$\tilde{r}_{t-l,t}^s = \frac{r_{t-l,t}^s - \mu_{t-l,t}^m}{\sigma_{t-l,t}^m}$$

We define a 3-class classification problem, the dependent variable $y_{t,t+h}^s|t$ can have 3 different values. We adopt a "holding" period that we review data upto timestep $t + h$ and compute the returns $r_{t,t+h}^s$ and sort the results. Class 1 refers to the securities in the upper third of the ranking. Class -1 and 0 refer to the securities in the lower and middle third respectively. We design the definition to maintain balanced classes.

4.3 Random Forest Model

We use random forest as an overlay of the original momentum signal. The model aims to improve the quality of the raw signals by unveiling hidden patterns accross different securities over the same time period. We set the number of estimators to 100 and use default configuration for other parameters.

4.4 Trading Strategy

We adopt a simple long-short strategy.

First, classify each stock $s_i \in S$ at $t+h$ given the information upto t . Second, compute the prediction confidence to obtain a weight matrix for trade execution. Third, long the top $k\%$ and short the bottom $k\%$ to create a long-short portfolio consisting of $2k\%$ stocks as the top securities are the most undervalued and the bottom ones are the most overvalued, currently $k = 5$.

The same procedures are performed on both the random forest model and raw signals.

5 Result*

The random forest model shows significant Sharpe ratio and win rate improvement in most most cases regardless of the sector and test periods. We report the aggregate performance of Sharpe and win rate in figure 2 and 3 respectively.

The results of different test periods are shown at the Appendix.

