Seemingly Virtuous Complexity in Return Prediction *

Stefan Nagel†
University of Chicago,
NBER, CEPR, and CESifo

April 4, 2025

PRELIMINARY AND INCOMPLETE

Return prediction with Random Fourier Features (RFF)—a very large number, P, of nonlinear transformations of a small number, K, of predictor variables—has become popular recently. Surprisingly, this approach appears to yield successful out-of-sample stock market index return predictions even when trained on datasets as small as $T = 12$ months with P in the thousands, and without shrinkage. However, I show that this apparent virtue of complexity is an illusion. When P far exceeds T, the return forecasts from the high-complexity predictive regression based on RFF effectively reduce to forecasts from a low-complexity kernel ridge regression. This regression constructs forecasts by positively weighting the past T returns—essentially a momentum strategy. The original K predictor variables serve only to define a distance metric: when a lagged predictor vector closely resembles the current one, the predicted return assigns greater weight to the corresponding past return. Moreover, because the distance to lagged predictors increases during periods of high predictor volatility, the weights on past returns decrease in such times—making the high-complexity RFF approach essentially a simple volatility-timed momentum strategy. Consistent with this interpretation, I find that both a kernel-based strategy and a basic volatility-timed momentum strategy explain the excess returns earned by the high-complexity RFF approach.

*I am grateful for comments to Darrell Duffie, Amit Goyal, Leonid Kogan, Ralph Koijen, Serhiy Kozak, and Dacheng Xiu.

†University of Chicago, Booth School of Business, 5807 South Woodlawn Avenue, Chicago, IL 60637, e-mail: stefan.nagel@chicagobooth.edu

I. Introduction

Research in machine learning has found that heavily overparametrizing models such that they fit the training data perfectly can result in good out-of-sample predictions. In a linear regression setting, this means that the number of covariates may far exceed the number of observations in the training data [Belkin, Hsu, Ma, and Mandal (2019), Bartlett, Long, Lugosi, and Tsigler (2020), Hastie, Montanari, Rosset, and Tibshirani (2022)]. These findings in machine learning have inspired a fast-growing literature in empirical asset pricing that uses Random Fourier Features (RFF)—a very large number, P, of randomized nonlinear transformations of a small number, K, of predictor variables—for return prediction and for modeling of the stochastic discount factor (SDF).[1]

Kelly, Malamud, and Zhou (2024) (KMZ), the pioneering paper using this approach, presents a stunning result. In a time-series setting for predicting excess returns on the CRSP value-weighed index, regressions with $P = 12,000$ RFF formed from $K = 15$ variables—predictors from Welch and Goyal (2008) augmented with lagged index returns—produce a market timing strategy with strong out-of-sample performance even with rolling training data windows as short as $T = 12$ months and in ridgeless regression, i.e., without explicit shrinkage. This is a stunning result because most of the Goyal-Welch predictor variables are highly persistent and the forecasting target is extremely noisy. Conventional wisdom is that it takes sample sizes that span decades, not just 12 months, to extract useful predictive information for future returns from these variables. The results are also puzzling from a theoretical perspective, as overfitting without explicit regularization is benign only in high signal-to-noise ratio settings (Mei and Montanari 2022), but not when signals are weak (Shen and Xiu 2025), which is arguably the case in a return prediction application.

Consistent with the conventional wisdom, I find that the high-complexity ridgeless regression with training window of $T = 12$ months and $P = 12,000$ RFF in fact does not learn to extract useful predictive information from the K predictors. Instead, the market-timing strategy based on the high-complexity ridgeless regression forecasts from short training windows reduces to a certain volatility-timed momentum strategy that forms the predicted market return as a weighted-average of the past monthly returns during the training window, with the weights decreasing in the level of the average volatility of the predictor variables in the training window. Importantly, the

1. See Kelly, Malamud, and Zhou (2024), Jensen, Kelly, Malamud, and Pedersen (2024), Didisheim, Ke, Kelly, and Malamud (2023), Didisheim, Ke, Kelly, and Malamud (2024), Kelly, Malamud, and Zhou (2022),

ridgeless regression does not learn from the data that a volatility-timed momentum strategy works. The reason why the market timing positions resemble a volatility-timed momentum is a rather mechanical one, and that this strategy happens to perform well in empirical data is a coincidence.

The reason why the strategy based on thousands of RFF predictors collapses to a simple volatility-timed momentum strategy follows from the properties of RFF. When P is much larger than K, then, as shown in Rahimi and Recht (2007) and Sutherland and Schneider (2015), inner products of vectors of large numbers of RFF converge to a Gaussian kernel of these K variables. Forecasts from ridgeless regression with RFF then reduce to forecasts from a kernel ridgeless regression where inner products of RFF are replaced by Gaussian kernels. The effective complexity of this kernel ridgeless regression is low because it is capped by the number of observations in the training window. In the case $T = 12$, the effective number of parameters is 12, orders of magnitude lower than the number of RFF when P is in the thousands. Predictive success in the large-P, small-T case therefore cannot be a consequence of virtue of complexity.

Examination of the kernel ridgeless regression estimator sheds light on the source of return predictability that the KMZ's approach effectively exploits. The kernel ridgeless regression constructs return forecasts by weighting the $T = 12$ lagged returns in the training data according to the distance of the current K-dimensional predictor variable vector to the predictor variable vector at the corresponding lag. For example, if the one-month lagged predictor vector is more similar to the current one than the two-month lagged predictor vector is, then the one-month lagged return receives a higher weight in the construction of the return forecast than the two-month lagged return. More generally, because predictor variables have persistence, more recent returns receive higher weights, and the weights are all positive. In other words, the return forecast resembles a momentum strategy. Furthermore, because distance between predictor vectors is smaller in times when predictors are not volatile, the magnitude of the weights is inversely related to predictor variable volatility in the training window. The two effects combined therefore generate a volatility-timed momentum strategy, where volatility refers to the volatility of the predictors in the training window.

This makes clear that the seemingly high-complexity ridgeless regression with RFF does not extract information about nonlinear relationships between the K predictor variables and future returns from the short training data. The only information used is the very limited information embodied in the distance of the lagged predictor vectors from the current predictor vector. This

limited information involves only two aspects. First, closer lags of predictor vectors have smaller distance to the current predictor vector. This is a property of any vector-autoregressive process with persistence and does not reflect predictive content that the predictor variables may have for future returns. Second, there are times when the distances between predictors are smaller than in other times. This reflects time-varying volatility of shocks to the predictors. This volatility does not embody predictive content of the predictors for future returns.

At a conceptual level, the method does is what it is supposed to do. When P is much larger than K, ridgeless regression with RFF converges to a low-complexity kernel ridge regression, which in turn is basically a nonparametric approach that predicts future returns by smoothing past returns in time periods when the predictor variable vector was similar to the current one at the time of prediction. With a very long training data set, the approach would then look for historical periods that could be decades ago, in which the vector of predictor variables looked similar to the current predictor vector, and then use the returns experienced around that time as a forecast for future returns. However, when the training data sets are very short, as in KMZ, such as in the rolling windows approach with $T = 12$ months, then all the method does is to average a few returns in recent months, as those are the ones with highest predictor-vector similarity.

Empirically, a market-timing strategy that directly uses the low-complexity kernel ridgeless regression approach based on the $K = 15$ input variables achieves abnormal out-of-sample performance that is slightly stronger than KMZ's high-complexity market timing strategy based on the ridgeless regression with thousands of RFF. Moreover, KMZ's high-complexity strategy earns zero alpha when evaluated with a respect to a two-factor model that includes the market factor and the return of the kernel ridgeless regression strategy. This is natural, as the market timing positions taken by the kernel ridgeless regression strategy are almost identical to those taken by KMZ's high-complexity RFF-based strategy. This confirms empirically that the ridgeless regression with RFF in short training windows does not learn about predictive relationships between the K variables and future returns in the training data. Evidently, the same market timing positions are reached with a low-complexity strategy that uses no information other than the distance of the low-dimensional predictor vectors from each other within the training data. In fact, even a simple volatility-timed momentum strategy that puts linearly declining weights on the past 12 months returns interacted with the reciprocal of predictor volatility in the past 12-month window achieves

similar performance as the high-complexity RRF-based strategy and, when used as an explanatory factor, it absorbs most of the abnormal performance of the high high-complexity RRF-based strategy.

I also show empirically that the high-complexity ridgeless regression with RFF does not extract information about predictive relationships from the training data. I generate artificial return data by adding a observations from a simulated MA(2) process with strong negative autocorrelation to the actual market index return data. When I feed this artificial return series into the high-complexity ridgeless regression with RFF, it still produces weights on recent past returns that are positive, and hence effectively still a volatility-timed momentum strategy, even though the returns data now features reversals, not momentum. As a consequence, the strategy earns negative abnormal returns out-of-sample. This shows that the high-complexity ridgeless regression with RFF does not learn from the training data about the presence of momentum or reversal effects.

While most of my analysis focuses on the time-series predictive regression setting of KMZ, I also show that similar complexity-reducing effects also appear in a cross-sectional asset pricing setting when P RFF of K firm characteristics are used as weights to construct RFF factors from a panel of stock returns. When P is much larger than T, a high-complexity SDF with thousands of RFF factors as in Didisheim, Ke, Kelly, and Malamud (2024) collapses to a low-complexity SDF. The mean-variance efficient portfolio weights implied by this SDF for the original stock returns are kernel-weighted averages of past returns, where past returns of stocks with similar characteristics receive higher weight.

II. Properties of Predictive Regressions with Random Fourier Features in the High-Complexity Case

I focus on the case of ridgeless regression. The results with ridgeless regression are the most remarkable finding in KMZ, as the predictive regression seem to achieve high predictive power without explicit ridge shrinkage in a setting where the number of predictor variables, P, vastly outnumbers the number of observations used to estimate the regression, T. Throughout the analysis below I focus on this case of $P = 12,000$ and $T = 12$, which produces the most surprising findings in KMZ.

II.A. Preliminaries: Predictability Induced by Standardization of the Dependent Variable

KMZ standardize the dependent variable in the predictive regressions, and the return to be forecasted, with the standard deviation of returns over the previous 12 months. This can generate predictability that does not exist in the returns before standardization. Appendix A discusses the issue in more detail. The bias in predictability is small, but to completely avoid it, I deviate from KMZ and work with returns that are not standardized.

II.B. Properties of Ridgeless Regression when $P > T$

Let T denote the number of observations used in rolling predictive regressions where the dependent variable, the return to be predicted, is measured in period $t-T$ to t, collected in the vector $\boldsymbol{y}_t$ and the realizations of the P predictor variables from $t-T$ to $t-1$ in the P columns of the $T \times P$ matrix $\boldsymbol{Z}_{t-1} = (\boldsymbol{z}_{t-1} \quad \boldsymbol{z}_{t-2} \quad ... \quad \boldsymbol{z}_{t-T})'$. I focus on the case $P > T$.

