Big Data and Finance

Individual Assignment

Kuan-Lun Chiang

01373799

Imperial College London

# I. Introduction

The research of Moskowitz and Grinblatt (1999) suggests that the industry momentum can increase the profitability of stock portfolios. Later, Behr, Guettler, and Truebenbach (2011) integrate the industry momentum with BSV (Brandt, Senta-Clara, & Valkanov, 2009), a parametric portfolio, to demonstrate that the effectiveness of industry momentum to stock portfolios. As a result, the industry momentum could be valuable to the investors and analysts.

Machine learning has been widely used for stock forecasting. Given the remarkable prediction power, machine learning model is expected to have good performance in forecasting industry values. However, not every industry information is valuable when predicting a specific industry, and fitting those noise data into prediction model could potentially hinder the prediction power. Although having the industry domain knowledge allows the analyst to find the critical features, the traditional two-pronged approach is not practical if the critical features cannot be identified (Chinco, Clark-Joseph, & Ye, 2015). Therefore, incorporating the prediction models with feature selection process could be an alternative for choosing relevant features.

In fact, there is a various feature selection models using for stock forecastings such as LASSO, Classification and Regression Tree (CART), and Principal Component Analysis (PCA) (Zhang, et al., 2013; Zhong & Enke, 2016). This report uses 48 industries portfolios to predict six industries portfolio values. There are four machine learning models used for the analysis including LASSO, Random Forest, SVM Regression, and K-NN Regression. Give the properties of those models, LASSO and Random Forest have feature selection processes. Therefore, these two models are further integrating with SVM Regression and K-NN Regression as a part of feature selection.

Even though the Efficient-Market hypothesis assumes no lead-lag effect, Fiedor (2014) examines the asynchronous relationships among financial instruments. This report will observe how the lag term features influence the model performance. This report is organized as in the order of data description, analysis, results and performance, discussion, and conclusion. In section II, this report will present the source of data and discuss some important information for our analysis. Section III demonstrates our interests and methodology of our analysis. Section IV and V show the results of our analysis and lead to relevant questions. Section VI will conclude this report.

# II. Data Description

## Industry Portfolios

There are 48 industry portfolios using for the analysis. These portfolios are constructed by various stock values from multiple sources and organized by the Standard Industry Code (SIC) (French, 2018). Therefore, every industry contains various sub-classes. For instance, the Beer & Liquor industry has sub-classes of beverage, malt beverages, malt, wine, and distilled and blended liquors. These sub-classes could be either supplementary or complementary to each other. Moreover, there could be more than one relationship between two industries. For example, the food product industry could be both complementary and substitution to the soda & candy industry.

## Dataset

The industry portfolio data starts from 1st July 1969 to 31st October 2017. To predict the industry portfolio returns, this analysis selects six industries from the 48 portfolios. Since industries may have several relationships, the analysis examines the average correlations of each industry. In turn, the three industries having the strongest and weakest average correlations are prepared for the prediction. The three industries with the strongest average correlations represent that the industry returns have stronger connections to all the other industry returns. These industries are machinery, wholesale, and business services. In contrast, the three portfolios having the weakest average correlations mean that their industry returns are more independent of the other industry returns. These industries are gold, smoke, and candy & soda.

# III. Analysis

## Motivation

The finding of Patrick, Andre, and Fabian (2011) uses industry momentum to improve the performance of the stock portfolio. Moreover, there are many people that implement machine learning models to predict stock portfolios, and some findings also integrate with dimensionality reductions such as LASSO or PCA (Zhong& Enke, 2016; Zhang, et al., 2013). Inspiring by those researches, we are interested in implementing machine learning models to predict industry portfolio values and examine whether the feature selection techniques can improve the prediction performance.

## Methodology

The prediction modeling consists of preparing the dataset for analysis, choosing model via time series cross-validation, and computing test error using test data. For the data preparation, the response variable for the analysis is the return values of the industry, and the predictors include the lag-values of the industry itself and the present-value of the other 47 industries.

As the research of Fiedor (2014) validates the lead-lag effect in the financial market, the lag-lead effect is likely to influence the stock market. Consequently, using different lag-terms may potentially improve the forecasting ability; our analysis examines three lag-terms, which are one day, five days, and ten days. After the data is prepared for modeling, the data is split into training and test sets.

The training data is used for model selection via time series cross-validation, while the test data is held out from the model selection process and provide unbiased scores. The time series cross-validation is compute via fixed window rolling horizon. The model will be fit with the data number of fixed window every time and compute the cross-validation error by using the validate data which is the next data point. Likewise, the test error is also computed by rolling horizon method with fixed window. In addition to those machine learning model based on distances such as LASSO, K-NN, and SVM, we normalize the training data by z-score transformation and use training data’s mean and standard deviation to normalize validate data.

During the model selection process, the window size and model parameter is the major hyperparameters that must be determined. The selection criteria are which window size and model parameter gives the lowest cross-validation error.

## Sparsity

Since there might be only some important predictors at one point in time, betting on sparsity could be useful when predicting the big and complex financial market (Chinco, Clark-Joseph, & Ye, 2015). Also, feature selection has been widely used in stock prediction, and there are many different selection models that have incorporated with different machine learning models (Zhang, et al., 2014). Our analysis will incorporate the prediction models with feature selection processes.