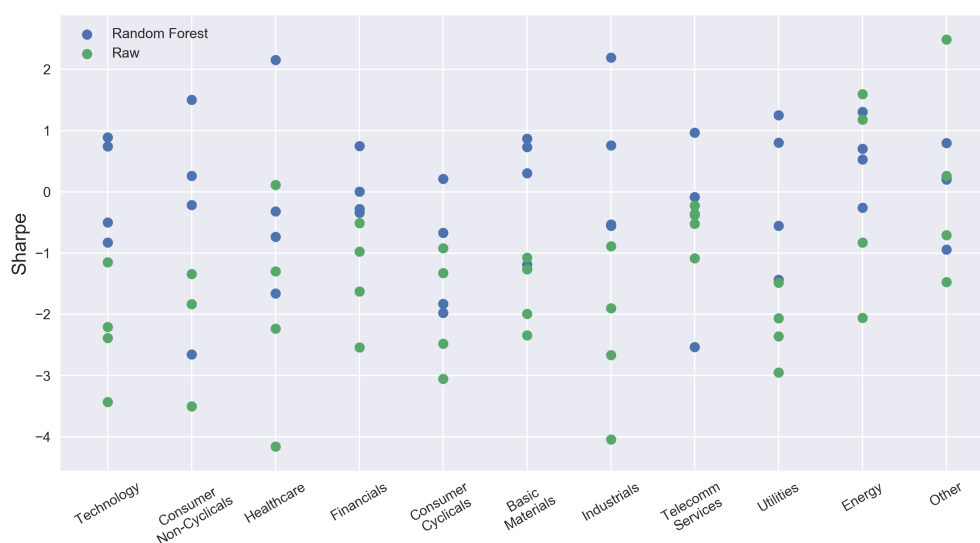


Figure 2: Sharpe Comparison of RF Model and Raw

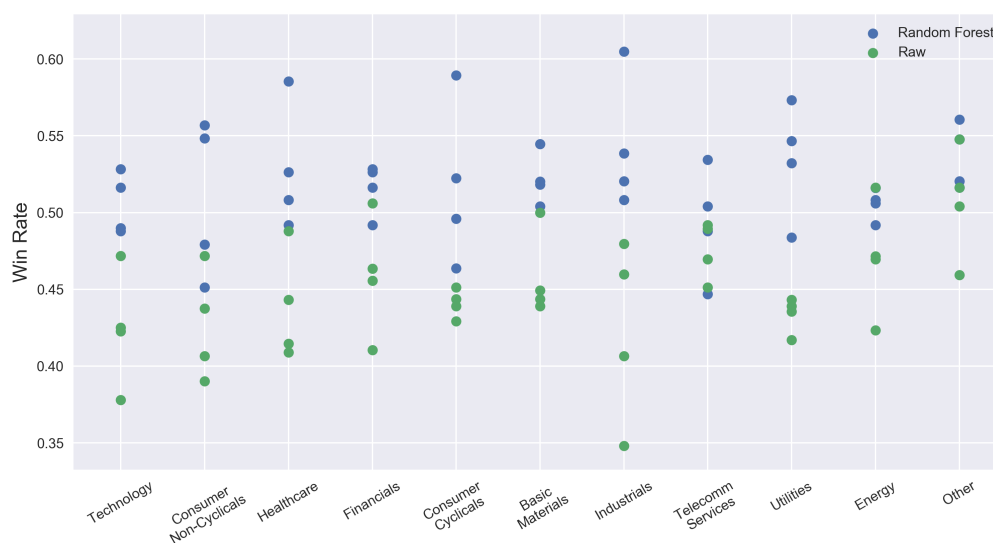


Figure 3: Win Rate Comparison of RF Model and Raw

6 Conclusion*

7 Future Work

We may expand our work to overcome current constraints, including but not limited to data problem, resource limitation.

Due to data limitations, we are restricted to only price/volume data at a daily frequency. The lack of variety and relatively low frequency hinder the testing of more

complex asset pricing models, trading strategies, as well as complex machine learning methods.

