A Machine Learning Approach to Risk Factors: *A Case Study Using the Fama–French–Carhart Model*

Joseph Simonian, Chenwei Wu, Daniel Itano, and Vyshaal Narayanam

Joseph Simonian
is the director of quantitative research at Natixis Investment Managers in Boston, MA.
joseph.simonian@natixis.com

Chenwei Wu
is a quantitative analyst at Natixis Investment Managers in Boston, MA.
chenwei.wu@natixis.com

Daniel Itano
is a senior quantitative analyst at Natixis Investment Managers in Boston, MA.
daniel.itano@natixis.com

Vyshaal Narayanam
is a data science co-op at Natixis Investment Managers in Boston, MA.
vnarayanam@natixis.com

Perhaps the most important defining characteristic of factor models is that they must explain asset behavior to a sufficient degree given a restricted set of explanatory variables. Given this, the primary challenge for anyone building a factor model is to settle on a set of factors that on one hand can adequately explain portfolio behavior over time, and on the other is simple enough to remain computationally tractable. In this way, the challenge faced in building a factor model is the same faced by scientists when building theories to explain natural phenomena, in which the trade-off between informative power and simplicity is also a fundamental consideration.

Although it is generally accepted that factor models should be built based on the foregoing principles, we often see practitioners developing and using factor models that deviate from them in significant ways. This is especially the case with the models that underlie many commercially available risk platforms, which often include hundreds of correlated variables that are presented as factors. The reason why commercial risk platforms take a maximalist approach to factor modeling is likely rooted in their motivation to provide a comprehensive picture of the risk exposures driving portfolio behavior. It is also rooted in their use of linear models. A linear factor model that restricts itself to a small number of factors faces the risk of providing an inadequate picture of portfolio behavior over a given measurement period. As a result, commercial risk platforms try to cover their bases by including a multitude of factors so that no exposure is seemingly unaccounted for. Despite this technical maneuver, the resulting frameworks are usually not genuine linear factor models, because of their size and the presence of correlated variables, nor are they maximally informative, because of their inability to account for the nonlinear behavior of and/or interaction effects among factors.

A natural response to the shortcomings of linear factor models is to recommend the use of nonlinear factor models; however, parametric nonlinear models have a number of shortcomings. First, the structure of the latter models is often heavily dependent on the sample data. As the sample expands or contracts, we at times find that the function specified by the model changes, sometimes dramatically. Second, unlike linear models, parameter estimates cannot always be derived analytically. Rather, solutions are often found using iterative methods, in which initial values are posited for each unknown

target variable before various optimization techniques are invoked to home in on a solution. Although iterative methods can be useful, the optimizations that drive them may ultimately fail to converge if the initial values are too distant from possible solution values. Initial values that are remote from optimal values can also cause convergence to a local solution rather than a global solution.

As a remedy to the drawbacks of both linear models and parametric nonlinear models, in this article the authors present a factor framework based on a machine learning algorithm known as *random forests* (RFs) (Ho 1995, 1998; Breiman 2001). The authors show how to use the RF algorithm to produce models that, within a single framework, provide information regarding the sensitivity of assets to factors broadly analogous to those generated by more commonly used frameworks, but with a significantly higher level of explanatory power. Moreover, RF-based factor models are able to account for the nonlinear relationships, discontinuities (e.g., threshold correlations), and interactions among the variables, while dispensing with the need for complex functional forms or additional interaction terms (thus remaining in harmony with the principle of parsimony). In the last section of the article, the authors demonstrate how the framework can be combined with another machine learning algorithm known as *association rule learning* (ARL) to build effective trading strategies, using a sector rotation strategy as an example.

BASIC FEATURES OF FACTOR MODELS

Investment factor models are supposed to provide insight into the primary drivers of portfolio behavior. Formally, there are various ways to build a factor model (for a basic overview, see Connor 1995). Perhaps the simplest way is via an ordinary least squares (OLS) regression, in which the portfolio return is the dependent variable, and the risk factors are the independent variables. As long as the independent variables have sufficiently low correlation, different models will be statistically valid and explain portfolio behavior to varying degrees. In addition to revealing what percentage of a portfolio's behavior is explained by the model in question, a regression will also reveal the sensitivity of a portfolio's return to each factor's behavior. These sensitivities are expressed by the beta coefficient attached to each factor.

Factor sensitivities and measures of explanatory power are the defining characteristics of factor models and are present in other common frameworks, such as those based on principal component analysis (PCA) (see Jolliffe 2002). As we show later, factor models based on machine learning can also describe the sensitivity of variables to the factors that explain them and provide information relating to the overall explanatory power of a given model. However, as previously mentioned, they also offer some distinct advantages over more traditional frameworks, such as the ability to capture nonlinear behavior and the interaction effects between factors. Additionally, RF models are generally less influenced by correlations between variables. Indeed, the question of multicollinearity does not enter into the picture when building an RF model in the way it does in an OLS regression.[1] One reason for this is that unlike OLS regression, RF models are estimated without requiring the inversion of a covariance matrix. Another distinct advantage of RF models is that they do not have strict parametric assumptions, nor do they rely on other time series assumptions such as homoskedasticity or independence of errors. Nevertheless, although RF models are relatively rule-free, it is our view that a fair amount of pre-model work should be done to ensure that the inputs into the model make sense from the standpoint of both investment relevance and economic coherence and possess a sufficient level of factor uniqueness to produce models that are both practical and free from explanatory redundancies. Although factor selection is an important aspect of building any factor model, it is especially critical when using machine learning-based methods.

