Maximizing the Sharpe Ratio: A Genetic Programming Approach*

Yang Liu
Tsinghua University

Guofu Zhou
Washington University in St. Louis

Yingzi Zhu
Tsinghua University

Current version: November, 2020

*We are grateful to Chris Neely and the seminar participants at London Business School, Sichuan University, Washington University in St. Louis and Zhejiang University, and conference participants at 2018 International Accounting and Finance Doctoral Symposium, 2019 Conference on Finance Predictability and Data Science, 2019 China Finance Review International Conference, and 2020 China FinTech conference in Qingdao for very helpful comments. Liu and Zhu acknowledge the financial support from National Natural Science Foundation of China (# 71572091). Part of this work is accomplished while Liu is visiting Washington University in St. Louis.

Send correspondence to Guofu Zhou, Olin School of Business, Washington University in St. Louis, St. Louis, MO 63130; e-mail: zhou@wustl.edu; phone: 314-935-6384.

Maximizing the Sharpe Ratio: A Genetic Programming Approach

Abstract

While common machine learning algorithms focus on minimizing the mean-square errors of model fit, we show that genetic programming, GP, is well-suited to maximize an economic objective, the Sharpe ratio of the usual spread portfolio in the cross-section of expected stock returns. In contrast to popular regression-based learning tools and the neural network, GP can double their performance in the US, and outperform them internationally. We find that, while the economic objective plays a role, GP captures nonlinearity in comparison with methods like the Lasso, and it requires smaller sample size than the neural network.

JEL Classification: G12, G14, G15

Keywords: Machine Learning, Genetic Programming, Cross-sectional Returns, Predictability

"One general law, leading to the advancement of all organic beings, namely, multiply, vary, let the strongest live and the weakest die."

– Darwin, C., On the origin of species, 1859.

1. Introduction

Machine learning (ML) revolutionizes the research in all sciences, with its presence almost everywhere today. In finance, its applications at present appear concentrated in estimating the cross-section expected stock returns, perhaps because explaining why different assets have different returns is one of the central questions of finance. For examples, Chinco, Clark-Joseph, and Ye (2019) apply LASSO to analyze cross-firm return predictability at the one-minute horizon. Kozak, Nagel and Santosh (2020) provide a Bayesian LASSO approach to shrink factor dimensionality. Feng, Giglio, and Xiu (2020) focus on choosing factors. Gu, Kelly and Xiu (2020) apply a comprehensive set of ML tools, including generalized linear models, dimension reduction, boosted regression trees, random forests, and neural networks, to forecast individual stocks and their aggregates. Freyberger, Neuhierl, and Weber (2020) find nonlinear effects in the cross section of expected stock returns. Han, He, Rapach, and Zhou (2020) develop a new C-LASSO approach to handle a comprehensive set of firm characteristics. Filippou, Taylor, Rapach, and Zhou (2020) apply LASSO and neural network to predict foreign exchanges, and Guo, Lin, Wu, and Zhou (2019) conduct a machine learning study on corporate bonds. While all of these studies are different in their economic motivations, their solutions are based on the objective functions of the machine learning literature.

A natural questions is whether we can apply or extend existing learning tools to maximize directly our economic objective function at hand. In particular, while different data sets or different methodologies are employed, existing studies, such as Lewellen (2015), Green, Hand, and Zhang (2017), and Freyberger, Neuhierl, and Weber (2020), all assess the economic significance by examining the performance of the well known spread portfolio, which longs assets in the highest estimated expected return group, and shorts those assets in the lowest. Because the GP is constructed to directly maximize the Sharpe ratio, it is likely to produce a portfolio with a higher Sharpe ratio than methods that minimize mean-squared errors.

In this paper, we show empirically that it is indeed the case with the use of the generic programming (GP). While standard machine learning tools, such LASSO and neural network used in recent cross-section studies, are not readily to be adapted to maximize the Sharpe ratio directly, GP is known, at least as early as Neely, Weller and Dittmar (1997), and Allen and Karjalainen (1999) in the finance literature, as a flexible optimization method in setting the objective function. However, this flexibility or the GP in general have not receive much attention in the past, due to perhaps its extremely heavy burden in computations, explained further later. In comparison with existing GP studies, we optimize the Sharpe ratio of an investment portfolio in the cross-section context. In fact, our paper appears the first to use the GP to maximize the Sharpe ratio, and the first to apply it for forecasting returns in the cross-section.[1] Moreover, following Zhang and Bhattacharyya (2004) and Bhowan, Johnston, Zhang and Yao (2012), we incorporate new advances into the GP, allowing for multiple sets of parameters in training samples and an ensemble method in selecting the ultimately mapping function.