8 Appendix*

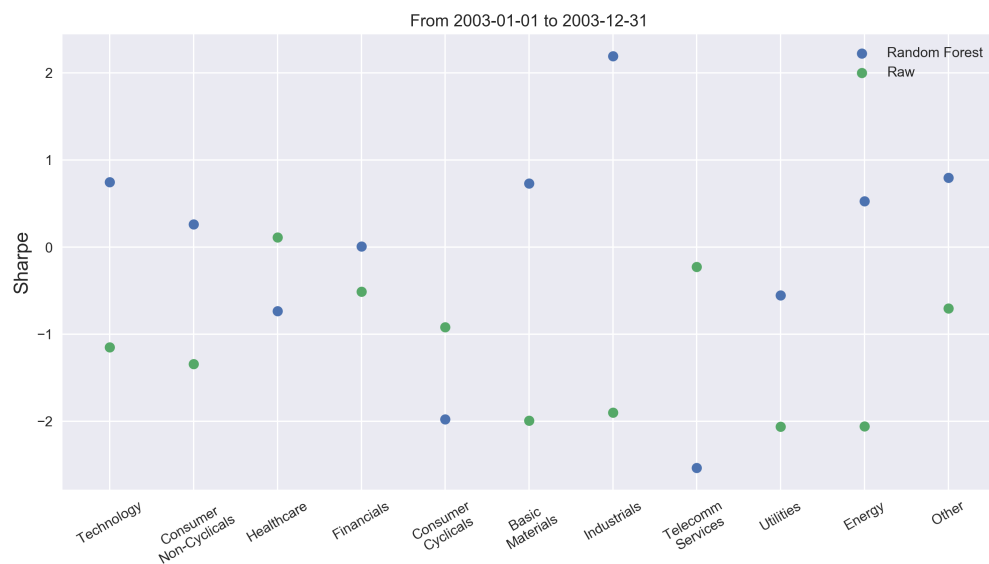


Figure 4: Sharpe Comparison at Test Period 1

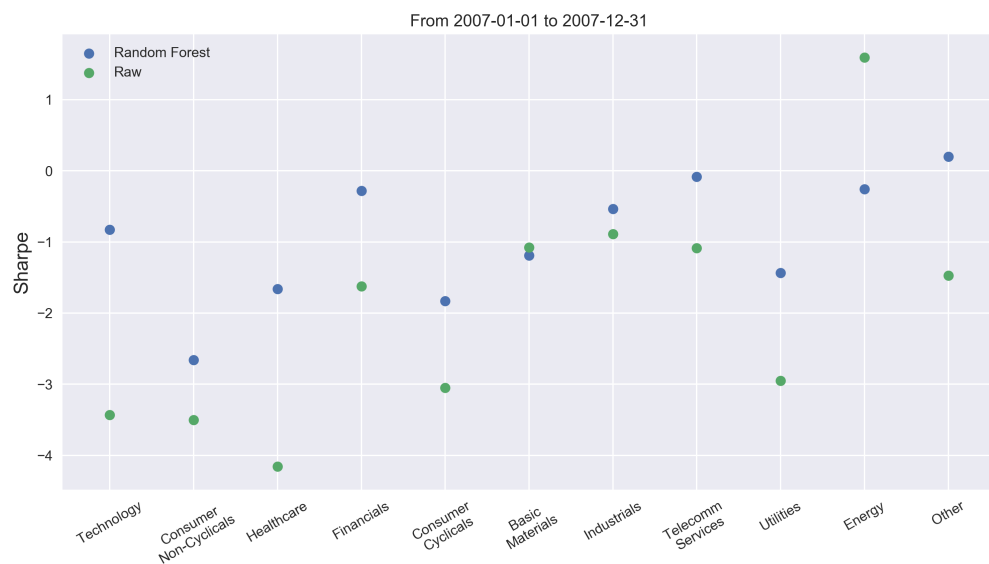


Figure 5: Sharpe Comparison at Test Period 2

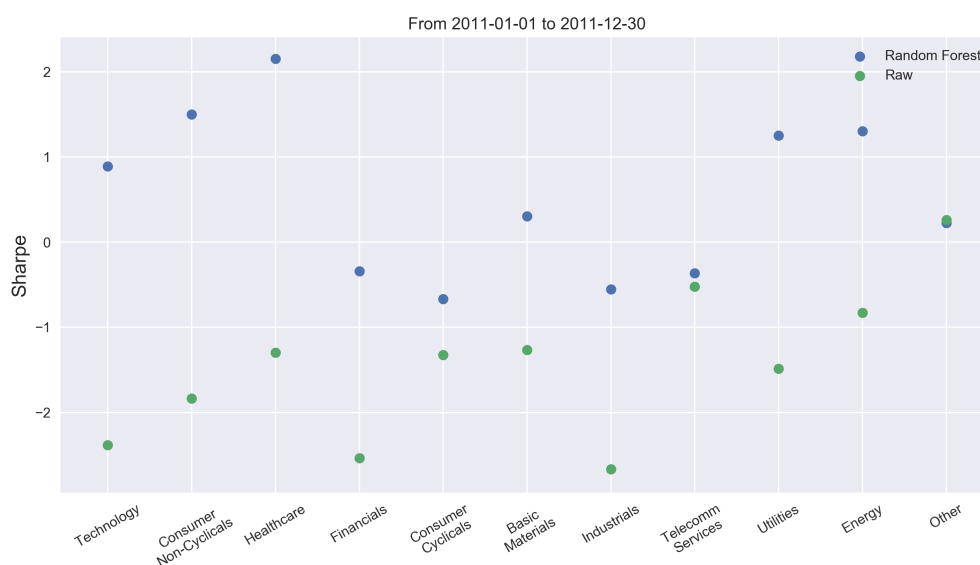


Figure 6: Sharpe Comparison at Test Period 3

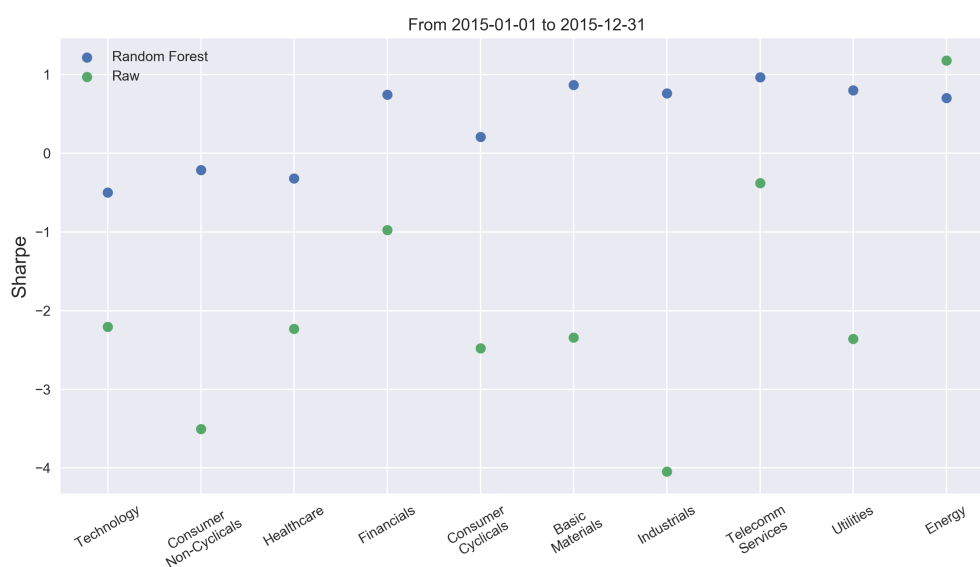


Figure 7: Sharpe Comparison at Test Period 4

Reference

- [1] William F. Sharpe. "Mutual Fund Performance". In: *The Journal of Business* 39.1 (1966), pp. 119–138.
- [2] William F. Sharpe. "The Sharpe Ratio". In: *The Journal of Portfolio Management* 21.1 (1994), pp. 49–58.