MACHINE LEARNING AND THE RANDOM FOREST ALGORITHM

Machine learning refers to a collection of computational techniques that facilitate the automated learning of patterns and the formation of predictions from data. As such, machine learning methods can be used to build models with minimal human intervention and pre-programmed rules. Machine learning algorithms are (very) broadly classified as either *supervised* or *unsupervised*

[1] Those concerned with multicollinearity may benefit from using PCA or LASSO (least absolute shrinkage and selection operator) in the pre-model stage of an analysis to aid in generating factors that are unique.

learning algorithms. Unsupervised learning algorithms include those encompassing clustering and dimension reduction, in which the goal is to draw inferences and define hidden structures from input data. Unsupervised algorithms are distinguished by the fact that the input data are not categorized or classified. Rather, the algorithm is expected to provide a structure for the data. A well-known example of an unsupervised learning algorithm is *k*-means clustering (Lloyd 1982). In contrast, supervised learning (including reinforcement learning) algorithms use input variables that are clearly demarcated. With supervised learning, the goal is to produce rules and/or inferences that can be reliably applied to new data, whether for classification or regression-type problems. The RF algorithm used in this article is an example of a supervised learning algorithm and has been shown to be extremely effective in a variety of scientific applications, such as medical diagnosis, genome research, and cosmology.

Some machine learning algorithms, including RF, incorporate decision trees, a tool that is helpful in analyzing and explaining complex datasets. For regression-type problems, decision trees start from a topmost or *root node* and proceed to generate *branches*, with each branch containing a condition, and a prediction in the form of a real-valued number, given the condition in question. Trees are composed of a series of conditions attached to *decision nodes*, which ultimately arrive at a *leaf* or *terminal node* whose value is a real number.[2] The latter value represents a predicted value for a target variable given a set of predictor values. In Exhibit 1, we show a simple example of a decision tree that analyzes the relationship between monetary policy and macroeconomic conditions.

Decision trees can be constructed using various procedures (e.g., ID3, CHAID, MARS). In this article, we use a procedure known as CART (classification and regression tree, a methodology developed by Breiman et al. 1984). CART uses an algorithm called *binary recursive partitioning*, which divides the input space into binary decision trees. In this procedure, features are evaluated using all sample values, and the feature that minimizes the cost function at a specific value is chosen as the best split. Recursive partitioning takes place at each level down the tree, and the value at each leaf of the tree is the average of all the resulting observations.

In Exhibits 2, 3, and 4, we proceed to describe binary recursive partitioning and the RF algorithm in formal detail. In doing so, we use (with some modification) the descriptions provided by Cutler, Cutler, and Stevens (2012). We begin with the definition of binary recursive partitioning in Exhibit 2.

RF uses an ensemble of decision trees in conjunction with the CART technique. Each tree in the ensemble is constructed via bootstrapping, which involves resampling from the data with replacement to build a unique dataset for each tree in the ensemble. The trees in the ensemble are then averaged (in the case of regression), resulting in a final model. The bootstrap aggregation of a large number of trees is called *bagging*.[3] We describe the RF algorithm formally in Exhibit 3.

Predicted values of the response variable for regression[4] at a given point x are given by

$$\hat{f}(x) = \frac{1}{J}\sum_{j=1}^{J}\hat{h}_j(x)$$

where $\hat{h}_j(x)$ is the prediction of the response variable at x using the jth tree (This formula concludes the algorithm in Exhibit 3).

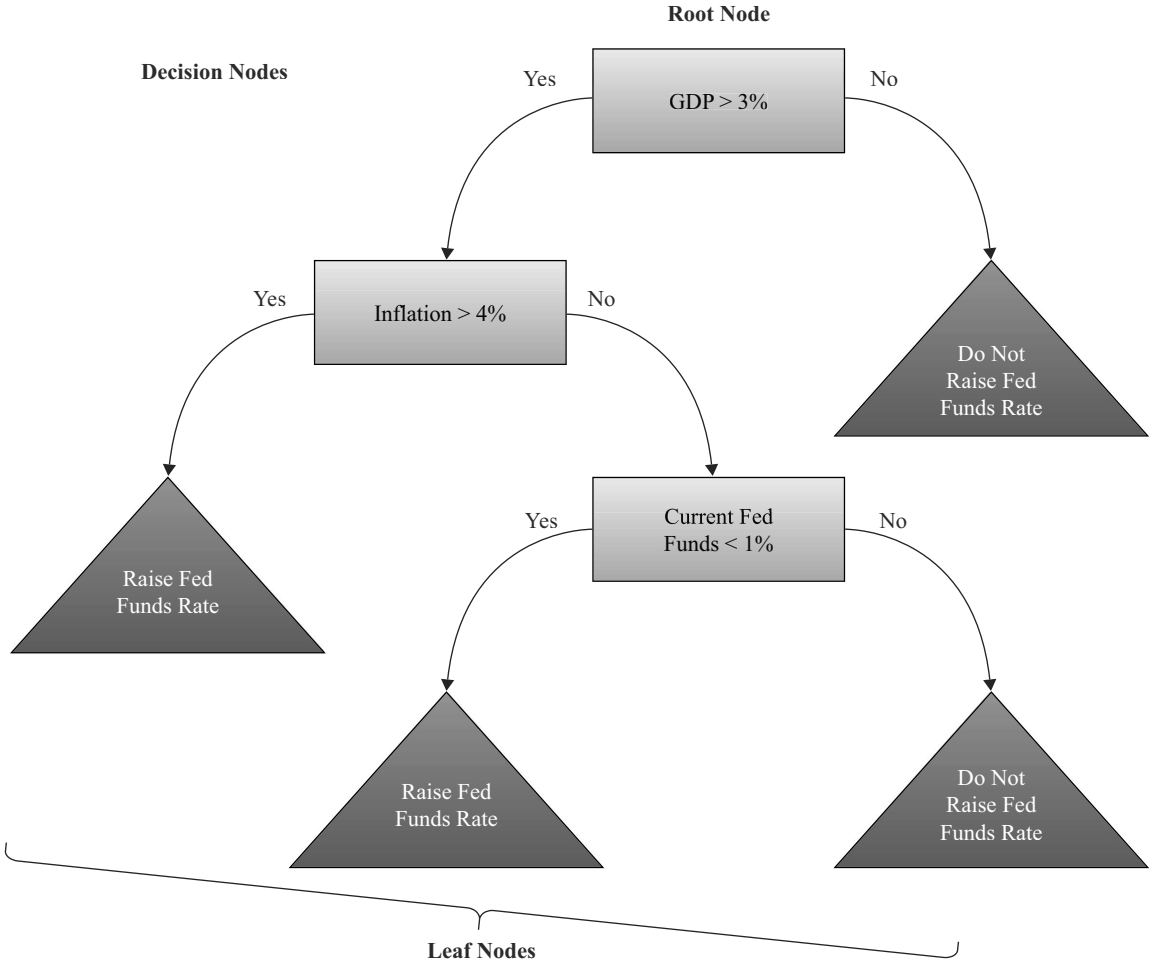
When a bootstrap is conducted, some observations are left out of the bootstrap. These are called *out-of-bag* (OOB) data and are used for measuring estimation error and variable importance. If trees are large, using all the trees may produce a false level of confidence in the predictions of the response variable for observations in the training set $\mathcal{D}$. To remedy this risk, the prediction of the response variable for training set observations is done exclusively with trees for which the observation is OOB. The resulting