With 15 firm characteristics that capture size, momentum and price trends over different time horizons, we find that the GP outperforms not only the leading regression-based machine learning methods, the ridge, LASSO, elastic net (Enet), PCR, and PLS, but also the powerful neural network models (NN1-NN5 with one to 5 layers) of Gu, Kelly, and Xiu (2020). In the out-of-sample period from 1991 to 2019, the GP yields the greatest average return of 1.71% per month, while the second greatest return, earned by NN2, is only 1.22%. As for the Sharpe ratio, GP has the largest annualized Sharpe ratio, 1.32. In contrast, the linear models produce a Sharpe ratio level of only about 0.70 (almost 50% lower) and the neural networks produce a level around 0.95 (almost 30% lower).

The largest differences occur during the post-2003 subperiod. Green, Hand, and Zhang (2017), noting a number of changes in firm reporting practice and government regulations, find that 2003 is a major structural break point in predicting the cross-section stock returns with firm characteristics. Their results are replicated with our data. Indeed, all the regression-based machine learning methods fail to generate significant average returns in their spread portfolios, though one of the five neural network models manages to get a significant average return of 0.53 per month with a t-stat of 2.19. In contrast, the GP obtains an average return of 0.72% per month with a t-stat of

[1] Assume knowing the true parameter, then there is certain equivalence between finding the stochastic discount factor and maximizing the Sharpe ratio. Kozak, Nagel and Santosh (2020) and Bryzgalova, Pelger and Zhu (2020) are the studies of the former based on firm characteristics.

3.06. Its Sharpe ratio is 0.77, still about 30% greater than any of the other strategies. In short, the GP really makes an important economic difference when it is used to maximize the Sharpe ratio of the spread portfolio.

What is the relation between the GP spread portfolio and other spread portfolios? We regress the expected stock return generated by GP on those generated by others in a cross-section regression, and then examine the decile portfolio sorted by the resulting residuals. Controlling for other models, GP persistently yields highly significant spread return. In contrast, controlling for GP, the spread return of other models shrinks almost to zero, indicating that GP contains substantially more information than other models by subsuming their predictability.

To understand the time-varying outperformance, we construct an idiosyncratic volatility (IVOL) index, defined as the cross-section average of the IVOL of individual stocks, to reflect the market information uncertainty. We find that the improvement of GP over other models is mainly attributed to its good performance during the high-IVOL periods. Since noise-to-signal ratio in the stock market is pretty high, the GP model which excels in extracting signals in market condition with high information uncertainty level is expected to generate more powerful predictability.

We also examine the relation of the GP with various well known factor models in the literature. Following Fama and French (1993), we construct a GP factor, GPF, based on a standard 2×3 double sorting of size and the expected return generated by GP. All these machine learning methods easily produce largely unexplained alphas of about 0.90% with significant t-statistic over 3, assuming the existing factor models. However, the GPF alone can explain all the other spread portfolios, the average absolute alpha is only up to 0.11% with a negligible t-statistic of 0.43. Moreover, the p-value of the Gibbons, Ross, and Shanken (1989) (GRS) test for the GPF to price these spread portfolios is 0.88, while the p-values for other factor models are less than 10^{-4}. Moreover, adding the GPF to existing factor models improves significantly the Sharpe ratios, implying that it can serve as an additional factor based on the Sharpe ratio test of Barillas and Shanken (2017).

The performance of the GP is robust in a number of ways. First, once we replace the 15 firm characteristics by the 15 used by Lewellen (2015), which are primarily fundamental variables, the superior performance remains. In fact, the GP performs even better relative to existing methods, with Sharpe ratio almost more than doubling those of others. Second, GP also performs well internationally in the other G7 countries. Its spread portfolio is economically and statistically significant across the 6 markets, and its Sharpe ratio is always the greatest, consistent with the US results.

Thirdly, the GP is also robust to alternative setups of two parameters that determine its search for the maximum. Denote *Pop* as the individual number in each generation of the GP algorithm, and *Gen* as the maximum number of the generations. As they characterize the searching depth, it is obvious that the in-sample performance increases in either of <*Gen*, *Pop*>. Indeed, even if the average Sharpe ratio for randomly generated individuals in the first generation is close to 0, the Sharpe ratio shows a strong increasing pattern as the generation increases. This evolution path suggests that GP indeed "learns" from the data and attempts to optimize the Sharpe ratio. However, increases in *Pop* is marginal compared with that of *Gen*. Moreover, we find that *Gen* also controls the volatility and convergence of the algorithm. Intuitively, simulated individuals in the earlier stage are more diversified, but as *Gen* increases, the new individual will evolve in the same direction guided by the objective. Hence, while achieving higher Sharpe ratios, they also become less-diversified and less volatile. Although the optimal <Gen, Pop> are chosen via validation, we examine a number of alternative parameters and find the results are robust.