The ridgeless OLS estimator in this case is

$$\hat{\boldsymbol{b}}_t = (\boldsymbol{Z}'_{t-1}\boldsymbol{Z}_{t-1})^+ \boldsymbol{Z}'_{t-1}\boldsymbol{y}_t \tag{1}$$

where $^+$ denotes the Moore-Penrose pseudoinverse. As the number of predictors is higher than the number of return observations in $\boldsymbol{y}_t$, the regression perfectly fits the training data in the window of length T.

With the predictor variable observations in period t collected in the vector $\boldsymbol{z}_t$, the predicted value for y_{t+1} is

$$\hat{y}^{\text{complex}}_{t+1|t} = \boldsymbol{w}'_t\boldsymbol{y}_t, \qquad \text{with} \quad \boldsymbol{w}'_t = \boldsymbol{z}'_t(\boldsymbol{Z}'_{t-1}\boldsymbol{Z}_{t-1})^+ \boldsymbol{Z}'_{t-1} \tag{2}$$

The predicted return, and hence the market timing position taken by this strategy, is therefore a weighted average of the T returns in $\boldsymbol{y}_t$. Considering that $P > T$ and using the rules for the

Moore-Penrose pseudoinverse, we can rewrite the weights as

$$\begin{aligned}
\boldsymbol{w}_t' &= \boldsymbol{z}_t'(\boldsymbol{Z}_{t-1}'\boldsymbol{Z}_{t-1})^+\boldsymbol{Z}_{t-1}' \\
&= \boldsymbol{z}_t'\boldsymbol{Z}_{t-1}^+(\boldsymbol{Z}_{t-1}')^+\boldsymbol{Z}_{t-1}' \\
&= \boldsymbol{z}_t'\boldsymbol{Z}_{t-1}'(\boldsymbol{Z}_{t-1}\boldsymbol{Z}_{t-1}')^{-1}(\boldsymbol{Z}_{t-1}\boldsymbol{Z}_{t-1}')^{-1}\boldsymbol{Z}_{t-1}\boldsymbol{Z}_{t-1}' \\
&= \boldsymbol{z}_t'\boldsymbol{Z}_{t-1}'(\boldsymbol{Z}_{t-1}\boldsymbol{Z}_{t-1}')^{-1}. \qquad (3)
\end{aligned}$$

This provides an interpretation of the T weights: They represent the coefficients in a regression of P predictor variable observations in the vector $\boldsymbol{z}_t$ on lagged observations of the predictors in periods $t-1$ to $t-T$. Roughly speaking, this regression evaluates which of the past vectors of predictors $\boldsymbol{z}_{t-1}, \boldsymbol{z}_{t-2}, ..., \boldsymbol{z}_{t-T}$ is most similar to $\boldsymbol{z}_t$. The regression coefficients, and hence the weights $\boldsymbol{w}_t$ then reflect this similarity. For example, if the predictors follow an autoregressive process with persistence, a $\boldsymbol{z}_{t-k}$ that is closer in time to $\boldsymbol{z}_t$ will tend to be more similar to $\boldsymbol{z}_t$ than predictor vectors that are more distant in time. As a consequence, the weights w_{t-k} for small k will tend to be higher than those for larger k. When these weights are then applied to lagged returns in the construction of $\hat{y}_t = \boldsymbol{w}_t'\boldsymbol{y}_t$, this results in a a version of a momentum strategy, with higher weights on the most recent returns.

That the predicted return is a weighted average of past returns is always true for any predicted regression estimated on past returns in the high-complexity case where the number of predictor variables exceeds the number of observations in the training window. The question is whether the weights placed on these past returns reflect any predictive information that predictor variables have about future returns. As I will show next, when the predictors are constructed as RFF of a small number of variables, more can be said about what kind of information from predictors is used in construction of the weights.

II.C. Forecasts from Ridgeless Regression with Random Fourier Features Collapse to Forecasts from a Low-Complexity Gaussian Kernel Ridgeless Regression

KMZ construct $\boldsymbol{z}$ as a very large number of nonlinear transformations of a small number of predictor variables from the Goyal-Welch data set as predictors. Specifically, the nonlinear transformations take the form of Random Fourier Features (RFF). Results on the convergence of inner products

of RFF help shed light on what happens to KMZ's ridgeless regression estimator when P is much larger than T.

The $K = 15$ predictor variables used by KMZ include 14 predictor variables from the Goyal-Welch data set as well as the one-month lagged market index return. KMZ then form RFF where consecutive elements i and $i+1$ of $\boldsymbol{z}_t$ are constructed as

$$\begin{pmatrix} z_{i,t} \\ z_{i+1,t} \end{pmatrix} = \sqrt{\frac{2}{P}} \begin{pmatrix} \cos(\gamma \boldsymbol{\omega}_i' \boldsymbol{x}_t) \\ \sin(\gamma \boldsymbol{\omega}_{i+1}' \boldsymbol{x}_t) \end{pmatrix} \quad \boldsymbol{\omega}_i \sim \text{ IID } \mathcal{N}(0, \boldsymbol{I}), \quad i = 1, ..., P/2 \tag{4}$$

with $\gamma = 2$. KMZ form up to 6,000 of such pairs.[2] Focusing on the highest number they consider, we then have $P = 2 \times 6,000 = 12,000$. KMZ then standardize the RFF, dividing by the within-training-window standard deviation of each RFF. I first ignore this standardization and will incorporate it in the next step.

Rahimi and Recht (2007) show that RFF can approximate kernels. Sutherland and Schneider (2015) show similar results for the RFF specification used by KMZ. The analysis in these papers focuses on the case where a kernel $k(\boldsymbol{u}, \boldsymbol{v}) = k(\boldsymbol{u} - \boldsymbol{v})$ is defined on $\mathbb{R}^K$ and $P < K$ RFF are used to approximate the kernel. This is the typical use-case for RFF. The idea is to reduce computational complexity relative to direct computation of kernels. In contrast, in KMZ's setting, the number of RFF is much larger than the dimension of $\boldsymbol{x}_t$. As a consequence, inner products of RFF virtually perfectly approximate a kernel

$$\boldsymbol{z}_t' \boldsymbol{z}_{t-k} \approx k(\boldsymbol{x}_t, \boldsymbol{x}_{t-k}), \tag{5}$$

which, given the standard normal distribution of the weights in KMZ's construction of the RFF, is a Gaussian kernel

$$k(\boldsymbol{x}_t, \boldsymbol{x}_{t-k}) = \exp\left(-\frac{\gamma^2}{2} \|\boldsymbol{x}_t - \boldsymbol{x}_{t-k}\|_2^2\right), \tag{6}$$

as shown in Sutherland and Schneider (2015).

With the notation

$$k(\boldsymbol{x}_t, \boldsymbol{X}_{t-1}) = \Big(k(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}), \quad ..., \quad k(\boldsymbol{x}_t, \boldsymbol{x}_{t-T})\Big) \tag{7}$$

2. KMZ do not pre-multiply by $\sqrt{2/P}$, but this is inconsequential for the resulting portfolio weights of the market timing strategy as this scalar factor cancels out in the weights (3).

and

$$K(\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-1}) = \begin{pmatrix} k(\boldsymbol{x}_{t-1}, \boldsymbol{X}_{t-1}) \\ ..., \\ k(\boldsymbol{x}_{T}, \boldsymbol{X}_{t-1}) \end{pmatrix} \tag{8}$$

we then get

$$\boldsymbol{z}_t' \boldsymbol{Z}_{t-1}' \approx k(\boldsymbol{x}_t, \boldsymbol{X}_{t-1}), \qquad \boldsymbol{Z}_{t-1} \boldsymbol{Z}_{t-1}' \approx K(\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-1}) \tag{9}$$

which leads to the key result that the ridgeless regression with a very large number $P > T$ of RFF becomes a ridgeless limit case of a kernel ridge regression where the predicted value is

$$\hat{y}_{t+1|t} \approx k(\boldsymbol{x}_t, \boldsymbol{X}_{t-1}) K(\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-1})^{-1} \boldsymbol{y}_t. \tag{10}$$

The kernels involved only take a small number of $K = 15$ inputs.[3] Moreover, the effective number of parameters in the construction of the predicted return is only $T = 12$.[4] Applied to training data sets of such short length, this is effectively a low-complexity regression, despite the seemingly high complexity suggested by the large number of RFF employed by KMZ. The complexity of the regression is capped by the number of observations in the training window.

The kernel form of the estimator in (10) is also revealing about why a market-timing strategy based on this estimator earns high returns out-of-sample. The estimator constructs a prediction of future returns by smoothing past returns. The predicted value in (2) is a weighted average of the T lagged returns in $\boldsymbol{y}_t$, where the weights depend simply on the similarity between $\boldsymbol{x}_t$ and the T columns of $\boldsymbol{X}$. If among these columns there are $\boldsymbol{x}_{t-k}$ for some lags k that are close in distance to $\boldsymbol{x}_t$, then prediction $\hat{y}_{t+1|t}$ is close to an average of the y_{t-k+1} at those lags k.

There are now two effects that largely account for how the weights on past returns look like. First, the less distant $\boldsymbol{x}_{t-k}$ from $\boldsymbol{x}_t$, the higher the weight that y_{t-k} gets in the construction of $\hat{y}_{t+1|t}$. If $\boldsymbol{x}$ follows an autoregressive process, one would expect higher weights for $\boldsymbol{x}_{t-k}$ that are

3. I refer to this regression as a kernel ridgeless regression, which is distinct from a kernel regression, say based on the Nadarya-Watson estimator, that simply weights observations of the dependent variable without involving the $K(.,.)^{-1}$ matrix.

4. See Hastie, Tibshirani, and Friedman (2009), chapter 7.6 for a definition of the effective number of parameters, or degrees of freedom, as the trace of the "hat-matrix" and Rasmussen and Williams (2006), chapter 2.6., for an application to Gaussian process regression, which is mathematically similar to kernel ridge regression. In this case here, the "hat-matrix" is the trace of $K(\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-1}) K(\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-1})^{-1} = \boldsymbol{I}$, which is equal to T. See also Bach (2024) for related asymptotic results in the case where $P, T \to \infty$ with P/T fixed, and the degrees of freedom in the ridgeless case are equal to T.

closer in time to $\boldsymbol{x}_t$. This makes the weights look like those of a momentum strategy. The predicted value will tend to be an average of lagged returns in nearby months. Second, in times when the $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-k}$ are subject to bigger noise shocks—which will tend to happen during periods of high volatility—they will differ more, and hence $k(\boldsymbol{x}_t, \boldsymbol{x}_{t-k})$ will be smaller, resulting in a smaller weight on y_{t-k}. The combination of the two effects is basically a momentum strategy interacted with a volatility-timing strategy that makes the momentum strategy less aggressive when volatility is high.