[2] Several stopping criteria can be used to halt the tree-building process—for example, a minimum number of samples in a leaf, the depth of the tree, and the total number of leaves.

[3] Bagging is useful because it generally reduces overfitting and has a lower variance when compared to processes that only use individual decision trees. An individual tree may end up learning highly idiosyncratic relationships among the data and hence may end up overfitting the model. Averaging ensembles of trees provides a better opportunity to uncover more general patterns and relationships between variables. Overfitting can also be addressed by using simpler trees (i.e., those with a lower number of splits).

[4] For classification, the prediction values are given by $\hat{f}(x) = \text{argmax}_y \sum_{j=1}^{J} I(\hat{h}_j(x) = y)$.

EXHIBIT 1
Example of a Decision Tree



predictions, fittingly labeled *out-of-bag predictions*, are defined in Exhibit 4.

For regression[5] with squared error loss, generalization error is generally measured using the OOB mean squared error (MSE): $MSE_{OOB} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{f}_{OOB}(x_i))^2$.

An RF analysis produces two basic outputs. The first output is simply a set of conditional values—for example, a set of factor returns and a predicted value for a dependent variable such as a portfolio return, given the posited factor returns. The second output is something called *feature importance* (FI). As its name implies, FI indicates the importance of each explanatory variable in contributing to the predicted value of the dependent variable in question.

We calculate FI using *mean decrease accuracy*, which measures the degree to which the predictive power of the model would be diluted if the values for the explanatory variable in question were randomly changed. The mechanics of FI measurement work as follows: Once the *j*th tree is generated, the values for the predictor variables are randomly permuted in the bootstrapped sample, and the prediction accuracy is recalculated. For regression, the FI for the observation is calculated as the difference between the MSE of the predictions using the permuted

[5] For classification with zero-one loss, the generalization error rate is given by $E_{OOB} = \frac{1}{N}\sum_{i=1}^{N} I(y_i \neq \hat{f}_{OOB}(x_i))$.

EXHIBIT 2
Algorithm for Binary Recursive Partitioning

Let $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$ denote the training data, with $x_i = (x_{i,1}, ..., x_{i,p})^T$.

1. Start with all observations $(x_1, y_1), ..., (x_n, y_n)$ in a single node.
2. Repeat the following steps recursively for each unsplit node until the stopping criterion is met:
   a. Find the best binary split among all binary splits on all p predictors.
   b. Split the node into two descendant nodes using the best split (step 2a).
3. For prediction at x, pass x down the tree until it lands in a terminal node. Let K denote the terminal node, and let $y_{k_1}, ..., y_{k_n}$ denote the response values of the training data in node k. Predicted values of the response variable for regression are given by $\hat{h}(x) = \overline{y_k} = \frac{1}{n}\Sigma_{i=1}^{n} y_{k_i}$

Note: For classification, the prediction values are given by $\hat{h}(x) = \text{argmax}_\gamma \sum_{i=1}^{n} I(\gamma_{ki} = \gamma)$ *where* $I(\gamma_{ki} = \gamma) = 1$ *if* $\gamma_{ki} = \gamma$ *and 0 otherwise.*
Source: Cutler, Cutler, and Stevens (2012).

EXHIBIT 3
Algorithm for Random Forests

Let $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$ denote the training data, with $x_i = (x_{i,1}, ..., x_{i,p})^T$.

For $J = 1$ *to* j:

1. Draw a bootstrap sample $\mathcal{D}_j$ of size N from $\mathcal{D}$.
2. Using the bootstrap sample $\mathcal{D}_j$ as the training data, fit a tree using binary recursive partitioning (Exhibit 2):
   a. Start with all observations in a single node.
   b. Repeat the following steps recursively for each unsplit node until the stopping criterion is met:
      i. Select m predictors at random from the p available predictors.
      ii. Find the best binary split among all binary splits on the m predictors from step i.
      iii. Split the node into two descendant nodes using the split from step ii.

Source: Cutler, Cutler, and Stevens (2012).

EXHIBIT 4
Algorithm for Out-of-Bag Predictions

Let $\mathcal{D}_j$ denote the jth bootstrap sample and $\hat{h}_j(x)$ denote the prediction x from the jth tree, for $j = 1, ..., J$. For $i = 1$ to N:

1. Let $\mathcal{J}_i = \{j: (x_i, y_i) \notin \mathcal{D}_j\}$, and let J_i be the cardinality of $\mathcal{J}_i$ (Exhibit 3).
2. Define the OOB prediction for regression[6] at x_i to be $\hat{f}_{\text{OOB}}(x_i) = \frac{1}{J_i}\sum_{j \in \mathcal{J}_i} \hat{h}_j(x_i)$.

