

The predictability of cross-sectional returns in high frequency

Yifan Wang[†]

Abstract Stock return forecast is of great importance to trading, hedging, and portfolio management. In this article, we apply LASSO and random forest to make rolling one-minute-ahead return forecasts of Dow Jones stocks, using the cross-section of lagged returns of S&P 500 components as candidate predictors. Although the number of candidate variables is large, the negative out-of-sample R^2 suggests that the predictions from LASSO and random forest give larger mean-squared error than the historical average. So, there is no evidence of predictability in the cross-sectional returns of large stocks in high frequency. The predictability presented by [Chinco et al. \(2019\)](#) might be due to the interaction between large and small stocks.

Keywords: Machine learning; Forecasting returns.

JEL Code: G12, G17.

1. Introduction

Stock return forecast is important in the financial world, and people started proposing forecasting methods very early. For example, [Dow \(1920\)](#) examined the role of dividend ratios in stock return prediction. [Graham et al. \(1934\)](#) argued that a high valuation ratio could indicate an undervalued stock. However, making forecasts is not an easy task, due to complex and fast-changing markets, and the many noise variables involved. A successful forecast always requires three things: a set of useful predictors, a reasonable statistical model, and a good estimation of the model.

A variety of factors are used to predict stock returns. Many are company characteristics, such as dividends ([Ball, 1978](#)), earnings ([Campbell and Shiller, 1988](#)), and book value ([Kothari and Shanken, 1997](#)). Some are macro-economic signals, such as inflation ([Lintner, 1975](#)) and aggregate consumption ([Lettau and Ludvigson, 2001](#)). These data are usually from companies' financial statements and government reports, so the prediction frequency is low (often annually or quarterly). Given that factors used in low-frequency predictions are typically not available with a finer time resolution, high-frequency forecasts are more challenging, and evidence of stock return predictability at high frequency is scarce ([Herwartz, 2017](#)).

Submitted on July 18, 2021. Revised on October 21, 2021. Accepted on October 21, 2021. Published online in March 2022. Editor in charge: Marcelo Fernandes.

[†]Princeton University, USA: yw18@alumni.princeton.edu

However, with the rapid growth of the Internet and computing technologies, trading frequency has increased to fractions of seconds (Aldridge and Krawciw, 2017). In high frequency, we can approach the stock return forecast problem using different signals and methods. Some use self-lagged returns to construct time-series models such as ARMA or GARCH (Herwartz, 2017), while others make use of micro-structure data to forecast returns (Huang and Stoll, 1994).

Chinco et al. (2019) propose a method to make one-minute-ahead return forecasts. They argue that since many variables are short-lived in high-frequency prediction, it is challenging both to identify the useful predictors and to estimate the model using traditional regression. They propose using LASSO to identify the “unexpected short-lived and sparse” signals. Since LASSO can select variables effectively, they include all cross-sectional returns of NYSE-listed companies (over 2000), and their three lags—more than 6000 variables, in total—as candidate predictors, then train LASSO on a 30-minute rolling basis. The results are amazing: using predictions from LASSO along with predictions from a traditional AR(3) model can increase the out-of-sample fit by more than 2%, which is a huge gain. This indicates high predictability and much valuable information in cross-sectional returns.

Inspired mainly by Chinco et al. (2019), in this article we use the cross-sectional returns of a full set of S&P 500 components to make one-minute-ahead return forecasts of stocks on the Dow Jones index. The S&P 500 are the five hundred largest stocks listed in exchanges in the United States, accounting for more than 60% of market capitalization (cap) of all stocks. The Dow Jones index, on the other hand, contains the thirty large companies whose market caps account for more than 15% of the entire market. Unlike Chinco et al. (2019), we focus on large stocks because they are the most liquid ones on the market, and account for a major portion of total trading volume. Large stocks are definitely more challenging to predict. Focusing on large stocks also allows us to examine the source of the predictability presented by those authors further, by comparing our results with theirs.

Beyond LASSO, we also extend the work of Chinco et al. (2019) to non-linear models. As those authors do, we use LASSO to reduce dimensions effectively. Because LASSO is a linear model, and non-linearity is essential in asset pricing (Freyberger et al., 2020), we introduce non-linearity into our predictions. We choose to use random forest (RF) because it is relatively efficient to train, compared to more complicated models such as neural networks. Trees have also already been proven to perform well in forecasting tasks (Gu et al., 2018). We expect that non-linearity can help improve prediction per-

formance.

We train the models on a rolling basis and set the training window at 30 minutes. Then we make rolling one-minute-ahead predictions using the trained models. Similar to [Chinco et al. \(2019\)](#), we construct several linear regression benchmarks with different benchmark predictors. The benchmark predictors are some long-lived traditional factors, such as market return, size factor, and value factor from the Fama-French three-factor model. Then we use machine-learning methods to see whether cross-sectional returns have predictability, by adding all the cross-sectional returns and their lags as candidate predictors. We eliminate the stocks for which we find no corresponding symbol, or that are no longer in the S&P 500 index, ending up with 397 stocks remaining. The lagged returns of these 397 stocks enter the machine-learning models as candidate predictors. It is impossible to estimate such a large model using linear regression, because the number of variables is much greater than the training-sample size. This is one of the benefits of using machine-learning methods: we do not need to pre-determine the predictors. Instead, we put forward a large set of variables and let the model decide. This can be beneficial, since some signals might carry valuable information for only a short period of time, thus being unlikely for a human to capture.