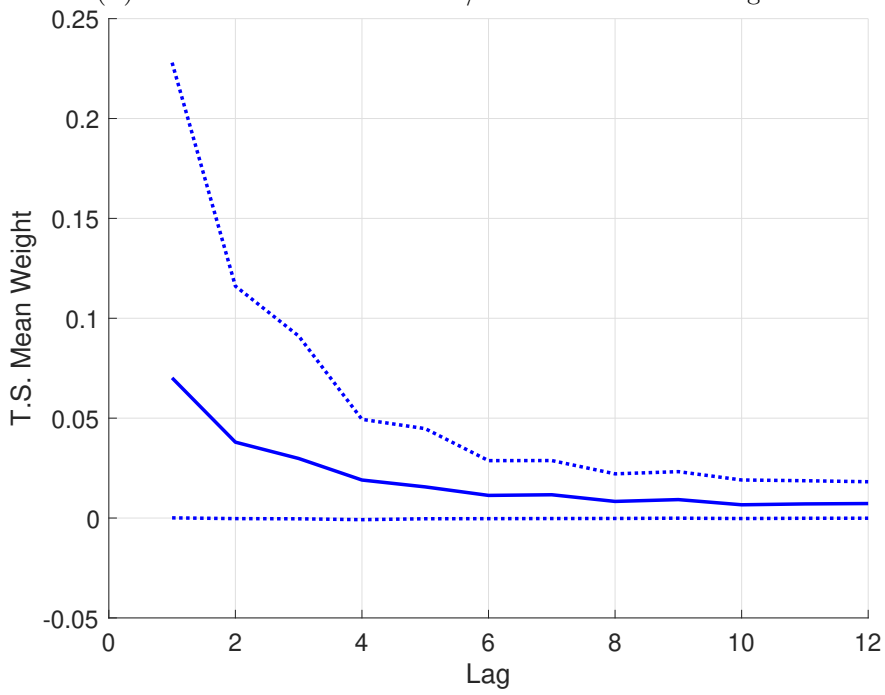
To illustrate how the weights on past returns look like, I use the replication data provided by KMZ to construct their estimator and the return predictions that follow. Figure I shows the weights averaged over 1,000 draws of random weights for 12,000 RFF and focusing on the ridgeless regression case with $T = 12$. The percentiles are based on the distribution over time of the averaged weights across these 1,000 draws of RFF.[5]

Panel A shows the time-series mean of the weights for the different return lags as well as the 10th and 90th percentiles. The results show that the ridgeless regression basically forms a momentum strategy with the highest weight on the most recent lagged return y_t and smaller weights on earlier return observations. This can be anticipated from the kernel ridge regression representation of the return prediction in (10). The weights depend on how similar the low-dimensional predictor vector $\boldsymbol{x}_t$ is to the predictor vectors $\boldsymbol{x}_{t-k}$ realized earlier in the time window of length $T = 12$. Naturally, predictor vectors that are closer in time are more similar, hence their distance to $\boldsymbol{x}_t$ is smaller, which results in a bigger weight for returns that are closer in time to t in the kernel ridgeless regression forecast in (10).

Panel B shows the cross-sectional mean of the $T = 12$ weights every month. There is strong time-variation in these weights. As I will show, this time-variation is closely related to the reciprocal of a volatility measure, which makes the return prediction one of a volatility-timed momentum strategy.

5. The variation of weights for different draws of RFF random weights is miniscule. Hence, just plotting the mean weights and percentiles for a single draw of RFF random weights would look almost identical.

(A) Time-Series Mean and 10th/90th Percentiles of Weights



(B) Cross-Sectional Means of Weights

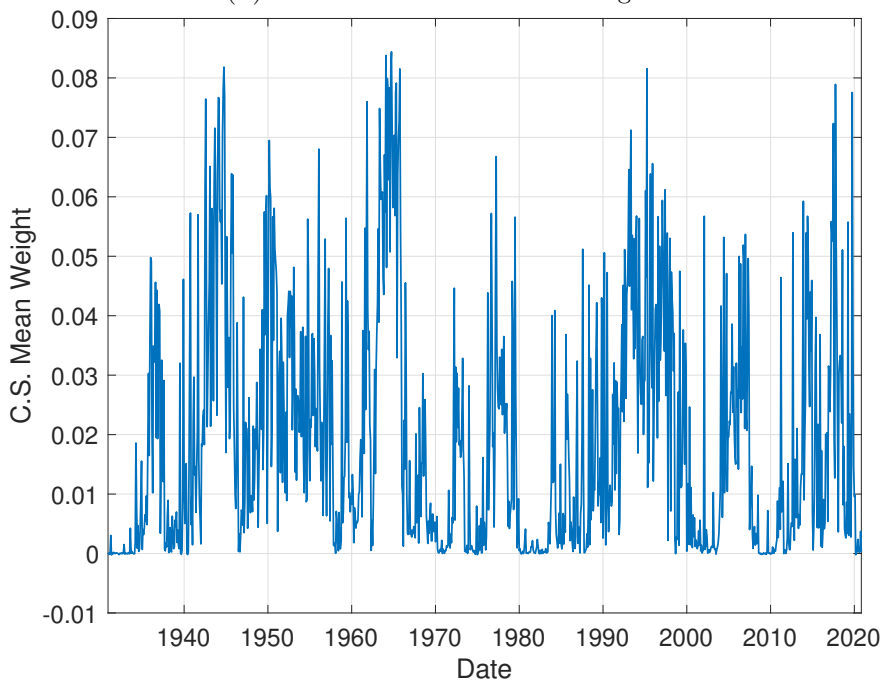


FIGURE I
Weights on T Past Returns in Ridgeless Regression Return Prediction

II.D. Within-Window Standardization of Random Fourier Leads to a Scaled Version of the Forecasts from Kernel Ridgeless Regression

One issue that still remains to be addressed is that KMZ do not use the RFF directly as predictors. Instead, within each regression time window, they standardize the RFF by dividing with the within-window standard deviation of each RFF. Let $\tilde{\boldsymbol{z}}$ denote the standardized RFF. Their inner product is

$$\tilde{\boldsymbol{z}}_t'\tilde{\boldsymbol{z}}_{t-k}' = \boldsymbol{z}_t'\boldsymbol{\Omega}_{t-1}^{-1}\boldsymbol{z}_{t-k}, \tag{11}$$

where $\boldsymbol{\Omega}_{t-1} = \frac{1}{T-1}\,\text{diag}(\boldsymbol{Z}_{t-1}'\boldsymbol{Z}_{t-1})$ is a diagonal matrix with P within-window variances the RFF on its diagonal. The results from Rahimi and Recht (2007) and Sutherland and Schneider (2015) on approximating kernels do not apply here anymore for this weighted inner product of RFF.

While I do not have a closed-form result that relates the inner products of standardized RFF to kernels, empirically it turns out that in KMZ's setting the return prediction based on the standardized RFF is extremely well approximated by a simple scaling up of the kernel in (10) to

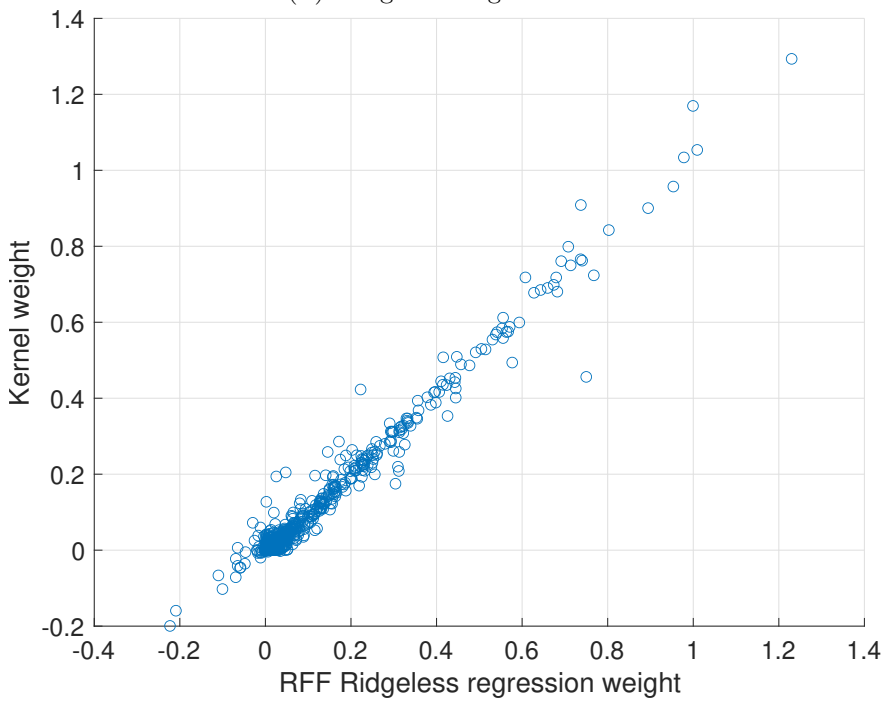
$$\hat{y}_{t+1|t}^{\text{kernel}} \approx 1.69 \times k(\boldsymbol{x}_t, \boldsymbol{X}_{t-1})K(\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-1})^{-1}\boldsymbol{y}_t \tag{12}$$

I refer to the market timing strategy based on this scaled version of the kernel ridge regression as the kernel approach.

For the same RFF as those in Figure I, Figure II shows that the weights on past returns implied by KMZ's high-complexity regression with $P = 12,000$ RFF are almost exactly the same as those resulting from the low-dimensional kernel approach in (12). Panel A shows the weights for the most recent lagged return y_t. As this weight is typically the biggest, it plays the biggest role in the prediction of return in month $t+1$. As Panel A shows, the weights on y_t in the rolling windows implied by the high-complexity ridgeless regression (horizontal axis) are almost the same as those implied by the low-complexity kernel approach (vertical axis). The correlation is 0.99. Panel B shows similar results for the lag 2 return, i.e., the weight on y_{t-1}. For more distant lags, the correlations between the different versions of weights are extremely high, too. Therefore, the extremely high-dimensional regression ridgeless using RFF collapses to a simple low-complexity kernel regression.