## Machine Learning Model

### Least Absolute Shrinkage and Selection Operator (LASSO).

LASSO is a linear regression model with first-norm penalties. Since the first-norm penalty will shrink some coefficients into zero, this linear model has the feature selection process, and it will filter out the irrelevant predictors during predictions. In our analysis, we will examine the performance of this model against the benchmark model and incorporate with SVM and K-NN Regression models.

### Random Forest.

Random Forest is an ensemble learning method which constructs multiple decision tree models while limiting the maximum features. In our analysis, this model will also be compared with benchmark model and incorporated its important features with SVM and K-NN regression. Since it limits the maximum features using in each decision trees, the feature selection process occurs when building trees. The feature importance is the frequency of the predictor used in the Random Forest model. Hence, we can set a threshold on the feature importance to select the important features for other models.

### Support Vector Machine (SVM) Regression.

SVM is a non-probabilistic machine learning model. It uses a hyperplane to separate different classes, and it usually incorporates with kernel trick. Kernel trick is a technique that projects the data into a higher dimension, and therefore help the SVM to separate the different classes. Given the kernel trick, SVM can fit data non-linearly. To separate those unrepeatable classes, there is a soft margin to allow the misclassification. The SVM Regression is to minimize the cost function and therefore find the coefficients. In this analysis, SVM Regression is using Gaussian kernel for predictions.

### K-Nearest Neighbor (K-NN) Regression.

K-NN is a probabilistic model based on the calculating the distances between each data point. The concept of K-NN is to determine the K nearest data point and classify the new data point according to the majority number of the nearest neighbor. Hence, the hyperparameter K needs to be determined via cross-validation. The K-NN Regression is based on computing the average of K nearest data point and fit into regression for prediction.

### Benchmark Model

The benchmark model is linear regression model with ordinary least square (OLS) estimator. The concept of linear regression is to find the linear relationship between predictors and response variable. To compare with the forecasting models, the window size of the benchmark model is in consist with the forecasting model. In other words, we use the window size of the forecasting model to compute the test error of the benchmark model. In turn, the test error of benchmark model is used for further evaluation.

# IV. Results and Performances

## Metrics

To measure the performance, there are three major metrics using in this analysis. These metrics are root-mean-square error (RMSE), cumulative sum of square error difference (ΔSSE), and out-of-sample R square. RMSE is the square root of the sum of square error divided by the number of test sample. It can be written as:

The cumulative sum of square error difference is the differences between the forecasting models’ cumulative square error and the benchmark model’s cumulative square error. Hence, if ΔSSE is positive, the benchmark model has better performance. Similarly, if ΔSSE is negative, the forecasting model has smaller cumulative test error. Hence, the negative number is preferable to the machine learning model. ΔSSE can be written as:

To compute the out-of-sample R square, we use one to minus the foresting model’s cumulative square error divided by the benchmark model’s cumulative square error square. Given this metric, the forecasting model has better performance if the out-of-sample square is positive. Likewise, if this metric has a negative number, the benchmark model has better performance. This metric also indicates how well the predictors explain the response variable. The out-of-sample R square can be written as:

## Test Results

### Model performance.

The result of test data indicates that the Random Forest outperform LASSO, SVM Regression and K-NN Regression (with or without feature selections) in most of the industries and different lag-terms. Apart from Random Forest, K-NN Regression without feature selection has good RMSE scores. The average RMSE scores of Random Forest and K-NN Regression across six industries are around 0.96 and 1.05, respectively.

Although the RMSE scores of SVM Regression did not perform well than another forecasting model, its performances are improved by feature selection process, especially incorporating with Random Forest. The improvement of SVM Regression’s forecast errors improves by approximately 0.03 with Random Forest selection. Although K-NN Regression with Random Forest selection has decent performances, there is no significant improvement by using Random Forest or Lasso selections.

By observing the ΔSSE and , Random Forest, LASSO, and K-NN family almost always beat the benchmark model and explain the industry portfolio values better. However, the performances of SVM family are significantly influence by different industries and lag-terms. Nonetheless, SVM family usually have greater ΔSSE and low in the beginning, and it will start to decrease ΔSSE the and improve after 500 days.

### Window size and feature selection.

In the analysis, there are five different window sizes tested, which are 20, 60, 120, 180, and 360 days. The results of every industry and model are in consist with choosing 360 days except for gold and smoke industries. These two industries are varied between 180 and 360 with different models.

For the feature selection, we found that the LASSO can find some relative long-live important features with direct relationships to machinery, wholesale, business services, and gold industries. For machinery industry, oil, coal, and steel industries are selected, while shipping industry is identified for business services and wholesale industries. For gold industry, LASSO chooses mines, and coal industries. These identified features are either the supply or the demand related to the target industry.

Like LASSO selection, Random Forest will typically have some long-live industries with immediate relations. Besides the long-lived features, Random Forest will choose several short-lived industries with or without direct relations to the target industry.

### Industry performance and lag-structure.

For business services, machinery, and wholesale, Random Forest model has the best performances, and K-NN Regressions with lag-term 1 and 10 have the best performances for smoke and soda industries. For gold industry, LASSO has the smallest forecast errors. Machinery, business services, and wholesale industries are relative stable to different lag-terms. Most of the machine learning algorithm has similar performances for these industries except for 5-day-lag-term in SVM with Random Forest selection process. SVM with Random Forest selection has exceptional high forest error with 5-day-lag-term across all industries.