We measure the accuracy of the predictions using out-of-sample R^2 , which is computed from mean-squared-error:

$$R_{oos}^2 = 1 - \frac{\sum (\hat{r}_t^{model} - r_t)^2}{\sum (r_t - \bar{r}_{t-30:t-1})^2},$$

where r^t is the true value at time t , \hat{r}_t^{model} is the prediction given by a certain model, and $\bar{r}_{t-30:t-1}$ is the sample average of the training sample. However, this is not what [Chinco et al. \(2019\)](#) use. They run the following regression:

$$r_{n,t} = \bar{a}_n + \bar{b}_n \hat{r}_{n,t}^{model} + e_{n,t},$$

where $r_{n,t}$ is the realized return, and $\hat{r}_{n,t}^{model}$ is the prediction given by the model. They report the R^2 from this regression. This measures how much variation in true values is explained by the predictions. More details about the metric are provided in later sections. For comparison, we also calculate the metric from their paper. We use it to explain some implications of our results.

Our results are surprising. LASSO and random forest do perform better than linear regression, but all models have a negative out-of-sample R^2 , indicating that they perform worse than the historical mean. This result is

stable during the whole sample period, and is robust with different training window sizes. So, we cannot say that there is any predictability in terms of out-of-sample R^2 . However, results from Chinco's metric suggest that the predictions from random forest can help explain some additional variation in the true values, i.e., adding random forest forecasts in addition to auto-regression predictions increases explanatory power in a statistically-significant way.

This article is arranged as follows. We review related literature in Section 2, then introduce the data and prediction methods in Section 3. Section 4 presents empirical results and implications. Section 5 concludes and provides further research ideas.

2. Related work

This article is related to a wide range of literature. First of all, it pertains to research on return predictability. Many scholars have proposed useful signals or methods to make predictions and find predictability in real data (Patelis, 1997; Campbell and Yogo, 2006; Ang and Bekaert, 2007). In particular, the well-known Fama-French three-factor model (Fama and French, 1993) is tested in many contexts. For example, Griffin (2002) finds that country-specific versions of Fama-French factors work better than global versions. Gaunt (2004) finds that size and value factors improve the predictability of the Fama-French model, using Australian stock market data. Suh (2009) tests it using CRSP data and finds that market premium is the most significant factor among the three. Some papers, however, present frustrating results. Welch and Goyal (2008) provide a good summary of widely-known stock return predictors. They test them on 30 years of data, and find that most signals in the literature are unstable, or even spurious. Moreover, most signals cannot beat the historical average, so give negative out-of-sample R^2 . Later, Campbell and Thompson (2008) propose adding weak restrictions on regressions to try to solve such challenges.

This article also builds on a large body of work on machine learning methods, especially on LASSO and random forest. LASSO was first introduced by Tibshirani (1996). Under the irrepresentable condition, LASSO achieves model selection consistency (Zhao and Yu, 2006). Also, LASSO has nice properties related to out-of-sample risk under certain mild conditions (Chatterjee, 2013). So LASSO is widely used as a method to select features when one believes that the true model is sparse. In this paper, we implicitly bet on sparsity when we use LASSO. The general method of random decision forests was first proposed by Ho (1995). Later, the idea of randomized node optimization was introduced by Dietterich (2000). Breiman (2001) proves

that there is a bound on the out-of-sample error of random forest, which depends on the strengths of the trees and their correlation.

In recent years, many authors have applied machine-learning techniques to financial economics, especially to asset pricing. [Rapach and Zhou \(2013\)](#) employ LASSO to predict global equity market returns using lagged returns of all countries. [Kozak et al. \(2017\)](#) argue that sparsity matters in predicting returns, and that imposing L1 & L2 penalties improves out-of-sample performance. [DeMiguel et al. \(2020\)](#) use LASSO to select six out of more than a hundred firm characteristics that are significant when constructing portfolios. [Gu et al. \(2018\)](#) compare multiple machine-learning models in forecasting asset returns. They find that trees and neural networks are the best-performing models, outperforming regressions in terms of out-of-sample R^2 . [Akyildirim et al. \(2021\)](#) use machine-learning classification algorithms, including random forest, to make predictions on the direction of cryptocurrency returns. They reach an average 55-65% predictive accuracy at the daily- or minute-level frequencies. All of these works have inspired us to apply machine-learning techniques to the topic we are interested in—asset return prediction.

3. Data and methodology

We now introduce our data and methods.

3.1 Data description

A. One-minute stock returns

The main data consists of the one-minute stock price series of S&P 500 companies.¹ We include only those stocks for which we can find corresponding symbols, and that are still in the S&P 500 index. In all, 397 stocks comprise our sample. For computational reasons, we limit the sample period to start on January 1, 2015, and end on February 17, 2017.² Every trading day is covered during this period, with 537 trading days in total. Each daily price series starts at 9:30 AM and ends at 3:59 PM. We exclude the last minute's data to eliminate the impact of close auction. This makes 399 returns for each stock, each day.