Figure III reinforces this conclusion. It shows the predicted returns, $\hat{y}_{t+1|t}^{\text{complex}}$ produced by the

(A) Weight for lag 1 return



(B) Weight for lag 2 return

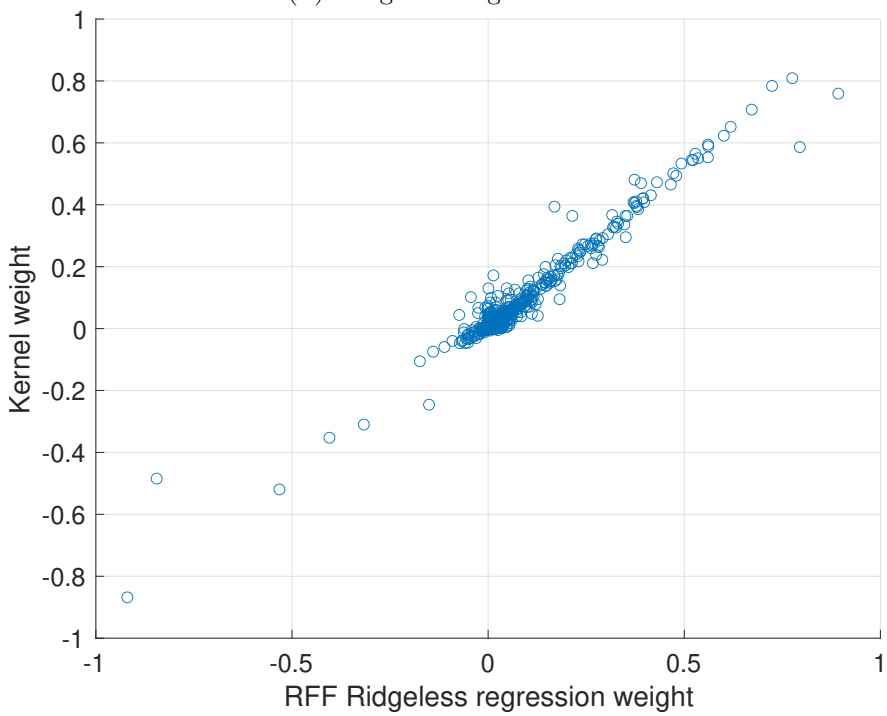


FIGURE II
Weights on Past Returns implied by Ridgeless Regression and Scaled Kernel Ridgeless Regression

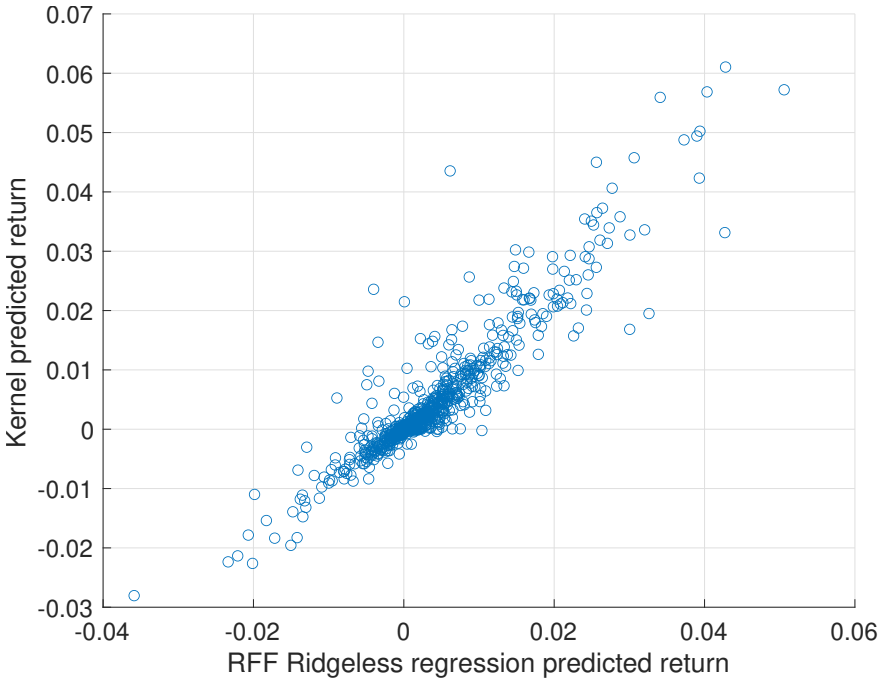


FIGURE III
Predicted Returns Implied by KMZ's Ridgeless Regression and Kernel Approach

high-complexity regression and $\hat{y}_{t+1|t}^{\text{kernel}}$ produced by the kernel approach. They are very similar. The correlation is 0.93.

II.E. Market-Timing Based on Gaussian Kernel Ridgeless Regression: Effectively A Volatility-Timed Momentum Strategy

That the high-complexity ridgeless regression collapses to a low-complexity kernel approach makes clear that the KMZ strategy's success in predicting returns out-of-sample based on estimates from extremely short rolling time windows does not come from the regression's ability to extract predictive information that nonlinear functions of the dividend-price ratio and other predictors have with regards to future returns. Instead, the high-complexity regression collapses to a kernel expression where the predicted return is essentially a simple momentum strategy that forms a weighted average of $T = 12$ lags of past returns, albeit with time-varying weights.

The weight that the predicted return $y_{t+1|t}^{\text{kernel}}$ puts on return y_{t-k} depends, via the Gaussian kernel

$$k(\boldsymbol{x}_t, \boldsymbol{x}_{t-k}) = \exp\left(-\frac{\gamma^2}{2}\|\boldsymbol{x}_t - \boldsymbol{x}_{t-k}\|_2^2\right). \tag{13}$$

on the distance between $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-k}$. Broadly speaking, this distance reflects two effects. First,

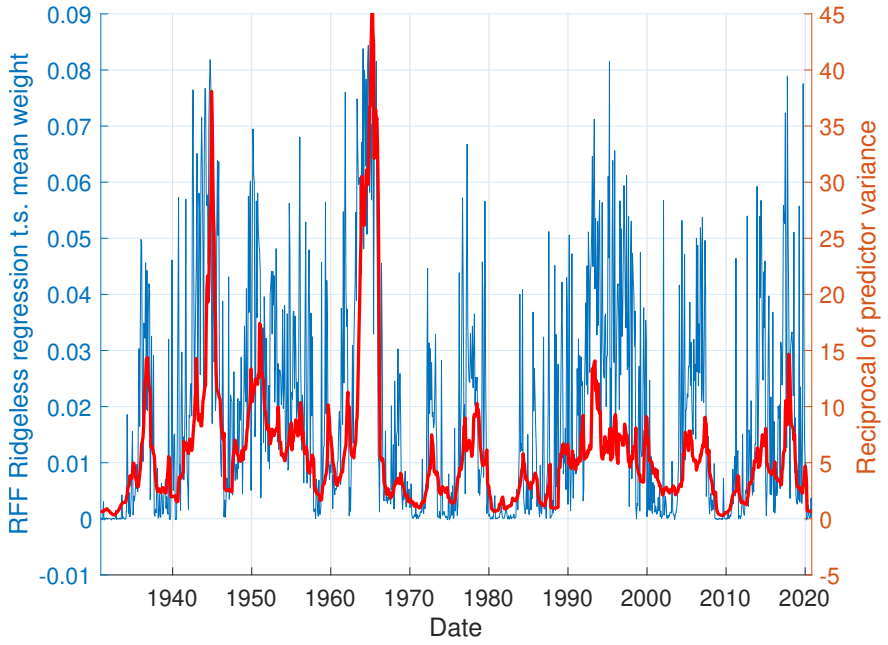


FIGURE IV
Ridgeless Regression Mean Weight on Past Returns and Reciprocal of Mean Predictor Variance

lags that are more distant in time (larger k) are less similar, which results in greater distance, and lower $k(\boldsymbol{x}_t, \boldsymbol{x}_{t-k})$. Second, the more volatile the predictors are within the time window of length T used to estimate the regression, the greater the distance between $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-k}$. The two effects combined produce a momentum strategy (higher positive weights for nearby lags of returns) that is volatility-timed (higher weights when predictor volatility is low and hence neighboring $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-k}$ are more similar).

Figure IV shows evidence of the negative relation between the magnitude of weights on past returns and predictor volatility. The figure shows the mean of the weights in (3) every month that the predicted value of KMZ's high-complexity regression using RFF places on the past 12 monthly lags of returns. This is the same time series as in Panel B of Figure I. For comparison, the figure also shows the reciprocal of the mean of the variance of the 15 predictor variables within each estimation window of $T = 12$ months. The comparison shows that weights on past returns produced by the ridgeless RFF strategy are highly correlated over time with the reciprocal mean predictor variance, consistent with the reasoning I outlined above: In windows in which the 15 predictors are less volatile, the predictor vectors $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-k}$ are at smaller distance, which results in higher weights assigned by the kernel (which in turn is approximated almost perfectly by the RFF).

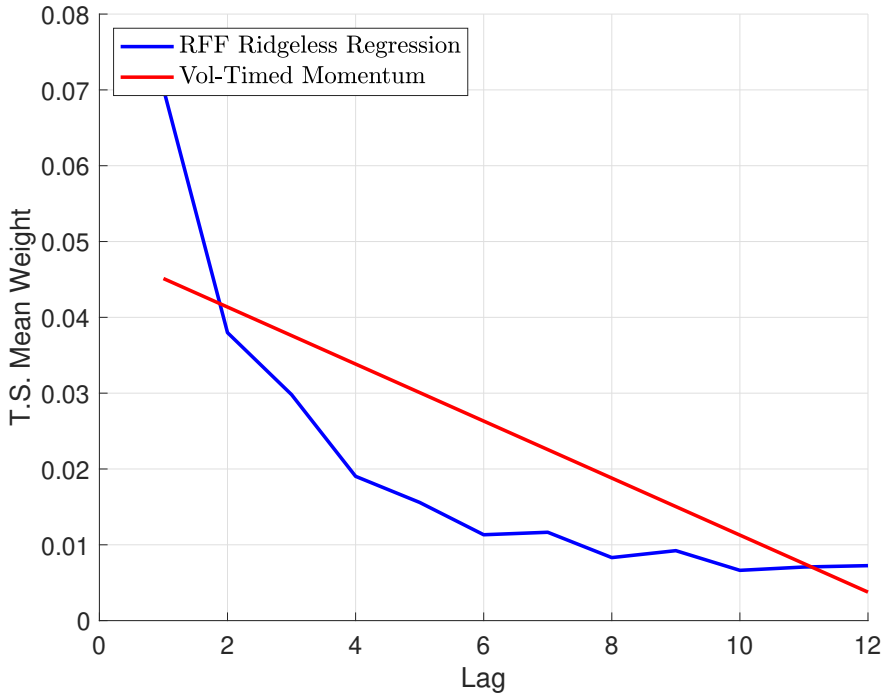


FIGURE V
Mean Weight on Past Returns in High-Complexity Ridgeless Regression with RFF and in the Volatility-Timed Momentum Strategy

These results suggest that it should be possible to approximate KMZ's market timing strategy with a very simple one: A momentum strategy that has weights that decay with lag length, similar to the decay of the time-series mean of weights shown in Panel A of Figure I, combined with a volatility-timing scaling factor that increases the magnitude of weights on past returns if the volatility of the 15 predictor variables is low. The following market timing strategy implements this idea:

$$\hat{y}^{\text{volmom}}_{t+1|t} = 0.05 \times \frac{1}{\hat{\sigma}^2_{x,t-1}} \times \sum_{k=0}^{11} \frac{12-k}{78} y_{t-k+1}. \tag{14}$$

Here $\hat{\sigma}^2_{x,t-1}$ is the average variance of the predictors in the training window of length $T = 12$. The weights on the past returns in the summation term are linearly declining and the division by 78 scales them to have a sum of unity. The initial multiplication by 0.05 makes the average magnitude of $\hat{y}^{\text{volmom}}_{t+1|t}$ similar to the average magnitude of the high-complexity strategy weights $\hat{y}^{\text{complex}}_{t+1|t}$, but this scaling has no effect on t-statistics and information ratios of the out-of-sample returns of the strategy.