The model performances in soda, smoke, and gold industries have greater variation corresponding to different lag-terms. For these three industries, there is no lag-term always having better results. Nevertheless, one day and ten days lag-terms usually have smaller forecast errors in smoke and soda industries. For gold industry, ten day-lag-term is usually preferable for most of the algorithms.

# V. Discussion

After examining the results of the analysis, there are three questions needed to be discussed. (1). Since machine learning algorithms can perform the non-linear transformation, do those models have better predictability? (2). Although our feature selection processes identify some features might be closely related to the response variable, these selections do not have significant improvements to the forecasting model. Is there any potential reason and resolutions? (3). In our analysis, some lag-terms are preferable than other. Is there any economical reason causing the differences? How does the lead-lag effect influence the industry portfolio forecast?

## Machine Learning Algorithm

In our analysis, there are four machine learning algorithms compared with linear regression. Except for LASSO, these algorithms have a non-linear transformation. Although the linear model might not be as sophisticated as non-linear models, LASSO demonstrates strong and stable forecast in our analysis. On the other hand, the SVM Regression family with Gaussian kernel has relative weak performances in our analysis comparing to other machine learning models. Nevertheless, Random Forest and K-NN Regression have shown remarkable predictability. Hence, there is no guarantee that non-linear models have better performances.

However, we found that the forecast errors of LASSO and linear regression are stable even if the lag-term is different. Given the non-linear transformation, machine learning models could be more sensitive to the training data. Thus, the irrelevant features or outliers could potentially decrease the prediction powers more than the linear model.

## Feature Selection

Feature selection allows us to identify the relevant predictors in the quantitative sense; however, these predictors do not necessarily have economical relations. In our analysis, LASSO and Random Forest successfully choose some features that could be closely related to the response variable. However, the performances of implementing these feature selection processes do not always improve the model performances.

There are some potential reasons that influence the improvement of feature selections. Even though feature selection process will choose the important predictors quantitatively, these features might not be able to explain the causality effect. Since the correlation between industry portfolios only represents the movement of two portfolio values, it does not explain what causes the raise or fall in portfolio values. In fact, LASSO and Random Forest do not examine the causality effect during the selection process (Zhang, et al., 2014). Research of casual feature selection algorithm has shown that correctly identifying the casual effect for forecasting stock values among features can have better performances than LASSO and classification tree (Zhang, et al., 2014).

The other potential reason could be that there are much noises existing in the data. Since machine learning algorithms can fit training data non-linearly, that noise would hinder the overall forecast power. Although LASSO and Random Forest use feature selection to achieve dimensionality reduction, the noises could be existing in those selected features. The research of Tsia and Hsiao (2010) combines multiple feature selection algorithms and achieve better results. By controlling the explained variation of Principal Component Analysis (PCA), the noise information could be reduced, thereby improving the quality of selected features.

## Lag Structure

Although different industries and models could have different preferences for the lag-terms, 1-day lag-term and 10-day lag-term are generally have better results than 5-day lag-term. These facts could be explained by the speed of information diffusion within the industry. Investors will react to the news; consequently, the stock prices will be influenced (Lo & MacKinlay, 1990; Mcueen et al., 1996). As Hou (2007) argued, the lead-lag effect is caused by the slow diffusion of bad news. As a result, the lag-terms have different performances.

Although our analysis could not validate information diffusion assumption, the asynchronous relationships among the industry portfolios have different forecast performances. However, the 5-day lag-term is generally not favorable for all industries and models. This result could be explained by the delay of information and investor reactions. Nevertheless, our lag-term only controls the industry itself. Our analysis does not represent the asynchronous relationships between different industries. To further examine the lead-lag effect, the model should control more lag-terms and examine their statistical significance. However, that is beyond the scope of this report. The detailed analysis of lead-lag effect could be a topic for future research.

# VI. Conclusion

This report examines the performances of four different machine learning models for six different industry portfolios by controlling its lag-term and the presence values of other 47 industry portfolio values. The results of the analysis indicate that Random Forest has the best predictability than the other three machine learning models. Besides the overall model performances, we found that the forecasting power of SVM improved by incorporating feature selection process. Although the improvement of feature selection process is not significant, we discuss two potential resolutions. In the discussion, we also discuss the potential reasons how lag-terms will influence the prediction.

# References

Andrew, L. W. (1990). When are contrarian profits due to stock market overreaction. *Oxford Journal*, 175-205.

Brandt, M. W., Senta-Clara, P., & Valkanov, R. (2009). Parametric portfolio policies: exploiting characteristics in the cross-section of equity returns. *The Society for Financial Studies*, 3411-3447.

Chinco, A., Clark-Joseph, A., & Ye, M. (2015). *Sparse signals in the cross-section of returns.* SSRN.

Fiedor, P. (2014). Information-theoretic appraoch to lead-lag effect on financial markets. *The European Physical Journal B*, 87-168.

French, K. (2018). *Detail for 48 industry portfolios*. Retrieved from http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/Data\_Library/det\_48\_ind\_port.html

Hou, K. (2007). *Industry information diffusion and the lead-lag effect in stock returns.* Oxford University Press.

Mcqueen, G., Pinegar, M., & Thorley, S. (1996). Delayed reaction to good news and the cross-autocorrelation of portfolio returns. *The Journal of Finance*, 889-919.

Moskowitz, T., & Grinblatt, M. (1999). Do industries explain momentum. *The Journal of Finance*, 1249-1290.

Tsai, C.-F., & Hsiao, Y.-C. (2010). Combining multiple feature selection methods for stock prediciton: union, intersection, and multi-intersection approaches. *Decision Support System*, 258-269.