Note: For classification, the OOB prediction is given by $\hat{f}_{OOB}(x_i) = \text{argmax}_\gamma \sum_{j \in \mathcal{J}}^{j} I(\hat{h}_j(x_i) = \gamma)$, *where* $\hat{h}_j(x_i)$ *is the prediction of the response variable at* x_i *using the jth tree.*
Source: Cutler, Cutler, and Stevens (2012).

data and the MSE of the predictions using the original data: $FI_j(i) = MSE_{OOB_{Permuted}} - MSE_{OOB}$.[6]

Next, a normalization is generally conducted to allow an assignment of a relative FI (RFI) value to each feature. The normalization is accomplished by adding the FI values for each factor in a single tree and dividing that value into the FI value for each factor. This will yield a cross section of FI values that sum to unity. This operation is repeated for each tree, and the normalized FI (NFI) values are then averaged across all the generated trees to produce an RFI value for a given feature k—that is, $RFI(k) = \frac{\sum_{j=1}^{J} NFI_j(k)}{J}$. The RFIs will also fall in the range [0,1] and sum to unity. As we shall see in the forthcoming sections of the article, the RFI measure plays a pivotal role in building and interpreting RF-based approaches to factor modeling.

BUILDING FACTOR MODELS USING RANDOM FORESTS

Factor models are generally articulated as linear models despite the drawbacks highlighted earlier. Linear models are preferred by practitioners because they generally present readily understandable and interpretable analysis. In contrast, machine learning approaches, although useful in uncovering the nonlinear behavior of and interaction relationships among variables, are often articulated in a way that makes their output unintuitive, and hence unattractive, to many investment professionals. Nonetheless, as we shall demonstrate, it is possible to interpret the results of an RF factor analysis in a way that is both tractable and practical.

To frame our discussion, we use a variant of the well-known Fama–French–Carhart (FFC) equity factor model (Fama and French 1992, 1993; Carhart 1997). The FFC model is a multifactor extension of the capital asset pricing model (CAPM) (Treynor 1961; Sharpe 1964; Lintner 1965; and Mossin 1966), where the market represents the sole source of systemic risk. The FFC model extends the CAPM framework by introducing three new factors in addition to the market factor: the size factor (small-cap stock returns minus large-cap stock returns), value (high book-to-price stock returns minus low book-to-price stock returns), and momentum (high-returning stocks minus low-returning stocks).[7] Both the CAPM and the FFC are typically expressed as linear models. Thus, the RF variant of the FFC presented here provides a counterpoint to its traditional representation.

We use the FFC model to explain the performance of the 10 primary sectors of the stock market, with each sector represented by its respective Dow Jones index. In an RF model, the FFC factors function as features that we use to predict return values for each of our sectors. It is thus possible to examine how various factors influence the predicted values for a target variable when the latter takes on different values. A natural way to do this is to divide the predicted sector returns into percentiles and observe the value that factor returns take at each of them. Here we select observations that map to the 10th, 25th, 50th, 75th, and 90th percentile values of the target variable, then observe how each factor's influence on the predicted value of the target variable differs at each percentile.[8] Doing this produces

[6] For classification, $FI(i) = E_{OOB_{Permuted}} - E_{OOB}$. We note that although we have chosen to use MSE as our operative measure of feature importance, it is not the only one available. Other commonly used metrics include mean absolute error, the Gini index, and entropy.

[7] The following are detailed factor descriptions obtained from Ken French's website (http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html):

- $R_m - R_f$, the excess return on the market, is the value-weighted return of all CRSP firms incorporated in the United States and listed on the NYSE, AMEX, or NASDAQ that have a CRSP share code of 10 or 11 at the beginning of month t, good shares and price data at the beginning of t, and good return data for t minus the one-month Treasury bill rate.
- SMB (small minus big) is the average return on three small portfolios minus the average return on three big portfolios: $SMB = \frac{Small\ value + Small\ neutral + Small\ growth}{3} - \frac{Big\ value + Big\ neutral + Big\ growth}{3}$
- HML (high minus low) is the average return on two value portfolios minus the average return on two growth portfolios: $HML = \frac{Small\ value + Big\ value}{2} - \frac{Small\ growth + Big\ growth}{2}$
- Momentum is the average return on two high prior return portfolios minus the average return on two low prior return portfolios: $Mom = \frac{Small\ high + Big\ high}{2} - \frac{Small\ low + Big\ low}{2}$

[8] It is also possible to organize the explanatory variables into percentiles and investigate how the predicted values for the target variable change in response to significant shifts in the values of the predictors.

information regarding the sensitivity of sector returns to factor returns that is similar to that provided by a quantile regression (Koenker 2005). Observing variable behavior across percentiles is useful because doing so often reveals asymmetric relationships between factors and target variables within a set of observations.

In Exhibit 5, we show the returns for each FFC factor at different percentiles, as well as the predicted equity sector return. We also show the RF model R^2 value and the OLS R^2 value for each sector. As the exhibit shows, the R^2 produced by the RF model for each sector is, in general, significantly higher than that produced using an OLS regression. We also see that examining sector returns at different percentiles allows us to observe the varying influence of the FFC factors as the level of the predicted sector returns changes. In some cases, we see significant divergences between sector and factor returns. For example, for the consumer staples sector, at the 10th and 25th percentiles, all of the FFC factor returns are significantly more negative than the sector's predicted returns. One can interpret this result as reinforcing the sector's reputation as a defensive "low beta" sector. The opposite is true for the financials and materials sectors, whose predicted 90th percentile returns are significantly higher than the FFC factor returns.

In Exhibit 6, we show the RFI of each factor. Again, a factor's RFI indicates its importance in predicting sector returns when compared to the other factors in a set. Because RFI values sum to unity, they are naturally viewed as weights. As such, RFI values can plausibly be used to offer guidance in portfolio construction along the lines of a traditional returns-based style analysis (RBSA).[9] Assuming that investible proxies are available for the factors used in a given model, RFI values can be used to inform the weighting of the proxies used as constituents in a portfolio seeking to mimic the behavior of a target strategy. The RF model, however, possesses an advantage over a standard RBSA in generally providing a much better fit, as evidenced by the R^2 values it produces.