In our mean specifications, we use a 30-minute estimation window and include three lagged returns, so the first one-minute-ahead forecast is made at 10:04 AM each day. We train models on a rolling basis, thus training 356 models and making 356 predictions (10:04 AM, 10:05 AM, ..., 3:59 PM)

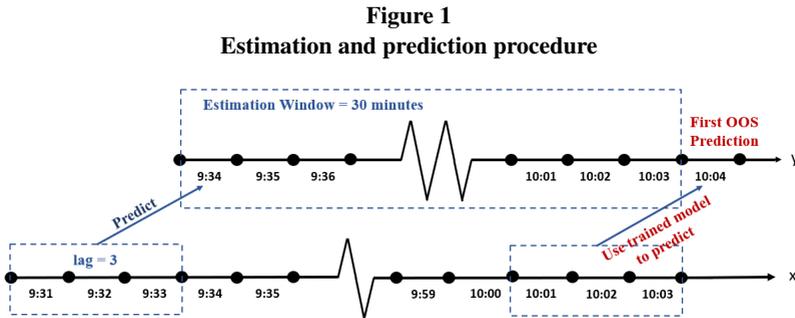
¹All thirty Dow Jones index components are in the S&P 500 index.

²February 17, 2017, is the last day for which data is currently available.

for each stock during one day. We train separate models for 30 predicted stocks, so each day we train $356 \times 30 = 10,680$ models. In total, we train $10680 \times 537 = 5,735,160$ models throughout the sample period.

We choose the lag to be 3 because [Chinco et al. \(2019\)](#) show that choosing $lag = 3$ performs best (they had tried $lag = 1, 2, 3, 4, 5$), and that the number of lags does not change the conclusion.

Figure 1 summarizes the procedure for making the first prediction at 10:04 AM.



B. Benchmark predictors

Like [Chinco et al. \(2019\)](#), we include several long-lived predictors in the benchmark linear regressions. We use AR(3) terms and Fama-French three factors. We use one-minute returns of iShares ETFs to construct three return series as proxies for Fama-French factors.³ To be more specific, the market factor is the one-minute return of iShares S&P 500 ETF (code IVV); SML (size factor) is the return of iShares Russell 2000 ETF (code IWM) minus the return of iShares Russell 1000 ETF (code IWB); HML (value factor) is the return of iShares S&P 500 Value ETF (code IVE) minus the return of iShare S&P 500 Growth ETF (code IVW).

We have three specifications for linear regression. The first is the AR(3) model, including only the stock itself's three lagged returns in the regression. The second is AR(3) terms plus three lagged market returns, for six regressors in total. The third is AR(3) terms plus three lagged Fama-French factors, thus giving twelve regressors.

³iShares are ETFs managed by BlackRock.

3.2 Prediction methods

We train models using three methods. Linear regression is used as the benchmark, with LASSO and random forest the two representative machine-learning methods.

A. Linear regression

We run linear regression in the simplest way: by estimating the ordinary least square (OLS):

$$\text{Benchmark1: } r_{n,t} = \alpha + \sum_{i=1}^3 \beta_i r_{n,t-i} + \varepsilon_{n,t},$$

$$\text{Benchmark2: } r_{n,t} = \alpha + \sum_{i=1}^3 \beta_i r_{n,t-i} + \sum_{i=1}^3 a_i r_{t-i}^{mkt} + \varepsilon_{n,t},$$

$$\begin{aligned} \text{Benchmark3: } r_{n,t} = \alpha + \sum_{i=1}^3 \beta_i r_{n,t-i} + \sum_{i=1}^3 a_i r_{t-i}^{mkt} \\ + \sum_{i=1}^3 b_i SML_{t-i} + \sum_{i=1}^3 c_i HML_{t-i} + \varepsilon_{n,t}. \end{aligned}$$

α is the intercept, which equals the historical average of the training sample. β_i, a_i, b_i and c_i are regression coefficients.

B. LASSO

Now suppose that the n^{th} stock's return is possible to relate to the lagged returns of any of $N = 397$ stocks, for example,

$$r_{n,t} = \alpha + \sum_{n'=1}^{3N} \beta_{n'} x_{n',t-1} + \varepsilon_{n,t}.$$

It is impossible to use linear regression, since there are $3N + 1 = 1192$ coefficients to be estimated. So we bet on sparsity. We assume that there are only a handful of variables at each prediction time (in other words, $S \ll 30$ if S is the number of non-zero coefficients). If that is true, we can use LASSO to estimate the model, by adding an $L1$ penalty term:

$$\hat{\alpha}, \hat{\beta} = \arg \min_{\alpha, \beta} \left\{ \frac{1}{L} \sum_{l=0}^{L-1} (r_{n,t} - \alpha + \sum_{n'=1}^{3N} \beta_{n'} x_{n',t-1})^2 + \lambda \sum_{n'=1}^{3N} |\beta_{n'}| \right\}.$$

We use BIC to choose λ for two reasons: (a) it is more efficient than cross-validation (Medeiros and Mendes, 2016); and (b) the sample size for training is only 30, so splitting it further can make estimation even more challenging. We use the *glmnet* package in R to train the models.

Other choices of penalty term exist, inducing different methods, such as ridge (Hoerl and Kennard, 1970), elastic net (Zou and Hastie, 2005), and SCAD (Fan and Li, 2001). However, while other penalty functions have different assumptions about the underlying model, and thus may give different model estimation, their prediction performance is similar. In other words, some of them may not be model-selection consistent or may give biased coefficients, but in theory, they have similar mean-squared-error and prediction properties under some mild conditions (Zhao and Yu, 2006).