Zhang, X., Hu, Y., Xie, K., Wang, S., Ngai, E., & Liu, M. (2014). A causal feature selection algorithm for stock prediction modeling. *Neurocomputing*, 48-59.

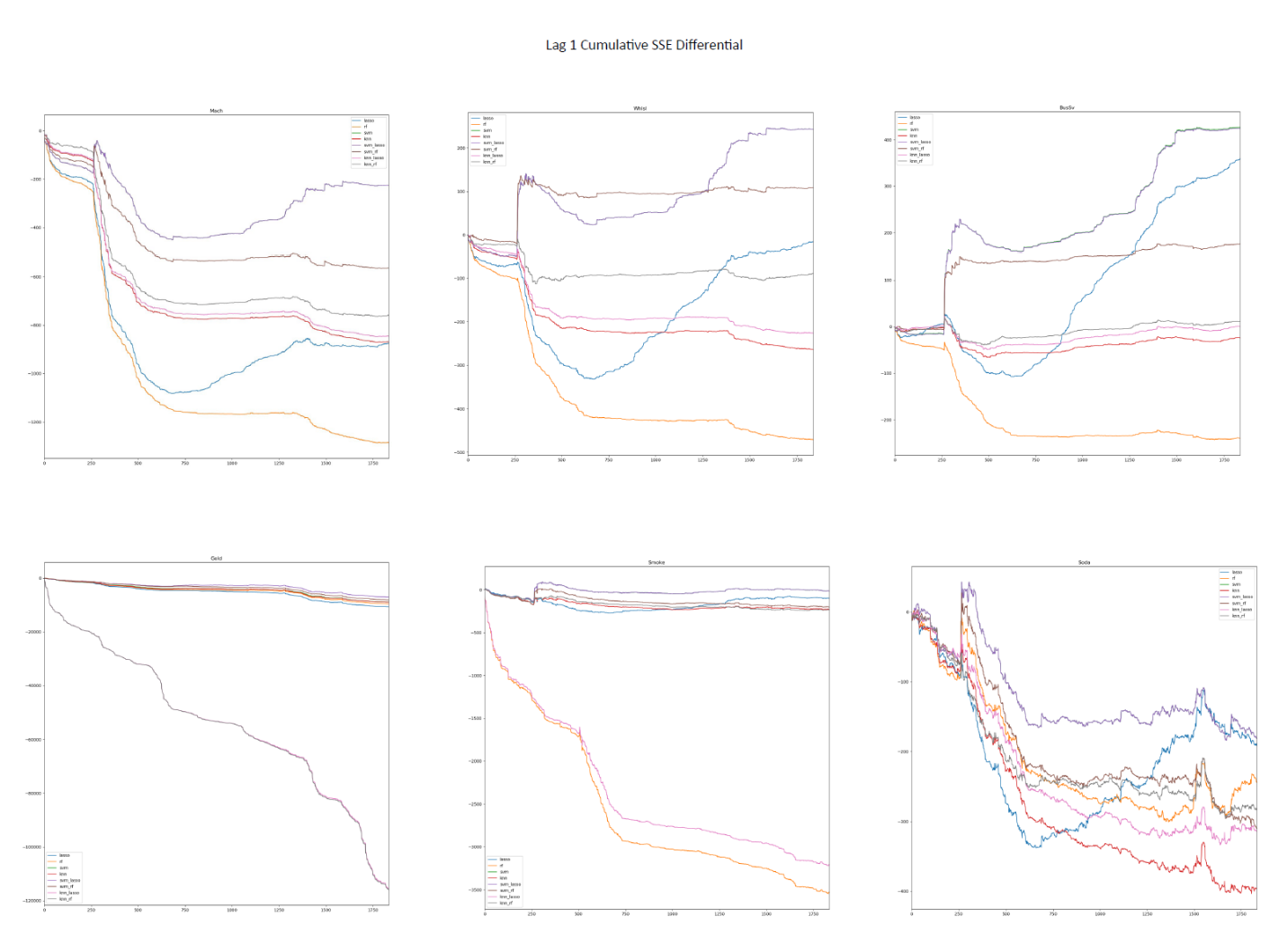
Zhong, X., & Enke, D. (2016). Forecasting daily stock market return using dimensionality reduction. *Expert Systems with Application*, 126-139.