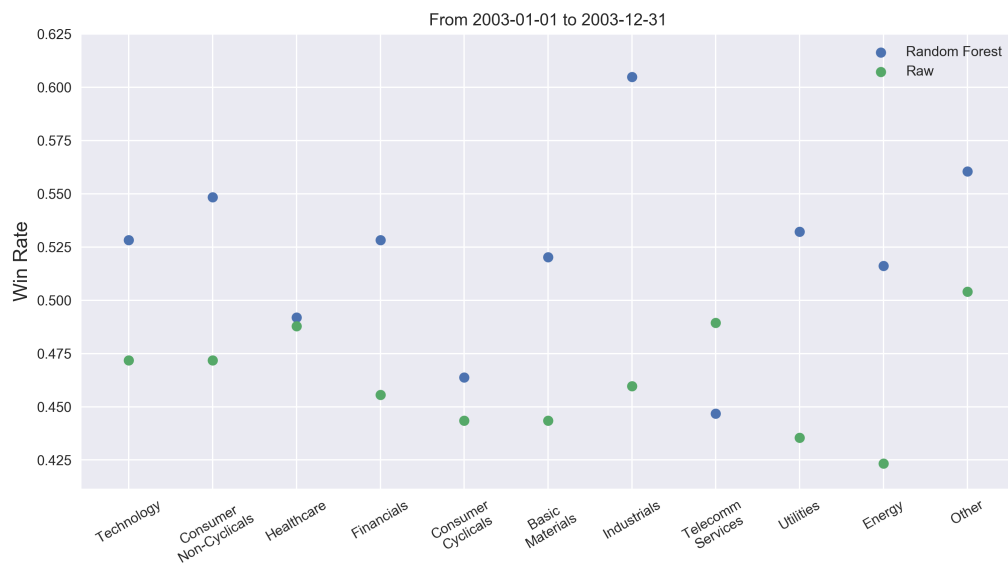


Figure 8: Win Rate Comparison at Test Period 1

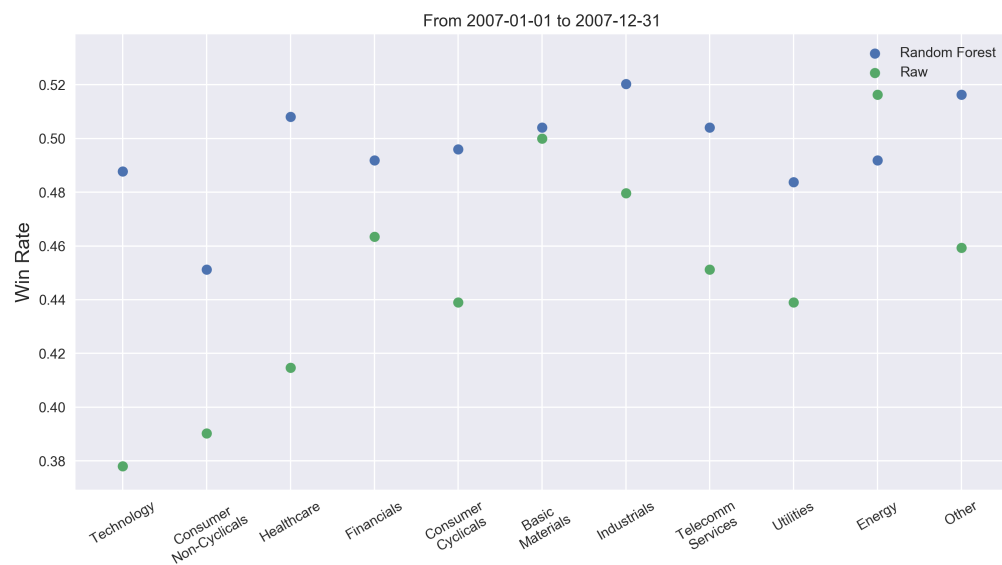


Figure 9: Win Rate Comparison at Test Period 2

- [3] Thomas Fischer and Christopher Krauss. *Deep learning with long short-term memory networks for financial market predictions*. Tech. rep. 11/2017. 2017. URL: <http://hdl.handle.net/10419/157808>.