USING FEATURE IMPORTANCES TO DERIVE PSEUDO-BETAS

Because the RF model captures hierarchical (non-geometric) relationships between factors, it cannot be understood as a direct analog of an OLS regression or PCA because it does not convey the individual directional relationships between factors and assets. It is nevertheless possible to provide an interpretation of the RF model output so that the influence of the predictors can be understood in a way that is similar to traditional models. Previous attempts to "beta-ize" tree-based predictors have, for the most part, been of a more formal nature (e.g., Friedman 2001). Here we take a more conceptual approach because our goal is merely to provide a translation of the RF model output to individuals who are more familiar with linear models. We do not recommend using the results of the translation for trading applications, but simply as a communication device.

Recall that a widely accepted definition of *beta* is the elasticity of one variable to another. If we assume factor independence, then as a first step we can simply divide the predicted target variable return by each predictor return to gain a raw elasticity value for each factor. For example, let us consider the returns at the median for the industrials sector in Exhibit 5 referenced earlier. In the following, we list the sector and factor returns, along with the raw factor elasticity values in parentheses next to each.

The raw elasticity values indicate a sort of ceteris paribus degree of target variable sensitivity to each predictor; however, the raw values provide an incomplete picture of the relationship between target and predictor variables because they do not account for each factor's importance as a predictor, something expressed by RFI values. As such, our second step is to weight each factor's

Industrials	Rm-Rf	SMB	HML	MoM
1.3%	0.9% (1.44)	–0.1% (–13.0)	–0.3% (–4.33)	0.3% (4.33)

[9] Returns-based style analysis was introduced by Sharpe (1988, 1992). It is a way of analyzing and replicating investment strategies by means of investable proxies. The analysis is regression based, expressed formally as $R_t^m = \alpha + \sum_{i=1}^{I} \beta_i R_t^i + \epsilon_t$, where R_t^m is the return stream for the investment strategy to be replicated, R_t^i is the set of return streams for the proxy returns, I is the number of investable proxies, and ϵ_t is the error term. Two important constraints are put in place to produce a combination of investable proxies suitable for a long-only implementation. First, each beta coefficient is constrained to be greater than zero—that is, $\beta_i > 0$, $\forall i$. Second, the sum of the betas is constrained to sum to unity—that is, $\sum_{i=1}^{I} \beta_i = 1$. As such, each beta is interpreted as a weight assigned to a particular investable proxy in a replication portfolio.

EXHIBIT 5

Factor Percentile Returns and Equity Sector Predicted Values (monthly returns, Jan 1991 to Aug 2018)

Percentile	Consumer Discretionary	Rm-Rf	SMB	HML	MoM	Consumer Staples	Rm-Rf	SMB	HML	MoM
10th	–5.2%	–4.7%	–3.3%	–3.0%	–4.7%	–3.3%	–12.4%	–10.8%	–7.6%	–21.2%
25th	–1.7%	–1.9%	–1.8%	–1.5%	–1.4%	–0.9%	–5.3%	–3.6%	–3.6%	–5.4%
50th	1.3%	1.4%	0.3%	0.2%	1.0%	1.0%	2.2%	1.0%	1.0%	1.9%
75th	4.4%	3.7%	2.4%	2.1%	3.3%	2.9%	5.1%	3.4%	3.4%	4.6%
90th	6.7%	5.9%	3.9%	3.9%	5.2%	4.1%	11.4%	22.1%	12.9%	18.3%
R^2	0.96					0.91				
OLS R^2	0.81					0.44				
Percentile	**Energy**	**Rm-Rf**	**SMB**	**HML**	**MoM**	**Financials**	**Rm-Rf**	**SMB**	**HML**	**MoM**
10th	–4.6%	–4.5%	–3.3%	–2.9%	–4.2%	–5.9%	–3.2%	–2.8%	–2.4%	–2.9%
25th	–2.1%	–1.2%	–1.3%	–1.1%	–0.8%	–2.4%	–1.2%	–1.2%	–1.1%	–0.6%
50th	0.8%	0.8%	–0.2%	–0.3%	0.2%	1.6%	1.4%	0.3%	0.1%	0.9%
75th	4.4%	3.8%	2.5%	2.2%	3.4%	4.1%	3.9%	2.5%	2.3%	3.5%
90th	6.5%	7.0%	5.0%	5.5%	7.0%	7.9%	4.8%	3.2%	3.2%	4.1%
R^2	0.90					0.96				
OLS R^2	0.40					0.84				
Percentile	**Health Care**	**Rm-Rf**	**SMB**	**HML**	**MoM**	**Industrials**	**Rm-Rf**	**SMB**	**HML**	**MoM**
10th	–5.5%	–17.2%	–17.3%	–11.1%	–34.4%	–4.6%	–4.8%	–3.5%	–3.2%	–4.9%
25th	–1.3%	–3.2%	–2.4%	–2.0%	–2.8%	–1.4%	–1.4%	–1.4%	–1.2%	–1.0%
50th	1.2%	1.4%	0.3%	0.3%	0.9%	1.3%	0.9%	–0.1%	–0.3%	0.3%
75th	3.3%	3.1%	1.5%	1.3%	2.6%	3.7%	3.2%	1.7%	1.5%	2.6%
90th	3.3%	3.1%	1.5%	1.3%	2.6%	6.4%	5.2%	3.5%	3.5%	4.6%
R^2	0.91					0.97				
OLS R^2	0.48					0.83				
Percentile	**Information Technology**	**Rm-Rf**	**SMB**	**HML**	**MoM**	**Materials**	**Rm-Rf**	**SMB**	**HML**	**MoM**
10th	–6.3%	–5.1%	–3.6%	–3.2%	–5.0%	–4.9%	–3.7%	–3.0%	–2.6%	–3.5%
25th	–2.9%	–3.0%	–2.7%	–2.3%	–2.7%	–2.7%	–2.0%	–1.9%	–1.6%	–1.6%
50th	1.6%	0.6%	–0.3%	–0.3%	0.2%	0.8%	0.6%	–0.4%	–0.4%	0.1%
75th	5.5%	5.2%	3.5%	3.5%	4.6%	3.2%	3.0%	1.6%	1.7%	2.6%
90th	9.0%	7.1%	5.4%	5.8%	7.6%	7.6%	4.7%	3.3%	3.2%	4.5%
R^2	0.96					0.94				
OLS R^2	0.77					0.66				
Percentile	**Telecom Services**	**Rm-Rf**	**SMB**	**HML**	**MoM**	**Utilities**	**Rm-Rf**	**SMB**	**HML**	**MoM**
10th	–6.7%	–9.2%	–5.4%	–6.2%	–9.2%	–5.1%	–12.7%	–11.1%	–7.9%	–21.5%
25th	–2.1%	–2.6%	–2.2%	–1.8%	–2.1%	–1.6%	0.3%	1.3%	0.6%	1.0%
50th	1.0%	1.4%	0.2%	0.1%	0.8%	1.1%	1.5%	0.5%	0.4%	1.0%
75th	3.8%	6.1%	5.9%	4.8%	7.0%	3.2%	4.0%	2.3%	2.3%	3.5%
90th	5.0%	7.9%	6.8%	7.5%	9.9%	5.5%	6.5%	4.5%	4.8%	6.2%
R^2	0.89					0.87				
OLS R^2	0.40					0.21				