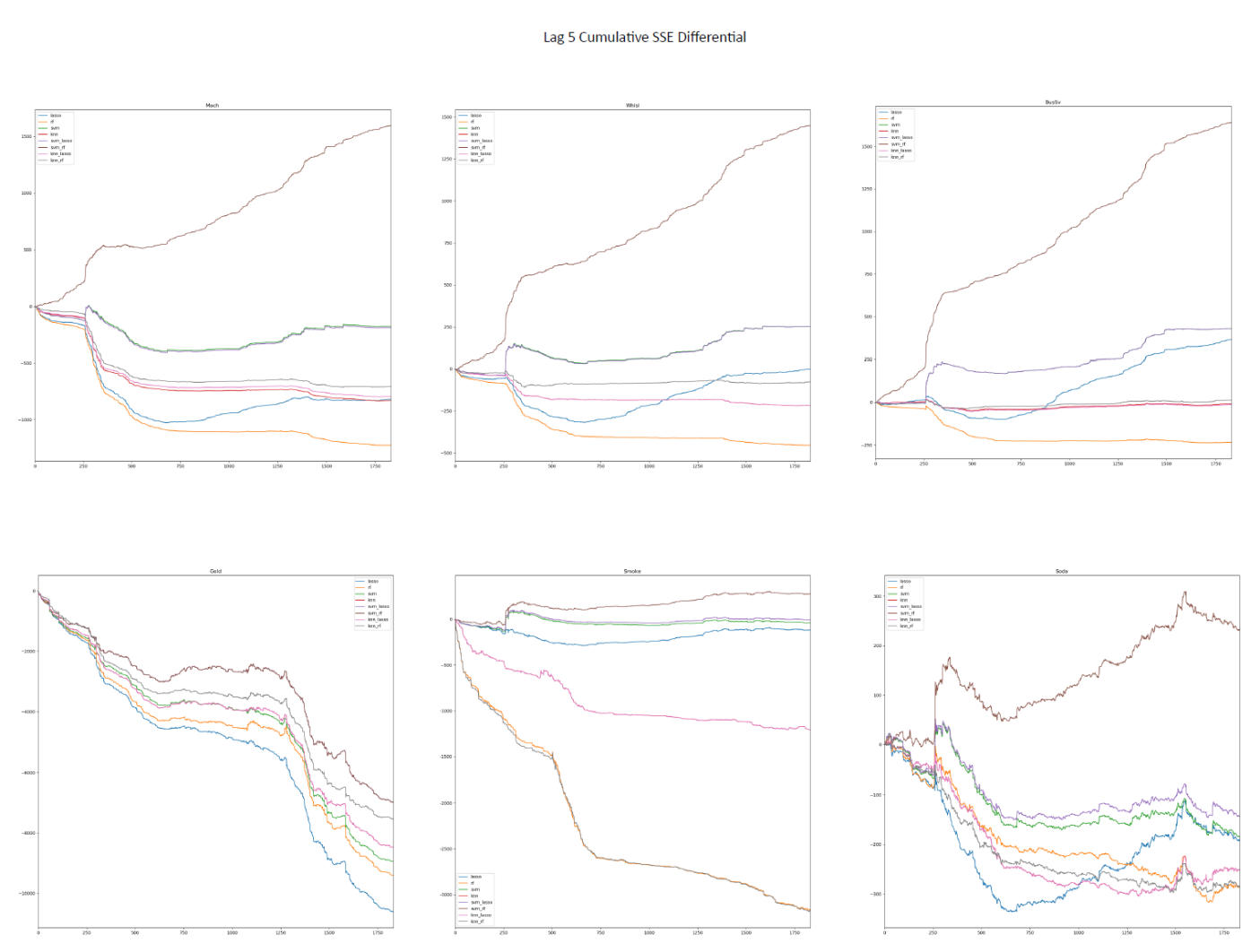
# Appendix

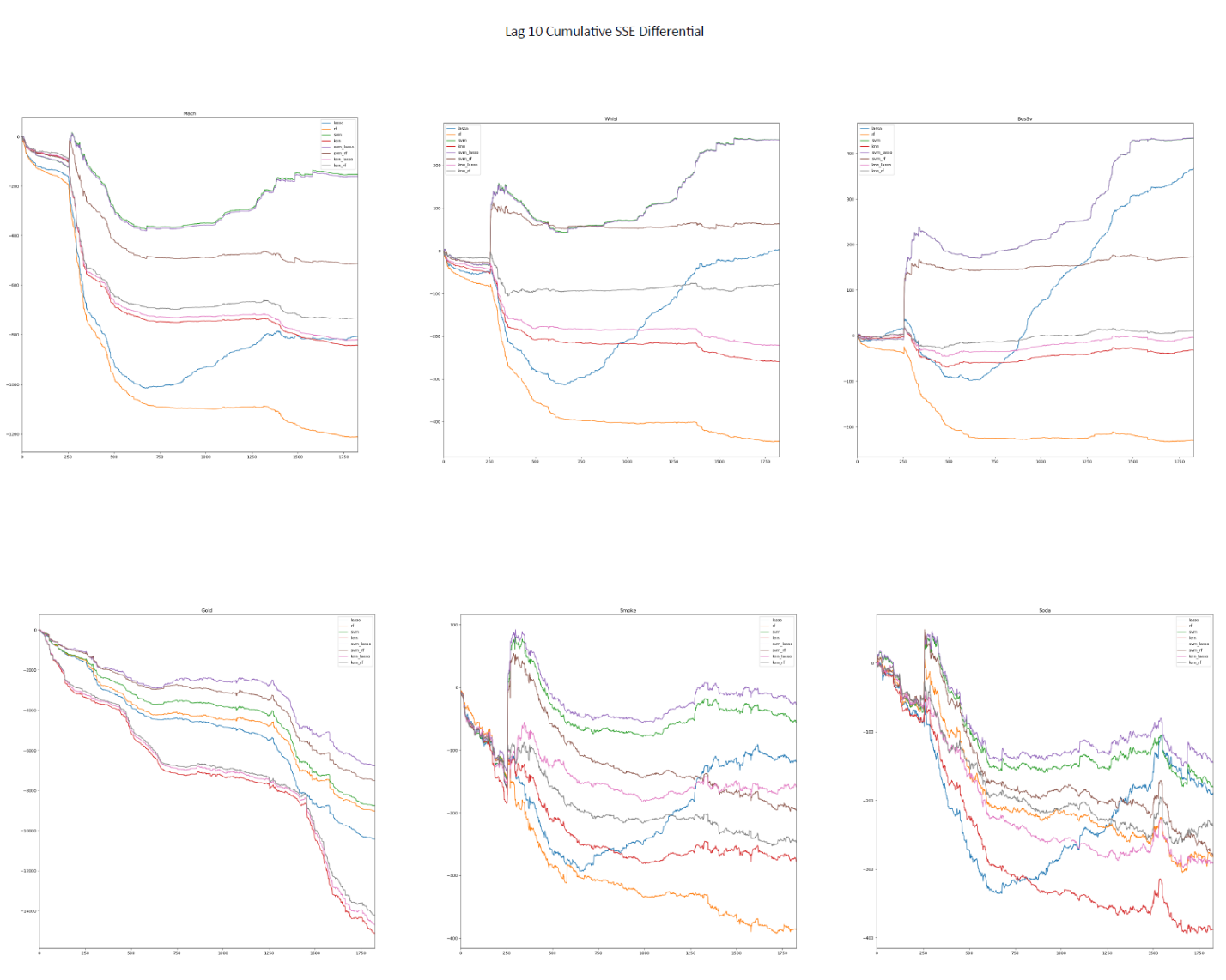
## Model Performance

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | BusSv | Gold | Mach | Smoke | Soda | Whlsl |
|  | Lasso | lag 1 | 0.6660 | 2.3241 | 0.6264 | 1.0519 | 1.1740 | 0.6276 |
|  | lag 5 | 0.6662 | 2.3262 | 0.6268 | 1.0492 | 1.1733 | 0.6277 |
|  | lag10 | 0.6668 | 2.3267 | 0.6272 | 1.0494 | 1.1742 | 0.6282 |
|  | Random Forest | lag 1 | 0.3424 | 2.4503 | 0.4109 | 1.0674 | 1.1611 | 0.3820 |
|  | lag 5 | 0.3405 | 2.4638 | 0.4133 | 1.1113 | 1.1508 | 0.3831 |
|  | lag10 | 0.3430 | 2.4859 | 0.4148 | 0.9772 | 1.1521 | 0.3858 |
|  | SVM Regression | lag 1 | 0.6930 | 2.5256 | 0.8639 | 1.0736 | 1.1757 | 0.7312 |
|  | lag 5 | 0.6915 | 2.5144 | 0.8627 | 1.0690 | 1.1745 | 0.7296 |
|  | lag10 | 0.6933 | 2.5159 | 0.8667 | 1.0655 | 1.1761 | 0.7316 |
|  | K-NN Regression | lag 1 | 0.4844 | 2.5177 | 0.6288 | 1.0179 | 1.1249 | 0.5089 |