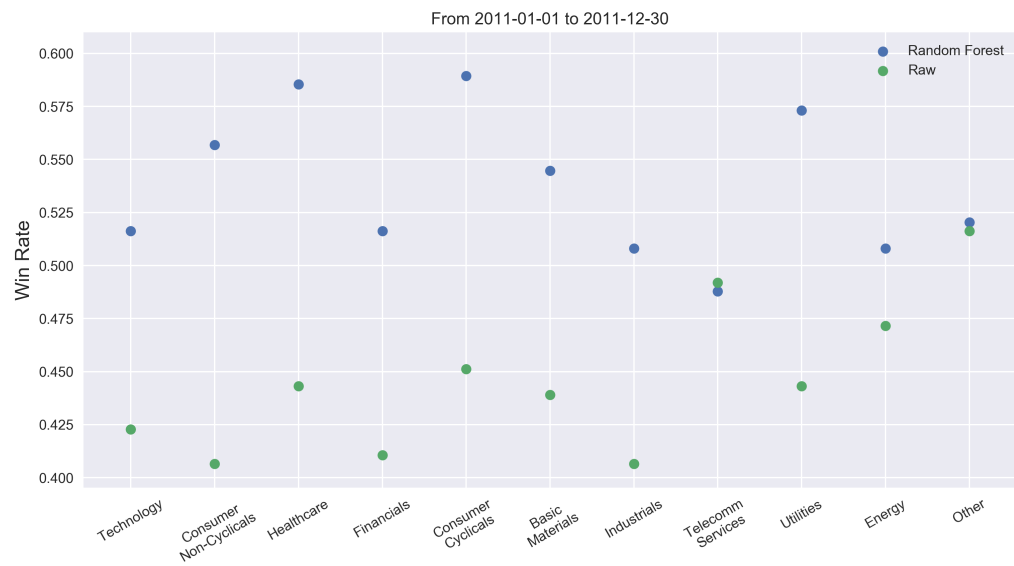


Figure 10: Win Rate Comparison at Test Period 3

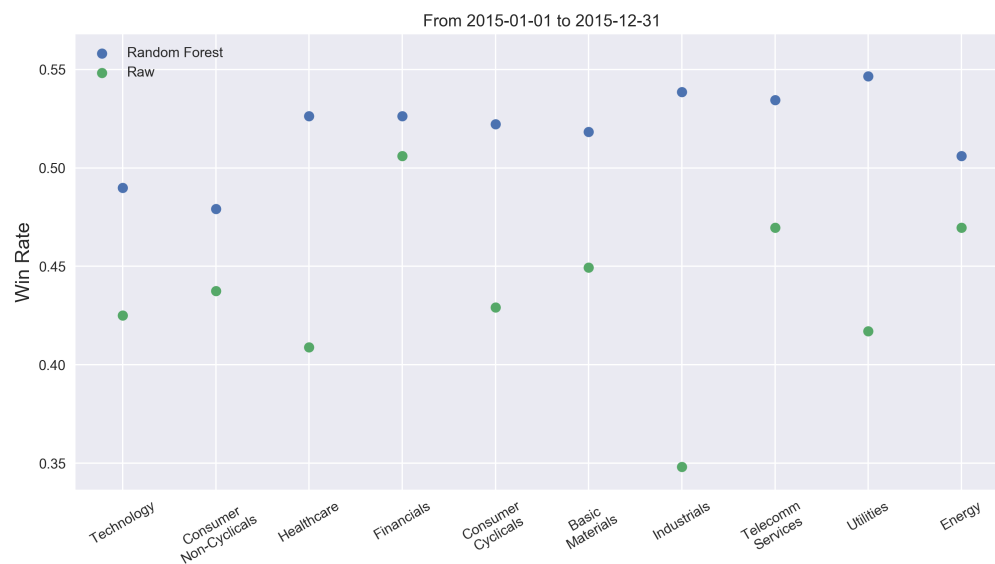


Figure 11: Win Rate Comparison at Test Period 4