Figure V compares the linearly declining weights of the volatility-timed momentum strategy with the weights implied by the high-complexity strategy based on RFF. They are not identical,

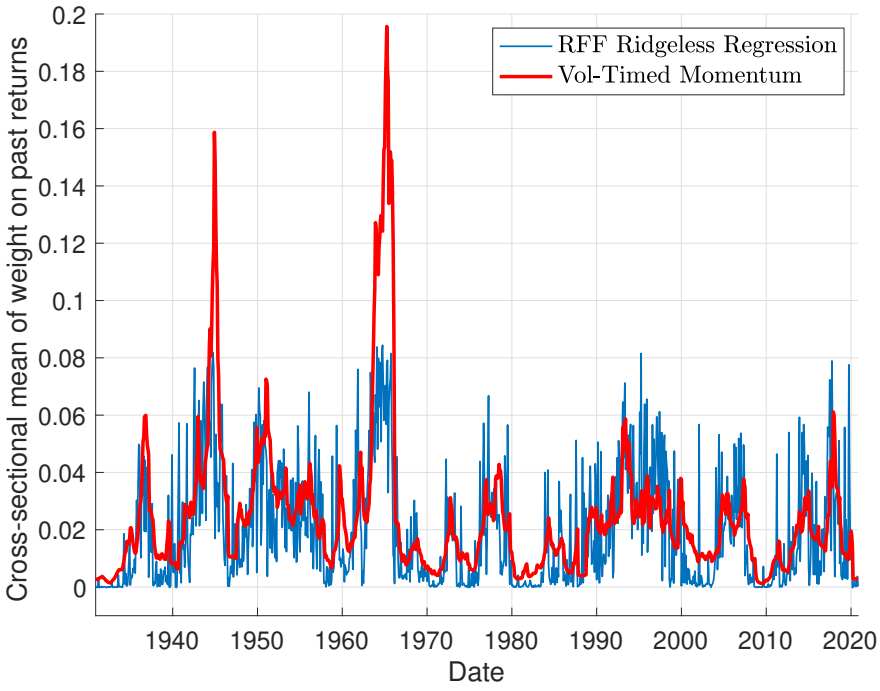


FIGURE VI
Mean Weight on Past Returns in High-Complexity Ridgeless Regression with RFF and Volatility-Timed Momentum Strategy

but the broad pattern of decline with lag length is similar.

Figure VI shows the cross-sectional means of the past-return weights of the two strategies each month. Much of the time-series variation is shared between the two series.

II.F. Spanning Tests: Any Virtue in Complexity?

Does the high-complexity approach with RFF provide any incremental predictive benefit relative to the lower-complexity kernel and volatility-timed momentum approaches? Table I presents the evidence.

Panel A reports the alpha of each strategy relative to a one-factor model with the excess return of the CRSP value-weighted index as the single factor, as well as the corresponding t-statistics and information ratios. The first column shows the result from KMZ that the high-complexity RFF strategy produces positive alpha, with high t-statistic (2.417) and information ratio (0.255).[6] Interestingly, the low-complexity kernel strategy and the volatility-timed momentum

6. The t-statistic and information ratio are slightly lower than those that KMZ report in their Figure 8 for the ridgeless case ($\log_{10} z = -3$) and high complexity ($c = 1000$). The reason for this discrepancy is that I do not standardize the variable to be predicted to avoid the bias discussed in Appendix A. If I standardize returns in the same way as KMZ do, I obtain a t-statistic of 2.811 and an information ratio of 0.296, which exactly matches the results in KMZ's Figure 8.

TABLE I
Out-of-Sample Market Timing Performance

The high-dimensional RFF market timing strategy in the first column uses RFF as predictors in (10) and it is the same as in the ridgeless regression case in KMZ with $T = 12$ and $P = 12,000$ RFF, but without standardizing the predicted return, for the reasons discussed in Appendix A. The abnormal returns of the market timing strategy based on high-complexity ridgeless regression are averaged over 1,000 draw of random weights in the construction of RFF. The second column shows the market timing strategy based on the low-dimensional Gaussian kernel ridgeless in (12). The third column shows the volatility-timed momentum strategy in (14). Panel A shows alphas and information ratios relative to a one-factor model with the monthly excess return of the CRSP value-weighted index as the single factor. Panel B uses a two-factor model with the CRSP value-weighted index and the return on the low-complexity kernel strategy as the two factors. Panel C uses the volatility-timed momentum strategy as the second factor along with the CRSP value-weighted index. Alphas are annualized in percent.

	High-Complexity RFF	Low-Complexity Kernel	Vol-Timed Momentum
	Panel A: One-Factor α (Market Factor)		
Alpha	0.034	0.040	0.034
(t-stat.)	(2.417)	(2.900)	(3.684)
Information Ratio	0.255	0.306	0.388
	Panel B: Two-Factor α (Market and Kernel Factors)		
Alpha	-0.001		
(t-stat.)	(-0.122)		
Information Ratio	-0.013		
	Panel C: Two-Factor α (Market and Vol-Timed Momentum Factors)		
Alpha	0.012		
(t-stat.)	(0.945)		
Information Ratio	0.100		

strategy produce t-statistics and information ratios that are even slightly larger than those of the high-complexity RFF strategy. This suggests that the high-complexity approach with RFF may not provide any incremental predictive benefits.

This is further reinforced by the spanning test in Panel B. Here I add the kernel strategy return as a second factor to the market factor. The alpha of the high-complexity RFF strategy drops to almost exactly zero, and the t-statistic and information ratios become very small. This is a natural outcome, given how similar the market timing positions of the two strategies are. Panel A in Figure VII shows that every month, the low-complexity kernel strategy takes almost exactly the same position as the high-complexity RFF strategy.

Panel C uses the volatility-timed momentum strategy as second factor. Here the alpha doesn't drop as much, but it still falls by about 2/3 compared with Panel A and the t-statistic falls far below conventional levels of significance. Relatedly, Panel B of Figure VII shows that the market

timing positions of the volatility-timed momentum strategy are very close to those of the high-complexity RFF strategy. This suggests that the volatility-timed momentum strategy, despite its extreme simplicity, captures much of the predictive power of the high-complexity RFF strategy.

Furthermore, the time-series of the kernel strategy and volatility-timed momentum strategy market timing positions shown in Figure VII also share the property of the positions of the high-complexity RFF strategy that they are higher outside of recessions. KMZ interpret this property of the high-complexity RFF strategy as "the machine learning strategy learns to divest leading up to recessions." This is not the correct interpretation. Instead, the business-cycle pattern of the high-complexity RFF strategy positions simply reflects the fact that its market timing position is effectively a weighted average of past returns, with positive weights on past returns that are bigger in times when the predictor variables are less volatile. These periods of low predictor volatility are outside of recessions.

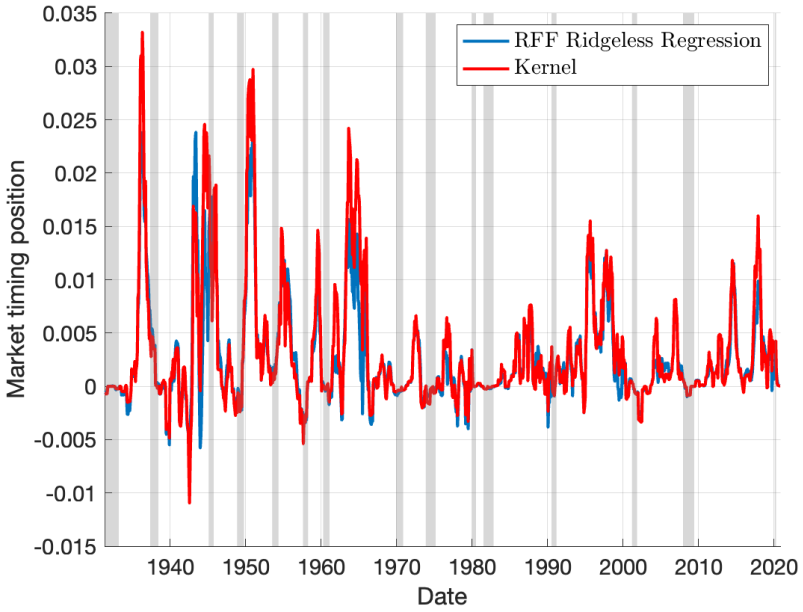
Figure VIII shows the cumulated abnormal returns of the three strategies with abnormal returns measured relative to the one-factor model with the market factor. The cumulated abnormal returns are divided by the full-sample standard deviation, which makes them interpretable as proportional to a cumulated information ratio.

II.G. Out-of-Sample Market Timing Performance in Artificial Data with Reversals

The collapse of the high-complexity RFF-based ridgeless regression to a kernel regression leads to weights on recent past returns that are, by construction, always positive (aside from small noise distortions due to approximation error where the RFF do not perfectly approximate a Gaussian kernel.) Hence, the ridgeless regression does not learn from the data that there is a volatility-dependent momentum effect; it always constructs a volatility-timed momentum strategy, irrespective of the properties of the return data. The good out-of-sample performance is therefore due to a coincidence that a volatility-timed momentum effect is actually present in the data and not because the approach extracts predictive relationships from the data.