Source: http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html and Natixis Investment Managers.

EXHIBIT 6

Relative Feature Importance of Fama–French–Carhart Factors (Jan 1991 to Aug 2018)

	Rm-Rf	SMB	HML	MoM
Consumer Discretionary	0.84	0.04	0.05	0.07
Consumer Staples	0.50	0.20	0.15	0.15
Energy	0.52	0.13	0.18	0.17
Financials	0.72	0.07	0.16	0.05
Health Care	0.53	0.19	0.16	0.13
Industrials	0.84	0.05	0.07	0.04
Information Technology	0.73	0.05	0.15	0.08
Materials	0.69	0.07	0.11	0.13
Telecom Services	0.49	0.18	0.17	0.16
Utilities	0.37	0.25	0.20	0.18

Source: Natixis Investment Managers.

respective raw elasticity by its RFI to obtain a set of importance-adjusted elasticity values or *pseudo-betas*:

Rm-Rf	SMB	HML	MoM
1.21	–0.65	–0.30	0.17

We formally express the entire operation as

$$\frac{\text{Target variable value}}{\text{Predictor value}} \times \text{Feature importance}$$

Again, it is important to keep in mind that the intent here is not to discard the actual results of the analysis but to provide a simple way to facilitate communication with investment professionals who are accustomed to OLS betas, PCA loadings, and the like.

TRADING APPLICATION: BUILDING A SECTOR ROTATION STRATEGY USING THE RF FFC MODEL AND ASSOCIATION RULE LEARNING

In the previous sections of the article, we have shown how to use the RF algorithm to decompose risk ex post. In what follows, we adapt the framework for its use ex ante in trading applications. In particular, we apply our RF variant of the FFC model to build a sector rotation strategy. In doing so, we demonstrate how combining the output of an RF model with a simple, almost primitive signal can generate tradable information and provide the rudiments to developing a more sophisticated investment strategy. We do this to demonstrate the power of the RF model and to show that its effectiveness as an alpha generation tool does not necessarily depend on a complicated implementation.

We develop our trading strategy with the help of another machine learning methodology known as association rule learning (ARL) (Agrawal, Imieliński, and Swami 1993). ARL is a framework originally developed for discovering the relationships between sets of variables in a database. It can alternatively be viewed as a framework for deriving (learning) deductive inference rules from empirical data. In our example, we use ARL to establish a relationship between a pair of signals and the one-month-ahead return for a given sector over 18-month rolling windows. The signals are the RF-predicted return of a sector and the ratio of shorter-term to longer-term realized volatility (24-month vs. 36-month).[10] If (1) a positive relationship has been established between our signals and the one-month-ahead returns over the preceding 18-month window, (2) the ratio of shorter-term to longer-term volatility is less than one, and (3) the RF-predicted return for next month is greater than a designated threshold value, then we will own the sector for the month. Otherwise, the portfolio will carry a zero weight in the sector. The sectors that are owned will be equally weighted. We describe the association, trading, and portfolio construction rules that frame the strategy in formal detail in Exhibit 7.

We display out-of-sample backtest (Panel A) and bootstrap[11] results (Panel B) in Exhibit 8, comparing both unconstrained and constrained versions of our active strategy with a passive equal-weight portfolio.[12] In Exhibit 9, we show the cumulative out-of-sample backtest performance of each strategy. As we see in each exhibit, the active strategy outperforms the "no information" equal-weight portfolio, both in unconstrained form and with turnover constraints. The active strategy also exhibits respectable values for the

[10] The ratio of longer-term to shorter-term volatility has also been shown to reinforce other types of market signals (e.g., momentum). See Wang and Xu (2015) and Simonian et al. (2018).

[11] For the bootstrap, we use the stationary bootstrap approach described by Politis and Romano (1994), with an average block size of six months. The values in the exhibit are obtained by averaging 500 bootstrap samples.

[12] For each asset, $\textit{Constrained active weight} = \textit{Unconstrained active weight} \times 30\% + \frac{70\%}{\#\,\textit{Assets}}$.

EXHIBIT 7
Sector Rotation Strategy Rules

Association Rules[a]

Association rule: $X \Rightarrow Y$

Support of factors X in the set of observations T: $supp\ (X) = \dfrac{|\{t \in T; X \subseteq t\}|}{|T|}$

Lift of a rule, $X \Rightarrow Y$: $lift\ (X \Rightarrow Y) = \dfrac{supp\ (X \cup Y)}{supp\ (X) \times supp\ (Y)} = \dfrac{|\{t \in T; X \cup Y \subseteq t\}| \times |T|}{|\{t \in T; X \subseteq t\}| \times |\{t \in T; Y \subseteq t\}|}$

Trading Rules[b]

Association rule: $(VR_t < 1, \hat{r}_{t+1} > \alpha) \Rightarrow r_{t+1} > \beta$

where

$\hat{r}_{t+1}$ = RF prediction for one-month-ahead sector return

VR_t = *Volatility ratio at time t* = $\dfrac{Volatility\ (t-24,\ t)}{Volatility\ (t-36,\ t)}$

$VR_t < 1$: Near-term volatility is smaller than long-term volatility.