|  | lag 5 | 0.4897 | 2.5662 | 0.6206 | 1.1130 | 1.1591 | 0.5250 |
|  | lag10 | 0.4754 | 2.4734 | 0.6107 | 1.0073 | 1.1268 | 0.5008 |
| SVM Regression | Lasso | lag 1 | 0.6921 | 2.7170 | 0.8639 | 1.0736 | 1.1757 | 0.7312 |
| lag 5 | 0.6916 | 2.7195 | 0.8591 | 1.0774 | 1.1841 | 0.7294 |
| lag10 | 0.6935 | 2.7220 | 0.8641 | 1.0729 | 1.1845 | 0.7314 |
| Random Forest | lag 1 | 0.5863 | 2.6120 | 0.7493 | 1.0248 | 1.1455 | 0.6794 |
| lag 5 | 1.0679 | 2.7190 | 1.3080 | 1.1469 | 1.2681 | 1.0889 |
| lag10 | 0.5817 | 2.6480 | 0.7441 | 1.0289 | 1.1529 | 0.6539 |
| K-NN Regression | Lasso | lag 1 | 0.6921 | 2.7170 | 0.8639 | 1.0736 | 1.1757 | 0.7312 |
| lag 5 | 0.6916 | 2.7195 | 0.8591 | 1.0774 | 1.1841 | 0.7294 |
| lag10 | 0.6935 | 2.7220 | 0.8641 | 1.0729 | 1.1845 | 0.7314 |
| Random Forest | lag 1 | 0.4978 | 2.5764 | 0.6391 | 1.1389 | 1.1445 | 0.5285 |
| lag 5 | 0.4867 | 2.5662 | 0.6370 | 1.1130 | 1.1591 | 0.5250 |
| lag10 | 0.4912 | 2.5185 | 0.6204 | 1.0389 | 1.1502 | 0.5213 |