While the math is clear on this point, it may nevertheless be useful to illustrate this empirically. I modify the return data to such that it exhibits short-term reversals instead of momentum by adding an MA(2) process with negative autocorrelation to the original returns y_t. The artificial

(A) Comparison with Low-Complexity Kernel Strategy



(B) Comparison with Volatility-Timed Momentum Strategy

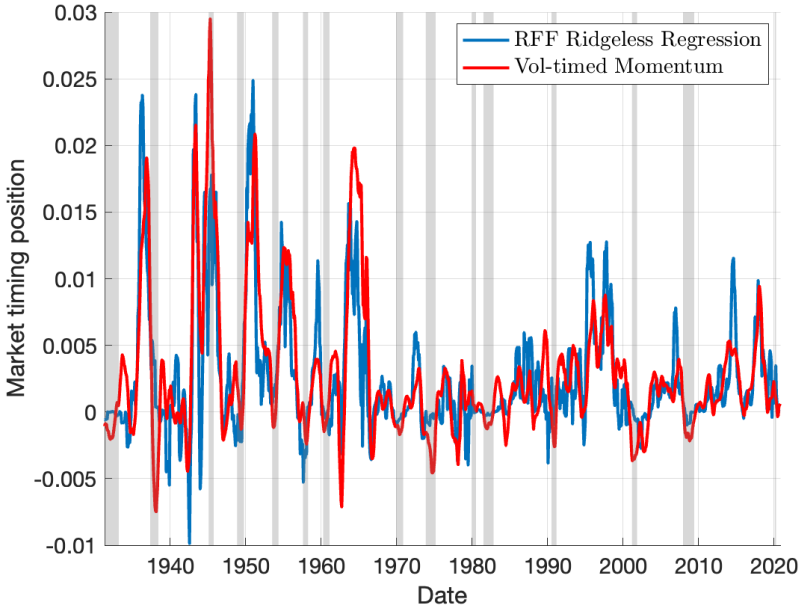


FIGURE VII
Market Timing Positions of High-Complexity Ridgeless Regression, Kernel, and Volatility-Timed Momentum Strategy

Six-month moving averages of market timing positions $y_{t+1|t}^{\text{Complex}}$, $y_{t+1|t}^{\text{Kernel}}$, and $y_{t+1|t}^{\text{Volmom}}$ with $T = 12$, $P = 12,000$. The market timing positions $y_{t+1|t}^{\text{Complex}}$ are averaged over 1,000 draws of random weights in the construction of RFF.

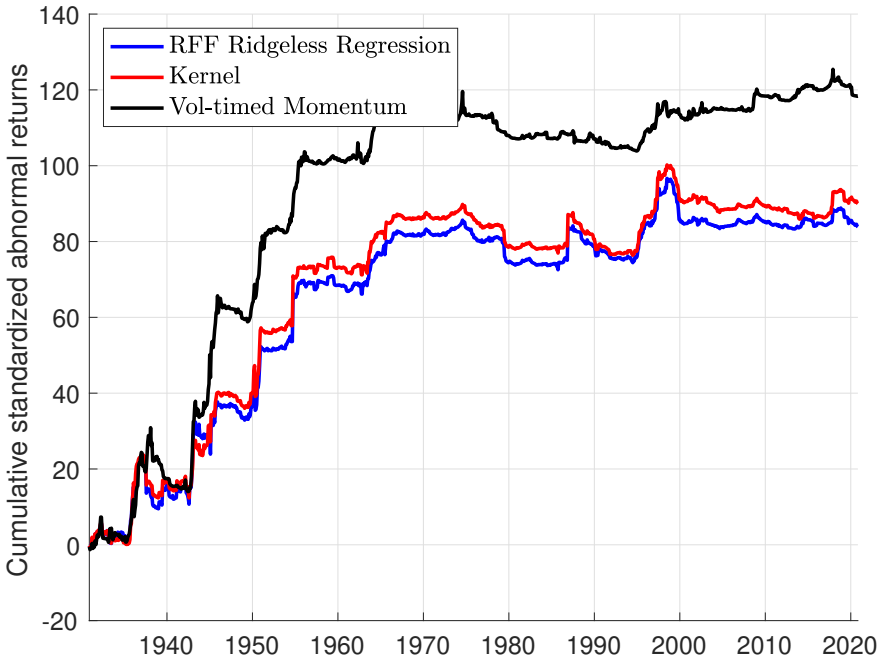


FIGURE VIII
Cumulative Standardized Out-of-Sample Abnormal Returns

Abnormal returns relative to one-factor model with excess return of CRSP value-weighted index as single factor, divided by full-sample standard deviation of the abnormal return. The abnormal returns of the market timing strategy based on high-complexity ridgeless regression with $T = 12$ and $P = 12,000$ RFF are averaged over 1,000 draw of random weights in the construction of RFF.

data of monthly market index returns then is

$$\tilde{y}_t = y_t + \xi_t - \theta_1 \xi_{t-1} - \theta_2 \xi_{t-2} \tag{15}$$

with $\theta_1 = \theta_2 = 0.2$ and where $\xi_t \sim \mathcal{N}(0, 0.01)$. I chose the parameters of the MA(2) process such that the negative autocorrelation is big enough to overcome the momentum effect in the actual market index returns. I then replace the market index returns in KMZ's data with this artificial return series and redo the estimation.

Table II reproduces the analysis of Table I, but now with the artificial market index return data. As expected, the abnormal performance of the high-complexity RFF-based strategy, as well as the kernel strategy and the volatility-timed momentum strategy are now all negative. This demonstrates that the high-complexity RFF-based strategy's weights on recent past returns are always positive. It does not learn from the data that there is a momentum effect (in the original data) or reversal effect (now here in the artificial data). All that matters for how the weights look like is the predictor-vector similarity. Despite the fact that the return data now features reversal

TABLE II

Out-of-Sample Market Timing Performance in Artificial Data with Reversals

The high-dimensional RFF market timing strategy in the first column uses RFF as predictors in (10) and it is the same as in the ridgeless regression case in KMZ with $T = 12$ and $P = 12,000$ RFF, but without standardizing the predicted return, for the reasons discussed in Appendix A. The abnormal returns of the market timing strategy based on high-complexity ridgeless regression are averaged over 1,000 draw of random weights in the construction of RFF. The second column shows the market timing strategy based on the low-dimensional Gaussian kernel ridgeless in (12). The third column shows the volatility-timed momentum strategy in (14). Panel A shows alphas and information ratios relative to a one-factor model with the monthly excess return of the CRSP value-weighted index as the single factor. Panel B uses a two-factor model with the CRSP value-weighted index and the return on the low-complexity kernel strategy as the two factors. Panel C uses the volatility-timed momentum strategy as the second factor along with the CRSP value-weighted index. Alphas are annualized in percent.

	High-Complexity RFF	Low-Complexity Kernel	Vol-Timed Momentum
	Panel A: One-Factor α (Market Factor)		
Alpha	-0.143	-0.137	-0.124
(t-stat.)	(-1.835)	(-1.757)	(-4.099)
Info Ratio	-0.193	-0.185	-0.432
	Panel B: Two-Factor α (Market and Kernel Factors)		
Alpha	-0.013		
(t-stat.)	(-0.517)		
Information Ratio	-0.055		
	Panel C: Two-Factor α (Market and Vol-Timed Momentum Factors)		
Alpha	-0.045		
(t-stat.)	(-0.628)		
Information Ratio	-0.066		

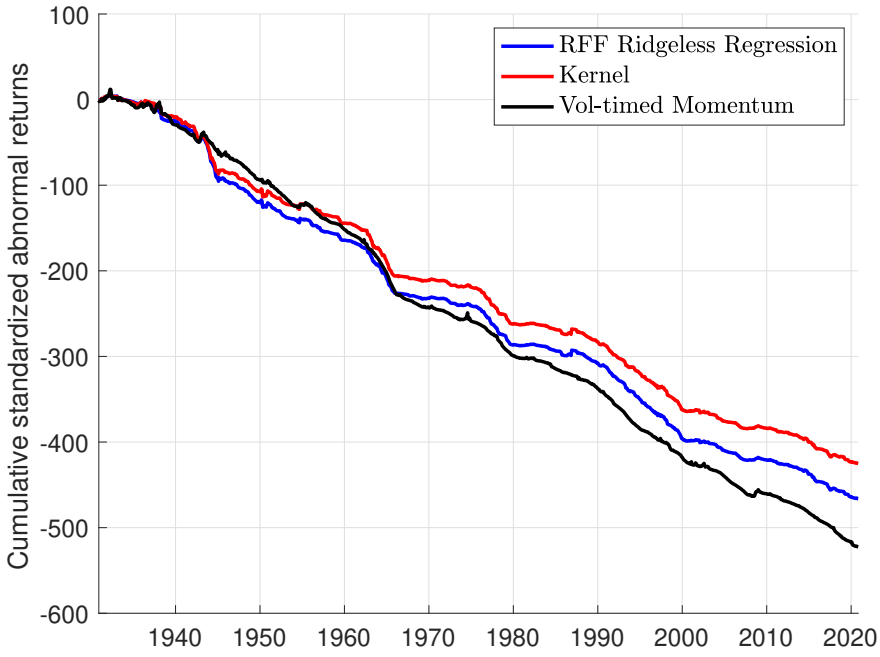


FIGURE IX

Cumulative Standardized Out-of-Sample Abnormal Returns in Artificial Data with Reversals

Abnormal returns relative to one-factor model with excess return of CRSP value-weighted index as single factor, divided by full-sample standard deviation of the abnormal return. The abnormal returns of the market timing strategy based on high-complexity ridgeless regression with $T = 12$ and $P = 12,000$ RFF are averaged over 1,000 draw of random weights in the construction of RFF.

effects, it is still the case that the predictor vectors in the most recent months are most similar to the current predictor vector, and hence the most recent past returns get a positive weight.

Figure IX illustrates the out-of-sample performance in cumulative terms. In sharp contrast to Figure IX, the cumulative abnormal performance is negative. This reflects the inability of the high-complexity RFF-based ridgeless regression, and the kernel approach that it approximates, to learn about the reversals present in the data in short training windows of only 12 months. These approaches still mechanically produce a volatility-timed momentum strategy, which performs badly in this artificial data.

III. Cross-Sectional Asset Pricing

I focused above on understanding the time-series predictability results in KMZ, as these are extremely puzzling given the conventional wisdom about predictive regressions for stock market index returns. However, the results above also shed light on what happens when large number of RFF-based factors are employed in a cross-sectional asset pricing setting.

Consider now an unbalanced panel setting as in Didisheim, Ke, Kelly, and Malamud (2024) (DKKM), with $n = 1, ..., N_s$ stocks at time s, each with a vector of $j = 1, ..., J$ characteristics $\boldsymbol{x}_{n,s}$, stacked into a $N_s \times J$ matrix $\boldsymbol{X}_s = (\boldsymbol{x}_{1,s}, ..., \boldsymbol{x}_{N_s,s})'$. The researcher uses a training sample of length T, with $T < N_s$ for all s.