C. Random forest

Trees are very different from regressions, as they are purely non-parametric. The idea of trees is easy to understand. They group the training data in a way that tries to capture the most similarity within groups. Using trees has many advantages. For example, trees are invariant to monotonic transformations of predictors, and a tree with depth L can capture $(L - 1)$ -way interactions (Gu et al., 2018).

However, trees are prone to over-fit, so researchers use some ensemble methods to combine forecasts from many different trees. In this article, we use random forest, which employs an ensemble method called “bagging.” Random forest has two sources of randomness. One comes from the bootstrap sample we draw, and another is the subset of the predictors we use when we grow a tree. This randomness aims to reduce the correlation between different trees and the variance of their average, thus stabilizing predictive performance.

We do not tune hyperparameters in this article, mainly because of the small training set. The two major hyperparameters in random forest are the number of trees and the size of the random subsets of features considered when splitting a node. We use the default value for feature size: $p/3 = 397$ (Breiman, 2001). Then we set the number of trees in the forest to be 100.⁴

4. Empirical results

We use out-of-sample R^2 to assess the performance of a method. The out-of-sample R^2 is a measure of how accurate the predictions are, compared

⁴We had tried both 100 and 300 trees in the main specification, and found no significant difference.

with the historical mean. It is computed by

$$R_{oos}^2 = 1 - \frac{\sum (\hat{r}_t^{model} - r_t)^2}{\sum (r_t - \bar{r}_{t-30:t-1})^2}, \quad (1)$$

where r^t is the true value at time t , \hat{r}_t^{model} is the prediction given by the model, and $\bar{r}_{t-30:t-1}$ is the sample average of the training sample.⁵ If the prediction has a smaller mean-squared-error than the historical mean does, the out-of-sample R^2 would be positive; otherwise, it is negative. We compute the out-of-sample R^2 each day for each stock, which means that the summation in Equation (1) is over 356 values. Then we take the average of all these R^2 .

4.1 Basic results

The results for different methods are summarized in Table 1. The values in parentheses are the standard deviations of R^2 . The first row presents the out-of-sample R^2 of random walk, i.e. all returns are 0. The first thing to notice is that only the first row reports a positive R^2 , which means that only random walk beats the historical mean in terms of out-of-sample R^2 .

Table 1
Out-of-sample R^2

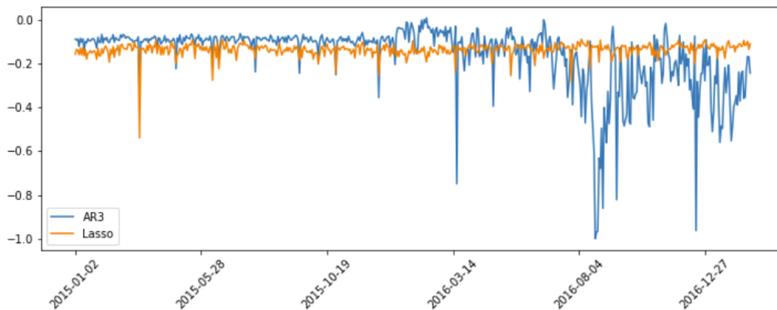
	average R^2 (%)
Random walk	3.551 (0.575)
AR3	-15.878 (14.461)
AR3+Market	-32.584 (19.950)
AR3+Market+Value+Size	-69.284 (17.876)
LASSO	-12.642 (3.035)
RF	-2.437 (1.223)

⁵Since sample size is 30 in the main specification, it is the average of 30 past returns.

Now we compare the results from the remaining non-trivial methods. The three linear regressions suggest clear over-fitting. AR(3) gives the best prediction among the three. As we use ever more predictors, the out-of-sample R^2 becomes increasingly negative. LASSO is a little bit better than linear regression, as its out-of-sample R^2 is larger than all linear regressions. The improvement is quite small, however, considering that we use many more variables than linear regressions as candidate variables. Random forest performs better than all the linear models, indicating non-linearity in the true model, but it is still worse than the historical mean. Also notable is that the standard deviations of R^2 from LASSO and random forest are much smaller than those of linear regressions, implying that machine-learning methods give more stable results than traditional regression.

Figure (2) also provides evidence for the time-series stability of machine-learning methods. We plot the daily average out-of-sample R^2 for AR(3) and LASSO. The figure shows that although AR(3) outperforms LASSO over a long period of time, it fluctuates considerably after March 2016. In contrast, the out-of-sample R^2 for LASSO is more stable during the whole sample period (around -13%).

Figure 2
Time series daily average R^2



We also use the model confidence set (MCS) approach to compare the prediction performance of the models. MCS was introduced by Hansen et al. (2011). Starting from the collection of all models, it uses an equivalence test and an elimination rule to identify the best collection of models sequentially. In our case, we include random walk, three AR models², LASSO, and random forest in the full collection to start. We concatenate the predictions of 30 stocks on the whole sample period to construct the loss matrix.

We use both squared-loss function and absolute-loss function to carry out MCS and use the $Tmax$ statistic from Hansen et al. (2011). The confidence

level is 0.1. The elimination procedures are summarized as follows.

Table 2
Elimination order

	squared-loss	absolute-loss
	superior set	superior set
Random walk		
AR3	3	3
AR3+Market	2	2
AR3+Market+Value+Size	1	1
LASSO	4	4
Random forest	5	5

These two types of loss functions give the same elimination order, which is in line with results presented in Table 1. LASSO and random forest are eliminated after AR models, but they are not as good as random walk. In these six models, random walk is the best regarding squared loss and absolute loss.