To see the importance of setting an objective to maximize the Sharpe ratio, we provide an analysis of an alternative use of the GP algorithm with minimizing the conventional mean squared error (MSE) as the objective. The results show that our previous GP substantially outperforms this MSE-based GP by yielding a spread portfolio with higher return and Sharpe ratio, and by subsuming its predictability.

To understand what conditions that drive the performance of the GP, we conduct two types of econometric analysis. First, we simulate data from a linear model. In this case, the GP performs similarly with other models because if the data are truly linear, learning from minimizing the MSE should learn perfectly on the data, and so the the Sharpe ratio objective makes little difference. In the second case, we simulate the data from a nonlinear model. In this case, as expected, the GP substantially outperforms the linear regression-type models with much higher Sharpe ratios. While the neural network models should capture the nonlinearity, we find that they require relative larger sample size to perform well, explaining why they performs worse than GP in the real data sets.

Our paper adds to the small literature on the applications of GP into finance. Neely, Weller and Dittmar (1997) seems of the earliest studies in finance, who apply the GP to find profitable technical rules. Allen and Karjalainen (1999) apply the GP to find profitable trading rules to beat the S&P 500 index, but unsuccessfully. Recently, Brogaard and Zareei (2018), with modified algorithms, are able to identify stronger time-series predictability of the S&P 500 index. Ready (2002) also use GP to investigate the profitability of the technical trading rules on DJIA index. In addition, Dempster and Jones (2001) and Dunis, Laws, Middleton, and Karathanasopoulos (2015) apply it to currency and commodities. All these existing studies are about using the GP for time series prediction. In contrast, our paper is about cross section prediction. As mentioned earlier, the hurdle of applying the GP is computational time, which is especially critical in our cross section context which deals with thousands of stocks. Indeed, even on a server with an Intel Xeon E7-8890 and 512 GB memory, the computation time takes days for our study. Nevertheless, with increasing computing power each year, the application of the GP in finance will surely increase drastically over time, simply due to the flexibility of the algorithm that it can be used to maximize any economic objective.[2]

The rest of the paper is organized as follows. Section 2 discusses the data and the methodology of our GP model and other competing machine learning models. Section 3 presents the main results. Section 4 examines the robustness. Section 5 explores the explanation for GP's good performance. Section 6 concludes.

2. Data and methodology

In this section, we first introduce the data, and then discuss the GP algorithm for maximizing the Sharpe ratio in the cross-section, along with a review of other machine learning methods for comparison.

2.1. Data

As usual, we use all domestic common stocks listed on NYSE, AMEX, and Nasdaq stock markets, and exclude close-end funds, real estate investment trusts, unit trusts, American depository receipts, and foreign stock (or stocks that do not have a CRSP share code of 10 or 11). As the literature typically does, we employ the price filter to exclude the stocks with price below $5.

[2] Nordhaus (2001) shows that the computing power has increased by around 80% per year since 1980.

The primary set of characteristics consists of 15 variables: the market capitalization (size) and 3 past return-based signals, i.e., R_{-1}, $R_{-12,-2}$, and $R_{-60,-13}$, which correspond to the short-term reversal ($SREV$) of Lehmann (1990), Lo and MacKinlay (1990), momentum (MOM) of Jegadeesh and Titman (1993), and long-term reversal ($LREV$) of DeBondt and Thaler (1985), respectively. In addition, we also include the 11 price moving average (MA) signals used in Han, Zhou and Zhu (2016), including MAs of lag lengths of 3-, 5-, 10-, 20-, 50-, 100-, 200-, 400-, 600-, 800-, and 1000-days. Following the most recent studies, we normalize each indicator in the cross-section such that it has a mean of zero and a standard deviation of one without loss of generality. We use this characteristic set because it is easy to construct, making it ideal for comparison in international markets. However, since this set relies heavily on technical signals, we also use another 15 characteristics of Lewellen (2015), which are mostly fundamental variables, as a robustness check.

2.2. The GP algorithm

In this subsection, we first discuss the objective function and search space, and then we introduce the optimization procedure and hyperparameter tuning.