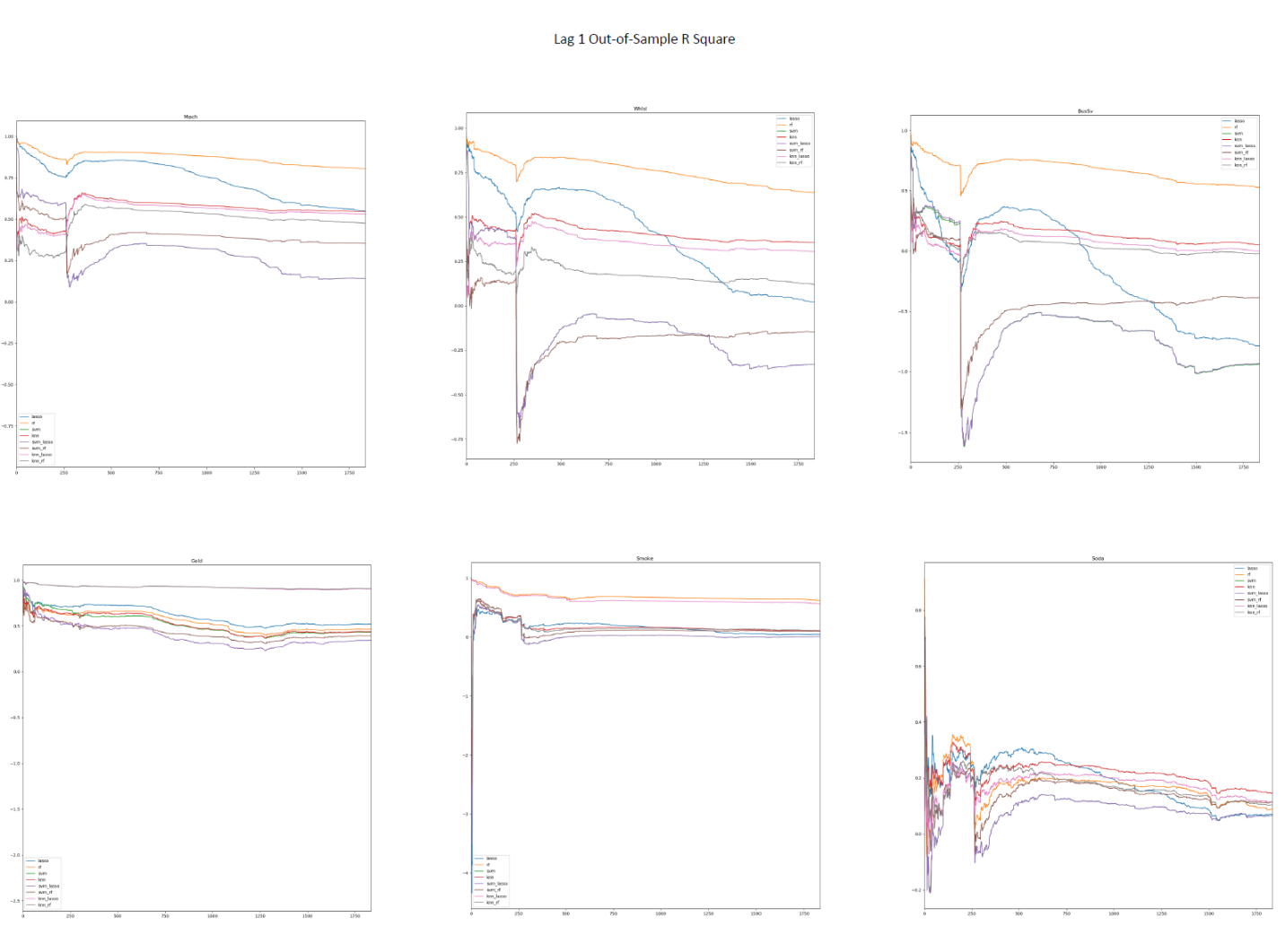
## SSE Differential by different lag terms