We still maintain hope for the cross-sectional returns. Perhaps the performance of the methods is different when they are “optimistic” or “pessimistic” about the future return. In other words, it is possible that LASSO and random forest are better at predicting positive results but worse at predicting negative results. If that is the case, we can still make use of these methods. For example, if it is very likely to be accurate when the method gives positive predictions, we can choose to act only when we observe a positive prediction. Bearing this in mind, we separate the samples into those with positive and negative forecasts, and compute the R^2 for both cases.

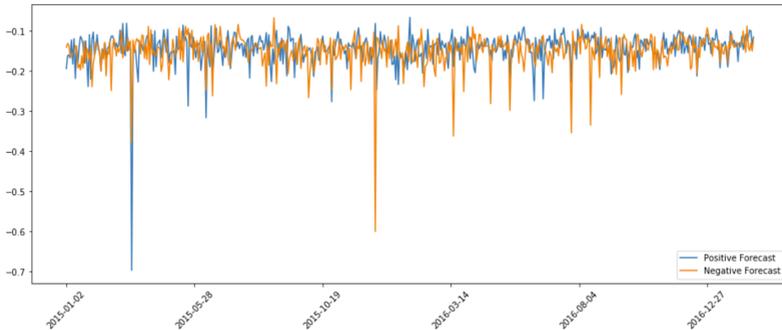
Unfortunately, the results show no significant difference between times with positive forecasts and times with negative forecasts. Table 3 summarizes the results. Indeed, the out-of-sample R^2 is a bit larger when the forecasts are positive, but the difference is negligible. We also plot the time-series out-of-sample R^2 of LASSO with positive and negative predictions separately. We observe no different patterns in the two time series.

Table 3
Positive vs negative

	AR3	LASSO	RF
positive	-17.714 (2.851)	-14.345 (1.542)	-2.974 (0.415)
negative	-18.241 (2.849)	-14.933 (1.384)	-3.535 (0.430)

Notes: These values are out-of-sample R^2 (in %) computed from Equation (1). As before, we compute the R^2 for each stock on each day, then take the average.

Figure 3
LASSO: Positive vs negative



4.2 Robustness checks

Although we had hoped that adding cross-sectional returns could help predict one-minute-ahead returns, the empirical results show no evidence of predictability. These results are surprising and somewhat frustrating. It confirms the extreme difficulty of forecasting Dow Jones stocks, even using all large stocks' cross-sectional returns.

In this section, we consider two variations of the main specifications. One is to train the models using longer windows. Over-fitting is a severe problem when we train these models, so larger training samples might help. Beyond that, we try to predict five-minute returns instead of one-minute returns. This task may be simpler, since a longer prediction horizon means less noise.

A. Different windows

There is a trade-off when selecting window size. On the one hand, a shorter window results in a smaller training set, leading to over-fitting; on the other hand, with a shorter window we capture more recent information from the market. If the training window is too large, we risk including too much outdated information in model training.

On average, LASSO selects 11.5 variables. This means that in the previous specification, we use 30 samples to estimate approximately 12~13 coefficients. It is a challenge. So we decide to try with longer windows: 60 minutes and 120 minutes. We use a shorter sample period (04/01/2016 to 02/17/2017) for this part, due to excessively long computation time.

Table 4
OOS R^2 with different windows

Window size	Random walk	AR3	LASSO	RF
30	3.615 (0.487)	-24.910 (17.935)	-13.118 (2.498)	-2.627 (1.314)
60	1.899 (0.361)	-3.813 (1.271)	-0.327 (0.563)	-2.682 (1.178)
120	1.081 (0.328)	-1.366 (1.039)	-0.014 (0.326)	-2.415 (1.238)

Note: The values in the first row are different from those in Table 1 because the sample period is shorter here.

According to Table 4, the qualitative conclusion does not change. Random walk is still the only method that beats the historical mean. The accuracy of AR(3) and LASSO does increase with longer training windows, but the out-of-sample R^2 is still negative. The performance of LASSO draws closer to the historical mean, as its out-of-sample R^2 is quite close to 0. However, this might be due to the fact that the trained models are so similar to taking the historical mean. Indeed, when window = 120, LASSO selects less than one predictor, on average. That means that LASSO does not select even one single variable most of the time, and the prediction is exactly the historical mean when this happens. The performance of random forest does not change much as we use a longer training window.

B. Five-minute return

We imagine that prediction could be easier when the prediction horizon is longer. After all, auto-correlation in monthly returns is larger than auto-correlation in minute returns because there is a lot of noise in high-frequency data. So we try to forecast the five-minute-ahead returns.

We compute the five-minute returns throughout the day: 77 returns each day (9:30-9:35, 9:35-9:40, . . . , 15:50-15:55). Then we use window = 30 and lag = 3, producing 44 forecasts per day. The models run during the full sample period (01/01/2015 - 02/17/2017). The results are shown in Table 5.

Table 5
OOS R^2 with 5-min return

	Random walk	AR3	LASSO	RF
average R^2	3.969 (1.977)	-10.164 (3.246)	-15.409 (4.256)	-3.983 (2.745)

The pattern of results here is very similar to that of one-minute returns. Random walk is the only method that beats the historical average. LASSO did an even worse job, compared to predicting one-minute returns. So we have no evidence of predictability in five-minute cross-sectional returns, either.