α and β are non-negative thresholds. In our example, we posit a value of 0 for α and 0.02 for β.

Support of the signal: $supp\ \left((VR_t < 1,\ \hat{r}_{t+1} > \alpha)\right) = \dfrac{\#\ of\ months\ (VR_t < 1,\ \hat{r}_{t+1} > \alpha)}{Window\ Length}$

Lift of the rule: $lift\ \left((VR_t < 1,\ \hat{r}_{t+1} > \alpha) \Rightarrow r_{t+1} > \beta\right) =$

$$\frac{\dfrac{\#\ of\ months\ (VR_t < 1,\ \hat{r}_{t+1} > \alpha,\ r_{t+1} > \beta)}{Window\ Length}}{\dfrac{\#\ of\ months\ (VR_t < 1,\ \hat{r}_{t+1} > \alpha)}{Window\ Length} \times \dfrac{\#\ of\ months\ (r_{t+1} > \beta)}{Window\ Length}}$$

Trading and Portfolio Construction Rules

1. *IF*, at the end of month t:
   a. A sector's $lift_{(t-18,t)} > 1.1$
   b. $VR_t < 1$
   c. $\hat{r}_{t+1} > \alpha$
2. *THEN* $w_i > 0$
3. *ELSE* $w_i = 0$
4. *Portfolio construction Rule:* $w_i = \dfrac{100\%}{N}$ if $N \neq 0$; $w_i = \dfrac{100\%}{\#\ Assets}$ if $N = 0$

where w_i is the portfolio weight of asset i, and N is the number of sectors with a positive weight in the portfolio.

---

[a] *An additional way of measuring rule strength is via the confidence of a rule, $X \Rightarrow Y$, where*

$$conf\,(X \Rightarrow Y) = \frac{supp\,(X \bigcup Y)}{supp\,(X)} = \frac{|\{t \in T\}; X \bigcup Y \subseteq t|}{|\{t \in T\}; X \subseteq Y|}$$

[b] *It is also possible to construct our trading rule using the confidence of a rule:*

$$conf\ ((VR_t < 1, \hat{r}_{t+1} > \alpha) \Rightarrow r_{t+1} > \beta) = \frac{\dfrac{\#\ of\ months (VR_t < 1, \hat{r}_{t+1} > \alpha, r_{t+1} > \beta)}{Window\ length}}{\dfrac{\#\ of\ months (VR_t < 1, \hat{r}_{t+1} > \alpha)}{Window\ length}}$$

Source: Agrawal, Imieliński, and Swami (1993).

EXHIBIT 8

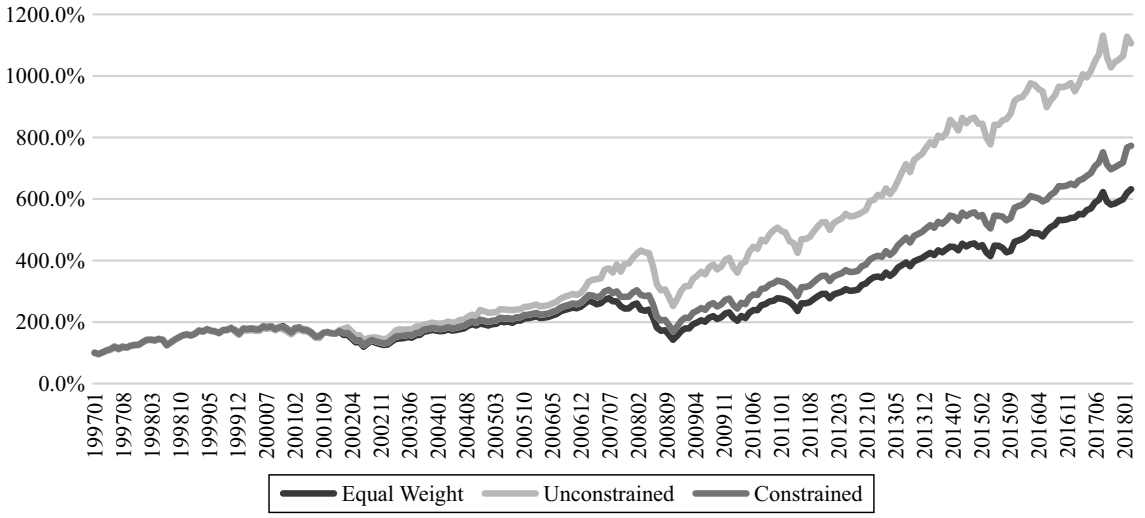
Sector Rotation Strategy Backtest and Bootstrap Results (Jan 1997 to Aug 2018)

	Annualized Return	Annualized Volatility	PSharpe 0.0	PSharpe 0.1	PSharpe 0.2	PIR 0.0	PIR 0.1	PIR 0.2	Turnover	Total Cumulative Return
Panel A: Out-of-Sample Backtest Results										
Equal-Weight	10.0%	13.8%	99.2%	78.8%	23.4%	N/A	N/A	N/A	0.0%	632%
Unconstrained Strategy	12.9%	14.2%	99.9%	94.9%	53.8%	96.9%	60.3%	10.3%	74.6%	1106%
Constrained Strategy	11.0%	13.6%	99.7%	87.3%	35.0%	97.8%	66.1%	13.6%	22.4%	773%
Panel B: Bootstrap Performance Results										
Equal-Weight	11.0%	13.4%	96.7%	78.9%	43.0%	N/A	N/A	N/A	0.0%	982%
Unconstrained Strategy	12.2%	14.4%	97.6%	82.2%	45.8%	66.7%	25.7%	4.2%	85.4%	1202%
Constrained Strategy	11.4%	13.4%	97.3%	81.3%	45.9%	66.7%	25.8%	4.3%	25.6%	1041%

Source: Natixis Investment Managers.