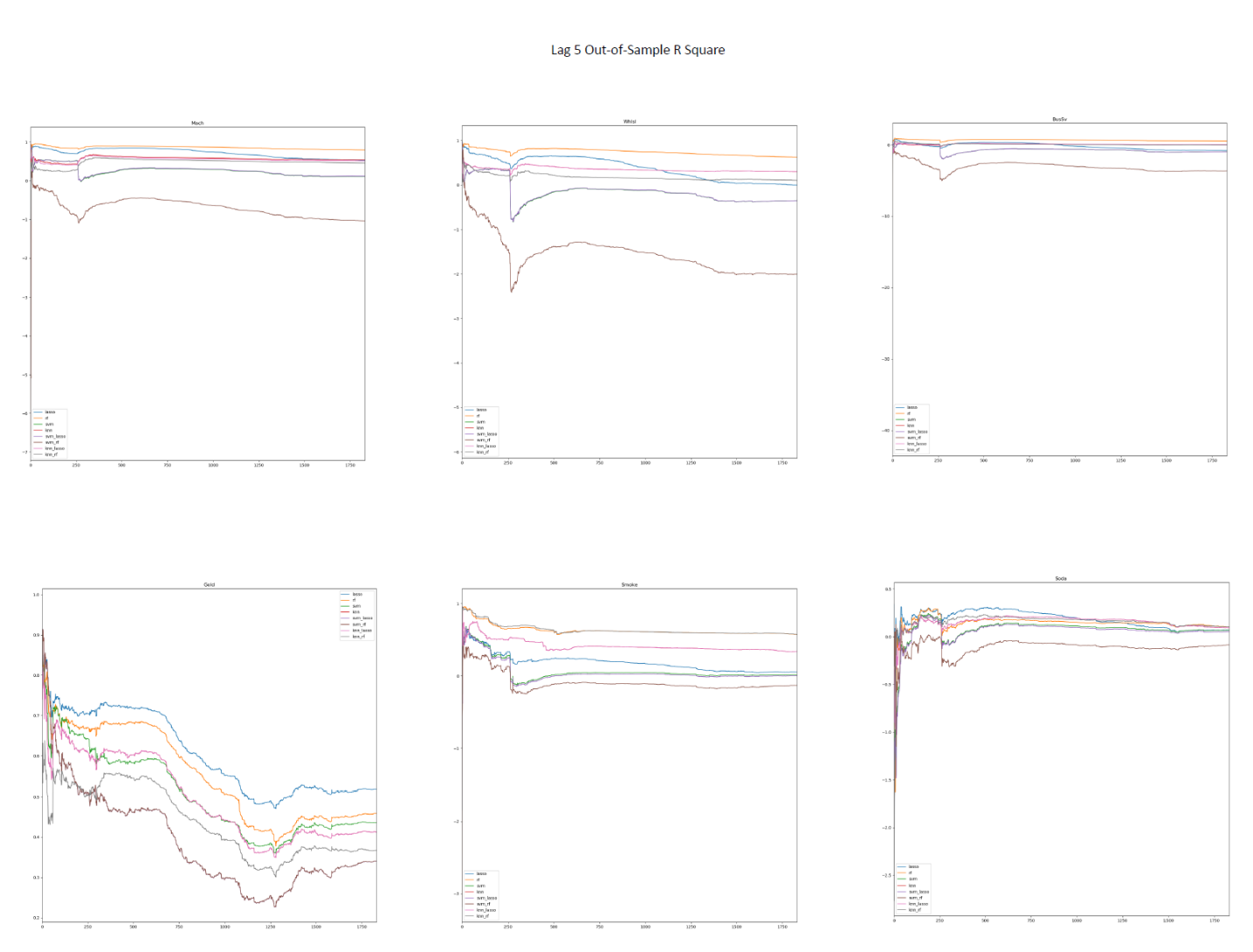


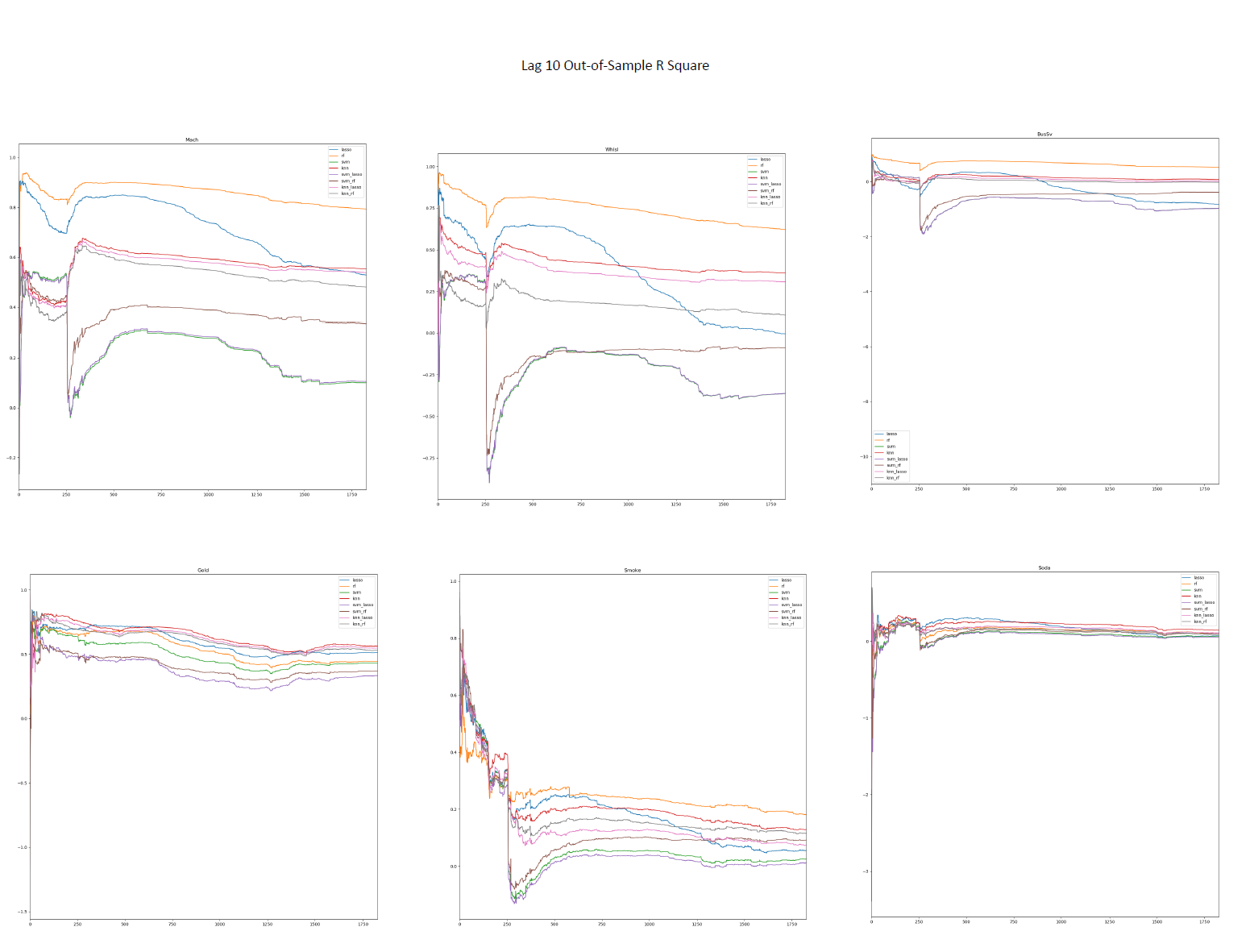




## Out-of-Sample R Square by different lag-terms







## Feature Selection