Construct RFF based on these characteristics for each stock n as

$$\begin{pmatrix} z_{n,i,s} \\ z_{n,i+1,s} \end{pmatrix} = \sqrt{\frac{2}{P}} \begin{pmatrix} \cos(\gamma \boldsymbol{\omega}_i' \boldsymbol{x}_{n,s}) \\ \sin(\gamma \omega_{i+1}' \boldsymbol{x}_{n,s}) \end{pmatrix} \quad \boldsymbol{\omega}_i \sim \text{ IID } \mathcal{N}(0, \boldsymbol{I}), \quad i = 1, ..., P/2 \tag{16}$$

DKKM randomly generate different values for γ for each i from a grid $[0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$. Here I assume that γ is non-random. Then, in each cross-section s, place the RFF in a $P \times N$ matrix $\boldsymbol{Z}_s$.

Let $\boldsymbol{r}_{s+1}$ be an N_s-dimensional vector of stock returns. Forming cross-products of returns and lagged RFF delivers a P-dimensional vector of RFF factors

$$\boldsymbol{f}_{s+1} = \boldsymbol{Z}_s \boldsymbol{r}_{s+1}. \tag{17}$$

As in DKKM, I assume that the SDF has a representation as

$$M_s = 1 - \boldsymbol{\lambda}' \boldsymbol{f}_s. \tag{18}$$

Solving for the minimum-norm solution of the sample moment conditions $\hat{E}[\boldsymbol{f}_s M_s] = 0$, and letting $\hat{E}[.]$ denote a sample average in the training data sample from $s = t - T + 1, ..., t$, delivers the ridgeless estimator of the prices of risk

$$\begin{aligned} \hat{\boldsymbol{\lambda}}_t &= \left(\hat{E}[\boldsymbol{f}_s \boldsymbol{f}_s'] \right)^+ \hat{E}[\boldsymbol{f}_s] \\ &= \left(\boldsymbol{F}_t' \boldsymbol{F} \right)^+ \boldsymbol{F}_t' \boldsymbol{\iota}, \end{aligned} \tag{19}$$

where $\boldsymbol{\iota}$ is a conformable vector of ones and $\boldsymbol{F}_t = (\boldsymbol{f}_{t-T+1}, ..., \boldsymbol{f}_t)'$ has dimension $T \times P$.

Using the same approach as in (3), I can write $\hat{\boldsymbol{\lambda}}_t$ as

$$\hat{\boldsymbol{\lambda}}_t = \boldsymbol{F}_t' \left(\boldsymbol{F}_t \boldsymbol{F}_t' \right)^{-1} \boldsymbol{\iota}. \tag{20}$$

In Section 4.5 in the theory part of their paper, DKKM discuss that the properties of the very large $P \times P$ matrix $\boldsymbol{F}_t'\boldsymbol{F}$ are difficult to characterize. However, in the ridgeless case, calculation of this matrix is not necessary—all we need is the much smaller $T \times T$ matrix $\boldsymbol{F}_t\boldsymbol{F}_t'$. Even though the length of $\hat{\boldsymbol{\lambda}}_t$ is P, the effective number of parameters is only T. As in the time-series setting earlier, the size of the training data set caps the effective number of parameters.[7]

The prices of risk vector $\boldsymbol{\lambda}$ is proportional to the weights of the mean-variance efficient combination of the RFF factors. The estimated implied mean-variance efficient portfolio weights for the underlying N_t stocks, which can be applied out-of-sample to $\boldsymbol{r}_{t+1}$, are therefore proportional to

$$\boldsymbol{\omega}_t = \boldsymbol{Z}_t'\hat{\boldsymbol{\lambda}}_t = \boldsymbol{Z}_t'\boldsymbol{F}_t'\left(\boldsymbol{F}_t\boldsymbol{F}_t'\right)^{-1}\boldsymbol{\iota}. \tag{21}$$

Now note that

$$\begin{aligned}\boldsymbol{Z}_t'\boldsymbol{F}_t' &= \left[\boldsymbol{Z}_t'\boldsymbol{Z}_{t-T}\boldsymbol{r}_{t-T+1}, \quad ... \quad , \boldsymbol{Z}_t'\boldsymbol{Z}_{t-1}\boldsymbol{r}_t\right] \\ &\approx \left[K(\boldsymbol{X}_t, \boldsymbol{X}_{t-T})\boldsymbol{r}_{t-T+1}, \quad ... \quad , K(\boldsymbol{X}_t, \boldsymbol{X}_{t-1})\boldsymbol{r}_t\right],\end{aligned} \tag{22}$$

where $K(.,.)$ again denotes a Gaussian kernel matrix, as earlier, and the approximation follows for the same reasons I discussed earlier in Section II.C.

The $T \times T$ matrix $\boldsymbol{F}_t\boldsymbol{F}_t'$ has on its diagonal $\boldsymbol{f}_s'\boldsymbol{f}_s$, for $s = t-T+1, ..., t$. These inner products have the following approximation, again for the reasons discussed earlier in Section II.C:

$$\begin{aligned}\boldsymbol{f}_s'\boldsymbol{f}_s &= \boldsymbol{r}_s'\boldsymbol{Z}_{s-1}'\boldsymbol{Z}_{s-1}\boldsymbol{r}_s \\ &\approx \boldsymbol{r}_s'K(\boldsymbol{X}_{s-1}, \boldsymbol{X}_{s-1})\boldsymbol{r}_s.\end{aligned} \tag{23}$$

Off-diagonal elements are

$$\begin{aligned}\boldsymbol{f}_s'\boldsymbol{f}_{s-k} &= \boldsymbol{r}_s'\boldsymbol{Z}_{s-1}'\boldsymbol{Z}_{s-1-k}\boldsymbol{r}_{s-k} \\ &\approx \boldsymbol{r}_s'K(\boldsymbol{X}_{s-1}, \boldsymbol{X}_{s-1-k})\boldsymbol{r}_{s-k}\end{aligned} \tag{24}$$

7. In the case $P < T$, we would have $\hat{\boldsymbol{\lambda}}_t = \left(\boldsymbol{F}_t'\boldsymbol{F}\right)^{-1}\boldsymbol{F}_t'\boldsymbol{\iota}$, i.e., the prices of risk are slopes in a regression of a vector of ones on the factors, with effective number of parameters $\text{tr}\left[\boldsymbol{F}\left(\boldsymbol{F}_t'\boldsymbol{F}\right)^{-1}\boldsymbol{F}_t'\right] = P$, but in the $P > T$ case, it is $\text{tr}\left[\boldsymbol{F}\boldsymbol{F}'\left(\boldsymbol{F}_t\boldsymbol{F}'\right)^{-1}\right] = T$. These T effective parameters are sufficient to fit the vector $\boldsymbol{\iota}$ perfectly each period, which makes the in-sample SDF equal to exactly zero each period.

i.e., a kernel-weighted sum of serial and cross-serial comoments of returns at lag k. As serial correlation of returns is very small, and squared expected returns are small relative to the second moments of returns, the matrix $\boldsymbol{F}_t\boldsymbol{F}_t'$ can be approximated by setting the off-diagonal elements to zero.

With this approximation and (22), the weight vector in (20) becomes

$$\boldsymbol{\omega}_t \approx \sum_{s=t-T+1}^{t} \frac{K(\boldsymbol{X}_t, \boldsymbol{X}_{s-1})\boldsymbol{r}_s}{\boldsymbol{r}_{s+1}' K(\boldsymbol{X}_s, \boldsymbol{X}_s)\boldsymbol{r}_{s+1}} \tag{25}$$

To understand what is going on, it is useful to focus on the element of the weight vector that applies to stock n,

$$\omega_{n,t} \approx \sum_{s=t-T+1}^{t} \frac{K(\boldsymbol{x}_{n,t}, \boldsymbol{X}_{s-1})\boldsymbol{r}_s}{\boldsymbol{r}_s' K(\boldsymbol{X}_{s-1}, \boldsymbol{X}_{s-1})\boldsymbol{r}_s}. \tag{26}$$

To determine the weight for stock n at the end of period t, this approach looks back at the whole panel of stock returns in the training data and constructs a weighted average of past returns of stocks. Consider one term in the summation. In the weighted average in the numerator, the kernel assigns higher weights to returns of stock observations with characteristics that are similar to $\boldsymbol{x}_{n,t}$. In other words, it's a standard kernel smoothing approach. The denominator introduces a (co-)variance timing effect on the weights. If the return of stock n is highly correlated with other stocks that have similar characteristics, then if $r_{n,s}$ is high, the denominator will also tend to be high because cross-products of $r_{n,s}$ with other stocks' returns will tend to be positive and high as well, and they will receive high weight, via the kernel matrix, in the calculation of the quadratic form in the denominator. In contrast, if a stock is uncorrelated with others, or has very different characteristics from others, then $r_{n,s}$ will not be downweighted by the denominator.

What if one used a similarly short training data window, say $T = 12$ months, as in the time-series predictability analysis in KMZ? Then estimated mean-variance efficient portfolio weights would have a lot of similarity with the market timing positions in KMZ. In this case, the weight of stock n would be a weighted average of the returns in the entire panel of stock returns during the window with size $T = 12$, but with higher weight given to stocks that are similar to stock n in terms of their characteristics, and to time periods in which these stocks (and stock n itself) are more similar in characteristics to stock n at time t. As characteristics closer in time will be more similar, the weights will look like a momentum strategy, but with positive weights given not

only to stock n's past returns, but also to the returns of characteristics-similar stocks. Finally, the denominator in (26) introduces a (co)-variance timing element, giving higher weight to returns from time periods with lower (co)-variances. All in all, this would become a sort of volatility-timed momentum strategy, but with past returns smoothed across similar stocks rather than just using own-stock past returns.

When the training window is much larger, such as with $T = 360$ months in DKKM's analysis, then the strategy becomes more sophisticated. In this case, the strategy effectively looks back at a much longer history and whether there were times and stock observations with characteristics similar to $\boldsymbol{x}_{n,t}$ and then smoothes the returns associated with these observations to obtain the numerator of the portfolio weight in in (26). The denominator contributes (co)-variance weights, again using characteristics-similar observations in the past.

But even with $T = 360$, the complexity of the RFF-based strategy is much lower than it may seem. For example, take the case of $P = 360,000$, which is the highest value considered in the empirical analysis of DKKM. As (25) shows, the calculation of mean-variance optimal portfolio weights really boils down to just averaging returns and cross-products of returns of N_s stocks over $T = 360$ periods, with weights controlled by kernels that are a function of $K = 130$ mostly slowly-moving underlying characteristics.