EXHIBIT 9

Cumulative Out-of-Sample Backtest Performance of Sector Rotation Strategy vs. Equal-Weight Portfolio (Jan 1997 to Aug 2018)



probabilistic Sharpe ratio (PSharpe) introduced by Bailey and López de Prado (2012)[13] and favorable values for the information ratio variant of the PSharpe (PIR), where the equal-weighted portfolio is used as the benchmark. The PSharpe measure is designed to show the probability of a strategy achieving a given Sharpe ratio threshold given a specific track record or backtest length and the

[13] The PSharpe is defined as

$$\widehat{PSR}(SR^*) = Z\left[\frac{(\widehat{SR} - SR^*)\sqrt{n-1}}{\sqrt{1-\hat{\gamma}_3 SR^* + \frac{\hat{\gamma}_4 - 1}{4}\widehat{SR}^2}}\right]$$

where Z [·] is the cumulative distribution function of a standard normal distribution, and $\widehat{SR}$ is the observed Sharpe ratio. SR^* is the predefined benchmark Sharpe ratio (ex ante Sharpe ratio), n is the number of periods over which the strategy's performance is tested, and $\hat{\gamma}_3$ and $\hat{\gamma}_4$ are the respective observed skewness and kurtosis values of the strategy.

presence of non-normal returns. The comparatively favorable results for our active investment strategy demonstrate that even with the barest of inputs, machine learning methods—and the RF and ARL frameworks in particular—provide powerful means to uncover useful patterns in investment data. It is a given that the strategy presented here could be built up and improved upon, with the introduction of new factors and/or a more nuanced treatment of existing inputs. Nevertheless, the results here convincingly speak to the investment insights that can be gained by practitioners willing to incorporate machine learning methods into their investment process.

CONCLUSION

Machine-learning approaches to risk factor modeling offer investment practitioners the ability to enrich their analysis by providing insight into relationships between variables that are unaccounted for in more traditional models such as OLS regression. By means of the RF algorithm, the authors uncover nonlinear relationships and interaction effects between the well-known FFC factors and show how to translate the output from the RF model so that it has the basic form of a more traditional factor model. In the last section of the article, the authors combine the RF algorithm with another machine learning framework, association rule learning, to build a sector rotation strategy. The article thus demonstrates that machine learning approaches can inform both risk analysis and portfolio management, providing readily usable output that can be communicated in a straightforward manner.

REFERENCES

Agrawal, R., T. Imieliński, and A. Swami. "Mining Association Rules Between Sets of Items in Large Databases." In *Proceedings of the 1993 ACM–SIGMOD '93*, Washington, D.C., May 1993, pp. 207–216. New York: ACM Press.

Bailey, D. H., and M. López de Prado. 2012. "The Sharpe Ratio Efficient Frontier." *Journal of Risk* 15 (2): 3–44.

Breiman, L. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Monterey, CA: Wadsworth & Brooks/Cole Advanced Books & Software, 1984.

Carhart, M. M. 1997. "On Persistence in Mutual Fund Performance." *The Journal of Finance* 52 (1): 57–82.

Connor, G. 1995. "The Three Types of Factor Models: A Comparison of Their Explanatory Power." *Financial Analysts Journal* 51 (3): 42–46.

Cutler, A., D. R. Cutler, and J. R. Stevens. "Random Forests." In *Ensemble Machine Learning*, edited by C. Zhang and Y. Ma, pp. 157–175. Boston: Springer, 2012.

Fama, E. F., and K. R. French. 1992. "The Cross-Section of Expected Stock Returns." *The Journal of Finance* 47 (2): 427–465.

——. 1993. "Common Risk Factors in the Returns on Stocks and Bonds." *Journal of Financial Economics* 33 (1): 3–56.

Friedman, J. H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." *Annals of Statistics* 29 (5): 1189–1232.

Ho, T. K. "Random Decision Forests." *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, Montreal, QC, 14–16 August 1995, pp. 278–282.

——. 1998. "The Random Subspace Method for Constructing Decision Forests." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (8): 832–844.

Jolliffe, I. T. *Principal Component Analysis*. New York: Springer-Verlag, 2002.

Koenker, R. *Quantile Regression*. Cambridge: Cambridge University Press, 2005.

Lintner, J. 1965. "The Valuation of Risk Assets and the Selection of Risky Investments in Stock Portfolios and Capital Budgets." *The Review of Economics and Statistics* 47 (1): 13–37.

Lloyd, S. P. 1982. "Least Squares Quantization in PCM." *IEEE Transactions on Information Theory* 28 (2): 129–137.

Mossin, J. 1966. "Equilibrium in a Capital Asset Market." *Econometrica* 34 (4): 768–783.

Politis, D. N., and J. P. Romano. 1994. "The Stationary Bootstrap." *Journal of the American Statistical Association* 89 (428): 1303–1313.

Sharpe, W. F. 1964. "Capital Asset Prices: A Theory of Market Equilibrium Under Conditions of Risk." *The Journal of Finance* 19 (3): 425–442.

——. 1988. "Determining a Fund's Effective Asset Mix." *Investment Management Review* 2 (6): 59–69.

——. 1992. "Asset Allocation: Management Style and Performance Measurement." *The Journal of Portfolio Management* 18 (2): 7–19.

Simonian, J., O. Sosa, E. Heilbron, M. Senoski, and T. McFarren. 2018. "Capital-Market-Aware LDI: Actively Navigating the De-Risking Journey." *The Journal of Portfolio Management* 44 (2): 130–135.

Treynor, J. L. "Market Value, Time, and Risk." Unpublished manuscript (1961), No. 95–209.

Wang, K. Q., and J. Xu. 2015. "Market Volatility and Momentum." *Journal of Empirical Finance* 30 (1): 79–91.

Disclaimer
This editorial reflects the current opinions of the authors, which are not those of Natixis Investment Managers.