4.3 Comparison with [Chinco et al. \(2019\)](#)

Our results are very different from those of [Chinco et al. \(2019\)](#). They find predictability in cross-sectional returns, and that using LASSO can greatly increase the out-of-sample fit. It can generate large benefits in high-frequency trading.

Many possibilities might explain this difference. The major one is that the data sets are different. [Chinco et al. \(2019\)](#) use the cross-sectional returns of all NYSE-listed companies to make predictions on 250 randomly-selected stocks, while we have a limited data set of only S&P 500 components. We also make predictions on 30 Dow Jones stocks, which are extremely large. We find negative out-of-sample R^2 , implying that adding cross-sectional returns of large stocks does not improve forecast accuracy. Combining our results with those of [Chinco et al. \(2019\)](#), we believe that the predictability they find comes mainly from small stocks, or from the interaction between large and small stocks. In other words, when predicting large stocks' returns, it is the returns of small stocks that make a large contribution.

Another reason is that we use different metrics to evaluate prediction performance. We use out-of-sample R^2 defined in Equation (1), which [Chinco](#)

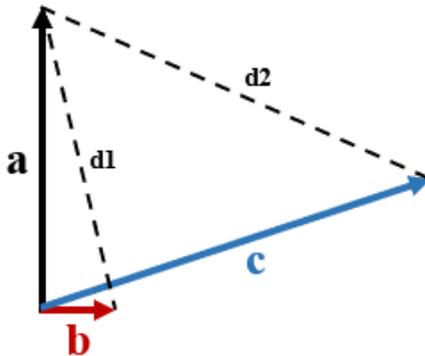
et al. (2019) do not present in their paper. Instead, they run the following regression and report the R^2 :

$$r_{n,t} = \bar{a}_n + \bar{b}_n \hat{r}_{n,t}^{model} + e_{n,t}. \quad (2)$$

The R^2 of this regression is the percentage of variation in true returns explained by the predictions. It is also the squared information coefficient (IC) of $\hat{r}_{n,t}^{model}$. So it measures the correlation between \hat{r}_t and r_t . This is distinct from measuring the distance between \hat{r}_t and r_t . See Figure 4 as an illustration. Think of three vectors: a , b and c . The distances between a,b and a,c are d_1 and d_2 , respectively. From the plot, we can see that $d_1 < d_2$. So b is closer to a in terms of distance. But the correlation between a and b is smaller because the angle between vector a and b is larger.

Both metrics have their own advantages. Using distance measures the precision of the prediction, and takes into account the magnitude of the prediction. In contrast, multiplying a constant does not change the IC of the prediction. However, IC is widely used in industry because prediction in finance is always quite noisy, and IC is a good measure of the prediction “shape.”

Figure 4
Illustration of two measures



Chinco et al. (2019) also report the increase in adjusted- R^2 after adding the LASSO predictions to the regression above. They argue that “using LASSO in addition to a benchmark model increases out-of-sample fit.” To see that, they run the following regression:

$$r_{n,t} = \bar{a}_n + \bar{b}_n \hat{r}_{n,t}^{bmk} + \bar{c}_n \hat{r}_{n,t}^{ML} + e_{n,t}. \quad (3)$$

If including the LASSO prediction increases the adjusted- R^2 statistic, it implies that the cross-sectional returns must explain some variation that is not already explained by the benchmark model. Define

$$\Delta R^2 \triangleq R^{2,both} - R^{2,bmk},$$

where $R^{2,both}$ is the adjusted- R^2 from (3) and $R^{2,bmk}$ is the adjusted- R^2 from (2). They find the average increase of adjusted- R^2 is 2.4%.

To better compare with [Chinco et al. \(2019\)](#), and to explain the implications of our results, we compute the metric in their paper as well. Obviously, the R^2 of random walk is 0. Random walk cannot explain any variation. We run regression (2) for every stock on every day,⁶ then compute the average adjusted- R^2 . These results are summarized in Table 6.

Table 6
Single model results

	mean (%)	95% CI	<i>p</i> -value
Historical mean	0.493 (0.217)	[0.067, 0.919]	0.012
AR3	0.567 (0.193)	[0.189, 0.945]	0.002
AR3+Market	0.552 (0.194)	[0.172, 0.932]	0.002
AR3+Market+Value+Size	0.497 (0.162)	[0.179, 0.814]	0.001
LASSO	0.439 (0.141)	[0.162, 0.715]	0.001
RF	0.447 (0.150)	[0.153, 0.742]	0.001

The historical average explains 0.49% of the variation in the true values. Linear regressions, LASSO and random forest do not give significantly better results. Lasso and random forest are worse than linear regression, indicating that using machine-learning predictions alone does not generate higher explanatory power. AR(3) is the single best model, in terms of adjusted- R^2 .

⁶There are 356 samples in each regression.

Chinco et al. (2019) argue that the absolute level of adjusted- R^2 does not matter; as long as the cross-sectional returns provide additional information relative to benchmarks, they are useful. So instead of just comparing the adjusted- R^2 from single models, the increase in adjusted- R^2 could be meaningful to explore. To test this, we run regression (3). The results are summarized in Table 7.