IV. Conclusion

The empirical success in out-of-sample return prediction with a large number, P, of predictors constructed as Random Fourier Features (RFF)—randomly weighted and nonlinearly transformed versions of a small number, K, of original predictor variables—is suggestive of a virtue of complexity in return prediction. However, the actual complexity is much lower than it may seem. First, the effective number of parameters is capped by the number of time-series observations, T, in the training windows. This reflects the singularity that results from having $T < P$. Second, when P is much larger than K, inner products of RFF converge to Gaussian kernels that take the K original predictor variables as inputs. This makes ridgeless regression essentially equivalent to kernel ridgeless regression. Predicted returns are then fairly simple kernel-smoothed averages of past returns, where the magnitude of the weights is governed by the distance of past realizations of the K-dimensional original predictor vector from the current one. Third, the typical predictor

variables in return prediction applications are persistent. This makes the kernel ridgeless regression weights on past returns fairly similar for neighboring lags of past returns.

In the time-series setting of KMZ, the discrepancy between the seemingly high degree of complexity and the actual complexity is extreme. In their ridgeless case with $T = 12$ months, $K = 15$, and $P = 12,000$, the predicted return can be expressed as as a weighted average of 12 lagged returns with weights that depend on the distance of the 15 lagged predictor variables from the current predictor variable vector. The good out-of-sample performance of a market-timing strategy based on RFF-predicted returns therefore does not arise from predictive information extracted from the predictors—which would be surprising in a training window as short as 12 months—but rather from the fact that this weighted average of past returns resembles a volatility-timed momentum strategy that has historically performed well. The market-timing success of this strategy is not evidence of a virtue of complexity as actual complexity is not high to begin with.

Similar complexity-reducing effects are at work in cross-sectional asset pricing when the SDF is expressed as a function of P RFF factors constructed by weighting original asset returns with RFF constructed from K original firm characteristics. In this setting, too, the training window length T caps the effective number of risk price parameters in the SDF, when P is much larger than K the predicted mean-variance efficient portfolio weights for the original assets become kernel-smoothed averages of past returns in the panel of asset returns, and neighboring lags of past returns will tend to get similar weights due to predictor persistence.

More generally, these results suggest that the sources of empirical success of highly flexible, nonlinear approaches in asset pricing could potentially have less to do with complexity than it may seem. Small T, which is typical in asset pricing applications, should also restrict the effective number of parameters of other nonlinear approaches, not just RFF. The similarities between RFF and neural networks raise the possibility that similar convergence to low-complexity kernel functions may also take place for complex neural networks when T is small. This seems like an interesting area for future research on machine learning in asset pricing.

References

Bach, Francis, 2024, "High-Dimensional Analysis of Double Descent for Linear Regression With Random Projections," *SIAM Journal on Mathematics of Data Science* 6, 26–50.

Bartlett, Peter L, Philip M Long, Gábor Lugosi, and Alexander Tsigler, 2020, "Benign overfitting in linear regression," *Proceedings of the National Academy of Sciences* 117, 30063–30070.

Belkin, Mikhail, Daniel Hsu, Siyuan Ma, and Soumik Mandal, 2019, "Reconciling modern machine-learning practice and the classical bias–variance trade-off," *Proceedings of the National Academy of Sciences* 116, 15849–15854.

Didisheim, Antoine, Shikun Barry Ke, Bryan T Kelly, and Semyon Malamud, 2023, "Complexity in Factor Pricing Models," Working paper, National Bureau of Economic Research.

Didisheim, Antoine, Shikun Barry Ke, Bryan T Kelly, and Semyon Malamud, 2024, "APT or "AIPT"? The Surprising Dominance of Large Factor Models," Working paper, National Bureau of Economic Research.

Hastie, T, A Montanari, S Rosset, and RJ Tibshirani, 2022, "Surprises in High-Dimensional Ridgeless Least Squares Interpolation," *Annals of Statistics* 50, 949–986.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning* (2nd edition). (Springer, New York, NY, 2009).

Jensen, Theis Ingerslev, Bryan T Kelly, Semyon Malamud, and Lasse Heje Pedersen, 2024, "Machine Learning and the Implementable Efficient Frontier," Working paper 22-63, Swiss Finance Institute.

Kelly, Bryan, Semyon Malamud, and Kangying Zhou, 2024, "The Virtue of Complexity in Return Prediction," *Journal of Finance* 79, 459–503.

Kelly, Bryan T, Semyon Malamud, and Kangying Zhou, 2022, "The Virtue of Complexity Everywhere," Working paper, Swiss Finance Institute.

Mei, Song, and Andrea Montanari, 2022, "The Generalization Error of Random Features Regression: Precise Asymptotics and the Double Descent Curve," *Communications on Pure and Applied Mathematics* 75, 667–766.

Rahimi, Ali, and Benjamin Recht, 2007, "Random Features for Large-Scale Kernel Machines," *Advances in neural information processing systems* 20.

Rasmussen, Carl Edward, and Christopher K.I. Williams. *Gaussian Processes for Machine Learning* (MIT Press, Cambridge, MA, 2006).

Shen, Zhouyu, and Dacheng Xiu, 2025, "Can Machines Learn Weak Signals?," Working paper, National Bureau of Economic Research.

Sutherland, Danica J, and Jeff Schneider, 2015. On the error of random fourier features. in *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 862–871.

Welch, Ivo, and Amit Goyal, 2008, "A Comprehensive Look at The Empirical Performance of Equity Premium Prediction," *Review of Financial Studies* 21, 1455–1508.

Appendix

A. Spurious Predictability Induced by Standardizing Returns

The following analysis shows how standardizing can induce spurious predictability under the null of no true predictability.

Let excess returns on a stock market index return be $r_{t+1} = \mu + \sigma_t e_{t+1}$, where $\mu > 0$ and e_{t+1} is IID noise with $\mathbb{E}_t\, e_{t+1} = 0$ and $\mathbb{E}_t\, e_{t+1}^2 = 1$. Assume that σ_t^2 is time-varying, i.e., $\text{var}(\sigma_t^2) > 0$. Assume that σ_t^2 is observable to a researcher. The researcher examines market excess returns standardized by conditional volatility:

$$y_{t+1} = \frac{r_{t+1}}{\sigma_t} = \frac{\mu}{\sigma_t} + e_{t+1}. \tag{A.1}$$

The first term, $\frac{\mu}{\sigma_t}$, now has predictable variation and $\mathbb{E}_t\, y_{t+1} = \frac{\mu}{\sigma_t}$ is time-varying.

While a predictability test applied to r with predictor σ_t^{-1} would yield

$$\text{cov}(r_{t+1}, \sigma_t^{-1}) = 0, \tag{A.2}$$

applied to y it yields evidence of predictability

$$\text{cov}(y_{t+1}, \sigma_t^{-1}) = \mu\, \text{var}(\sigma_t^{-1}) \tag{A.3}$$

and the regression slope coefficient in a regression of y_{t+1} on σ_t^{-1} is equal to μ and the intercept is zero. Therefore, the fitted prediction is $\mu\sigma_t^{-1}$, which is also equal to $\mathbb{E}_t\, y_{t+1}$. The predictable variation is

$$R^2 = \frac{\mu^2\, \text{var}(\sigma_t^{-1})}{\text{var}(y_t)} \approx 0.43\% \tag{A.4}$$

which is not negligible in monthly data. This calculation uses $\mu = 0.0068$ (mean of the CRSP index return in the Goyal-Welch data set), $\text{var}(\sigma_t^{-1}) = 108.80$ (using 12m lagged volatility of index returns as σ_t), and $\text{var}(y_t) = 1.19$.

Now construct a timing strategy based on the predicted value

$$f_{t+1} = y_{t+1} x_t, \qquad \text{with} \quad x_t = \mu\sigma_t^{-1} \tag{A.5}$$

(The return of this timing strategy will be the same as the return of a volatility-timing strategy with weight $\mu\sigma_t^{-2}$ applied to the non-standardized return y_{t+1}, which exploits that μ does not vary with σ_t).

To get alpha, let's first get the covariance with the market excess return:

$$\begin{aligned} \text{cov}\left(f_{t+1}, y_{t+1}\right) &= \text{cov}\left(\frac{r_{t+1}}{\sigma_t}\left(\frac{\mu}{\sigma_t}\right), y_{t+1}\right) \\ &= \mathbb{E}\left[\frac{\mu\sigma_t^2 e_{t+1}^2}{\sigma_t^2}\right] \\ &= \mu \end{aligned} \tag{A.6}$$

so

$$\begin{aligned}\beta &= \frac{\mu}{\text{var}(r_{t+1})} \\ &= \frac{\mu}{\mathbb{E}[\sigma_t^2]}.\end{aligned} \tag{A.7}$$

Therefore,

$$\begin{aligned}\alpha &= \mathbb{E}\left[f_{t+1}\right] - \beta\mu \\ &= \mu^2\,\mathbb{E}[\sigma_t^{-2}] - \mu^2\,\mathbb{E}[\sigma_t^2]^{-1} \\ &= \mu^2\left(\mathbb{E}[\sigma_t^{-2}] - \mathbb{E}[\sigma_t^2]^{-1}\right) \\ &> 0,\end{aligned} \tag{A.8}$$

where the bound in the last line follows from Jensen's inequality. With the same moments as I used for the R^2 above, and after annualizing,

$$\alpha \approx 0.24. \tag{A.9}$$

Using the standardized market return as benchmark, as in KMZ, the alpha is smaller: Covariance with the standardized market return

$$\text{cov}\left(f_{t+1}, y_{t+1}\right) = \mu\,\mathbb{E}[\sigma_t^{-1}] \tag{A.10}$$

and, since $\text{var}(y_{t+1}) \approx 1$,

$$\beta = \mu\,\mathbb{E}[\sigma_t^{-1}] \tag{A.11}$$

$$\begin{aligned}\alpha &= \mathbb{E}\left[f_{t+1}\right] - \beta\mu\,\mathbb{E}[\sigma_t^{-1}] \\ &= \mu^2(\mathbb{E}[\sigma_t^{-2}] - \mathbb{E}[\sigma_t^{-1}]^2) \\ &= \mu^2\,\text{var}(\sigma_t^{-1}),\end{aligned} \tag{A.12}$$

which comes out to be annualized about $\alpha \approx 0.06$ with empirical moments for μ and $\text{var}(\sigma_t^{-1})$. In simulations, I find the same value of alpha on average, and an information ratio of about 0.25, which is not a negligible magnitude!

In the simulations with window size $T = 12$, however, I find that the fitted values from predictive regressions with RFF don't get close to capturing all this predictability that the reciprocal volatility strategy above captures. The information ratio I find in these simulations for the RFF-based strategy is only around 0.05. So much of the predictability captured by the RFF is due to something else, not the above mechanical effect related to standardization. That said, as a general matter, that the above analysis shows that standardizing the dependent variable is a somewhat dangerous practice in a study that has the objective of documenting predictability. It's only a small part of the story in KMZ, but it could play a bigger role in others.