Table 7
Increase in Chinco's metric

	Benchmarks	$\bar{R}^2(\%)$	$\Delta\bar{R}^2(\%)$	95% CI
LASSO	AR3	0.763 (0.171)	0.196 (0.237)	[-0.268, 0.661]
	AR3+Market	0.764 (0.175)	0.212 (0.245)	[-0.267, 0.693]
	AR3+Market+Value+Size	0.762 (0.161)	0.266 (0.217)	[-0.159, 0.690]
Random forest	AR3	0.774 (0.181)	0.207 (0.086)	[0.038, 0.376]
	AR3+Market	0.774 (0.186)	0.222 (0.087)	[0.051, 0.393]
	AR3+Market+Value+Size	0.772 (0.186)	0.275 (0.081)	[0.116, 0.434]

Note: The first column presents the average adjusted- R^2 from regression (3). The second column is the average of increases in adjusted- R^2 , and the third column is the 95% confidence interval of the difference.

We use an approach from Giacomini and White (2006) to assess the statistical significance of ΔR^2 . We can see that the increases in adjusted- R^2 are very small, less than 0.45% in all cases. The increase is not significant when we add the LASSO predictions to the regression, but is statistically significant if we add the random forest predictions. This suggests that nonlinearity plays a role when combining the cross-sectional returns.

Although the increase is small (Chinco et al. (2019) report a 2.4% increase), adding cross-sectional returns does help explain the variation in returns. It seems that this conclusion contradicts what we find in out-of-sample R^2 . But it does not. As we explained, these two metrics focus on different aspects of the predictions. However, Table 7 suggests a possible direction that may improve the predictions. If cross-sectional returns can increase the correlation, but do not perform well in terms of mean-squared-error, there may be some biases in the predictions. It is possible to find a way to correct the bias and improve model performance.

As in the previous section, we also compute the ΔR^2 using the predictions of different windows. Table 8 shows the results of adding predictions from LASSO and random forest. It is surprising that we get statistically-significant increases in adjusted- R^2 for LASSO predictions. The increases are still small, but are significant at a 5% level.

Table 8
Different windows

Window		\bar{R}^2	$\Delta \bar{R}^2$	95% CI
60	AR3	0.652 (0.239)	- -	-
	LASSO	1.223 (0.347)	0.570 (0.196)	[0.185, 0.955]
	RF	1.131 (0.314)	0.479 (0.158)	[0.170, 0.788]
120	AR3	0.889 (0.360)	- -	-
	LASSO	1.594 (0.484)	0.704 (0.243)	[0.228, 0.181]
	RF	1.435 (0.403)	0.545 (0.166)	[0.220, 0.870]

Note: \bar{R}^2 for AR3 is the adjusted- R^2 from regression 2, while \bar{R}^2 for LASSO and RF is from regression (3). The second column shows the average and standard deviation of $\Delta \bar{R}^2$. The third column is the 95% confidence interval of $\Delta \bar{R}^2$.

5. Conclusion

It is an interesting yet challenging task to forecast stock returns in high frequency. In this article, we used cross-sectional returns of S&P 500 components to make one-minute-ahead return predictions of the Dow Jones stocks. We trained models on 30-minute rolling windows. We applied LASSO and random forest to train the models. We used three linear regressions with different sets of long-lived predictors as benchmarks. We computed out-of-sample R^2 to evaluate the performance of these methods.

The results are surprising. All methods give negative out-of-sample R^2 , which means that they have worse precision than the historical average. The results are stable during the whole sample period and are robust with different training window sizes. We find no evidence of predictability in the intraday cross-sectional returns of large stocks.

These results are different from those of [Chinco et al. \(2019\)](#). They find that using LASSO in addition to the AR(3) model increases the out-of-sample fit by more than 2.4%, on average. There are two possible implications of the different results. First, the predictability of cross-sectional returns is mainly from small stocks, or the interaction between large and small stocks. Second, although cross-sectional returns have some information about the “shape” of the returns, the predictions from LASSO and random forest are likely to be biased, i.e., they are not precise in levels, leading to large mean-squared-error. The reason for the bias is still unclear.

The unexpected results create ample need for further research. If it were possible to access the data of all NYSE-listed stocks, one could investigate directly whether the predictability is from the interaction between large and small stocks. The simplest way is to include only small stocks’ cross-sectional returns as candidate variables to forecast returns of large stocks and see if there is any predictability.

Acknowledgments

The author wishes to thank Marcelo Cunha Medeiros for advice and instructions on this thesis.

References

- Akyildirim, E., Goncu, A. and Sensoy, A. (2021). Prediction of cryptocurrency returns using machine learning, *Annals of Operations Research* **297**(1): 3–36.
- Aldridge, I. and Krawciw, S. (2017). *Real-time risk: What investors should know about FinTech, high-frequency trading, and flash crashes*, John Wiley & Sons.
- Ang, A. and Bekaert, G. (2007). Stock return predictability: Is it there?, *Review of Financial Studies* **20**(3): 651–707.
- Ball, R. (1978). Anomalies in relationships between securities’ yields and yield-surrogates, *Journal of Financial Economics* **6**(2-3): 103–126.
- Breiman, L. (2001). Random forests, *Machine Learning* **45**(1): 5–32.
- Campbell, J. Y. and Shiller, R. J. (1988). Stock prices, earnings, and expected dividends, *Journal of Finance* **43**(3): 661–676.

- Campbell, J. Y. and Thompson, S. B. (2008). Predicting excess stock returns out of sample: Can anything beat the historical average?, *Review of Financial Studies* **21**(4): 1509–1531.
- Campbell, J. Y. and Yogo, M. (2006). Efficient tests of stock return predictability, *Journal of Financial Economics* **81**(1): 27–60.
- Chatterjee, S. (2013). Assumptionless consistency of the Lasso, *arXiv preprint arXiv:1303.5817*.
- Chinco, A., Clark-Joseph, A. D. and Ye, M. (2019). Sparse signals in the cross-section of returns, *Journal of Finance* **74**(1): 449–492.
- Connor, G. and Sehgal, S. (2001). Tests of the Fama and French model in India.
- DeMiguel, V., Martin-Utrera, A., Nogales, F. J. and Uppal, R. (2020). A transaction-cost perspective on the multitude of firm characteristics, *Review of Financial Studies* **33**(5): 2180–2222.
- Dietterich, T. G. (2000). Ensemble methods in machine learning, *International workshop on multiple classifier systems*, Springer, pp. 1–15.
- Dow, C. H. (1920). *Scientific stock speculation*, Magazine of Wall Street.
- Fama, E. F. and French, K. R. (1993). Common risk factors in the returns on stocks and bonds, *Journal of Financial Economics* **33**(1): 3–56.
- Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties, *Journal of the American Statistical Association* **96**(456): 1348–1360.
- Freyberger, J., Neuhierl, A. and Weber, M. (2020). Dissecting characteristics nonparametrically, *Review of Financial Studies* **33**(5): 2326–2377.
- Gaunt, C. (2004). Size and book to market effects and the Fama French three factor asset pricing model: Evidence from the Australian stockmarket, *Accounting & Finance* **44**(1): 27–44.
- Giacomini, R. and White, H. (2006). Tests of conditional predictive ability, *Econometrica* **74**(6): 1545–1578.
- Goyal, A. and Welch, I. (2003). Predicting the equity premium with dividend ratios, *Management Science* **49**(5): 639–654.

- Graham, B., Dodd, D. L. F., Cottle, S. et al. (1934). *Security analysis*, McGraw-Hill New York.
- Griffin, J. M. (2002). Are the Fama and French factors global or country specific?, *Review of Financial Studies* **15**(3): 783–803.
- Gu, S., Kelly, B. and Xiu, D. (2018). Empirical asset pricing via machine learning, *Technical report*, National Bureau of Economic Research.
- Hansen, P. R., Lunde, A. and Nason, J. M. (2011). The model confidence set, *Econometrica* **79**(2): 453–497.
- Harvey, C. R., Liu, Y. and Zhu, H. (2016). ...and the cross-section of expected returns, *Review of Financial Studies* **29**(1): 5–68.
- Herwartz, H. (2017). Stock return prediction under garch—an empirical assessment, *International Journal of Forecasting* **33**(3): 569–580.
- Ho, T. K. (1995). Random decision forests, *Proceedings of 3rd international conference on document analysis and recognition*, Vol. 1, IEEE, pp. 278–282.
- Hoerl, A. E. and Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems, *Technometrics* **12**(1): 55–67.
- Huang, R. D. and Stoll, H. R. (1994). Market microstructure and stock return predictions, *Review of Financial Studies* **7**(1): 179–213.
- Jobson, J. D. and Korkie, B. (1980). Estimation for Markowitz efficient portfolios, *Journal of the American Statistical Association* **75**(371): 544–554.
- Kothari, S. P. and Shanken, J. (1997). Book-to-market, dividend yield, and expected market returns: A time-series analysis, *Journal of Financial Economics* **44**(2): 169–203.
- Kozak, S., Nagel, S. and Santosh, S. (2017). Shrinking the cross section, *Technical report*, National Bureau of Economic Research.
- Lettau, M. and Ludvigson, S. (2001). Consumption, aggregate wealth, and expected stock returns, *Journal of Finance* **56**(3): 815–849.
- Li, X., Huang, X., Deng, X. and Zhu, S. (2014). Enhancing quantitative intra-day stock return prediction by integrating both market news and stock prices information, *Neurocomputing* **142**: 228–238.

- Lintner, J. (1975). Inflation and security returns, *Journal of Finance* **30**(2): 259–280.
- Medeiros, M. C. and Mendes, E. F. (2016). L1-regularization of high-dimensional time-series models with non-Gaussian and heteroskedastic errors, *Journal of Econometrics* **191**(1): 255–271.
- Mizuno, H., Kosaka, M., Yajima, H. and Komoda, N. (1998). Application of neural network to technical analysis of stock market prediction, *Studies in Informatic and Control* **7**(3): 111–120.
- Patelis, A. D. (1997). Stock return predictability and the role of monetary policy, *Journal of Finance* **52**(5): 1951–1972.
- Rapach, D. and Zhou, G. (2013). Forecasting stock returns, *Handbook of Economic Forecasting*, Vol. 2, Elsevier, pp. 328–383.
- Suh, D. (2009). Stock returns, risk factor loadings, and model predictions: A test of the CAPM and the Fama-French 3-factor model.
- Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso, *Journal of the Royal Statistical Society: Series B (Methodological)* **58**(1): 267–288.
- Welch, I. and Goyal, A. (2008). A comprehensive look at the empirical performance of equity premium prediction, *Review of Financial Studies* **21**(4): 1455–1508.
- Zhao, P. and Yu, B. (2006). On model selection consistency of Lasso, *Journal of Machine Learning Research* **7**: 2541–2563.
- Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net, *Journal of the Royal Statistical Society: series B (statistical methodology)* **67**(2): 301–320.