2.2.1. Incorporating economic objective

Our economic objective is to maximize the Sharpe ratio of a portfolio based on firm characteristics, which is of importance to an investor or fund manager who would like to achieve the maximum economic gains from the information on characteristics. While we find that it is difficult to solve this problem using other existing machine learning tools, the GP appears the best to fit the purpose.

Mathematically, our objective is to find a function $G(\cdot)$ to maximize the Sharpe ratio (SR) of the usual decile long-short spread portfolio, but here the long and short legs are determined endogenously,

$$\max_{G(\cdot)\in\mathcal{M}} SR(Spread(G(\cdot))), \tag{1}$$

where $\mathcal{M}$ is the search space, $G(\cdot)$ is a function mapping from the stock characteristics to the expected return, and $Spread(G(\cdot))$ is the resulting spread portfolio. In particular, suppose X is a panel data of stock characteristics, in which $X_{i,t}$ is a vector of characteristics for stock i on month

t. Denote the expected return for stock i in month t generated by $G(\cdot)$ as

$$ER_G^{i,t} = G(X_{i,t-1}). \tag{2}$$

Then, we can sort stocks by $ER_G^{i,t}$ in each month into decile groups and construct a value-weighted spread portfolio, so weighted as all other portfolios in the paper, and denote it as $Spread(G(\cdot))$. Put differently, we want to search for the optimal function $G(\cdot)$ to maximize the Sharpe ratio of $Spread(G(\cdot))$.

Genetic programming (GP) is a supervised machine learning method based on the principle of Darwinian natural evolution. Since its launch by Koza (1992), GP has been successfully applied in various fields, such as economics, finance, and engineering. GP randomly generates initial population of a certain number of individuals, each of which is a solution candidate to the given problem. The performances of the solution candidates are evaluated according to a problem-specific fitness function (objective function), which defines the environment for the evolution. Then, the individuals are randomly selected as parents individuals, with the selection probabilistically biased in favor of the relatively fit members. Next, the parents individuals are combined by genetic operators, such as crossover and mutation, to creates offspring individuals. Afterward, successive generations are generated in the same way until the final generation.

For the optimization problem of Equation (1), the GP is ideal, as it is often used for solving optimization problems with objective functions which are non-differentiable or difficult to be expressed in other optimization approaches. In addition, as a non-parametric model, GP can discover both the model structure and model parameters, and thus are more flexible in exploring nonlinear predictability. Moreover, due to the stochastic nature, it is less likely to converge to local optima, and it is generally suitable to search for global optimum in large search space.

2.2.2. Representation and search space

In GP, the solution candidates are represented as tree structures and can be encoded as function $G(\cdot)$ mapping from characteristics X to expected returns, which is discussed in the Online Appendix in more detail. Each individual $G(\cdot)$ is build of two basic primitives, the *terminal* nodes and *function* nodes. Essentially, the terminal nodes provides the inputs to the GP program, and it includes the input characteristics (X) and some random constants. The *function* nodes comes from a predefined function set. Panel A of Figure 1 shows an example of the tree-structure individuals. It consists of two characteristics of X_1 and X_2, a random constant of 1, and two function operators of MULTIPLY ($\times$) and ADD ($+$). It can be coded as a function $G(X) = X_1(X_2 + 1)$. In terms of its economic interpretation, this solution represents such a hypothesis about the cross-section of stock returns that stocks with greater X_1 tends to have higher future returns. In addition, it also assumes that this effect increases with X_2 by adding an interaction item of X_1 and X_2.

The search space $\mathcal{M}$ is spanned by a large set of functions combining an indicator set and an function set. The indicator set X includes the firm characteristics such as the 15 discussed in section 2.1.. The function set includes both commonly used linear and nonlinear operators, examples of which the linear functions are ADD, MINUS, NEGATIVE, and the nonlinear ones are MULTIPLY, DIVIDE, SIN, COS, ABS, and bool-type operator CMP. This enables GP to exploits both the linear and nonlinear predictability of the characteristics. However, though we do not assume any specific function form for $G(\cdot)$, we limit the maximum of tree depth to 30 for tractability. This still enables a sufficiently large space of millions of candidate solutions, and controls the model complexity and overfitting at the same time.

2.2.3. Optimization

It is important to examine how the GP selects the individuals to maximize the Sharpe ratio. Different from the common gradient-based method, the optimization of GP is based on the principle of Darwinian natural evolution.

Essentially, GP optimizes the given problem by iteratively producing offspring individuals based on genetic operators and then selecting strong individuals by the natural selection principle. The direction of the evolution is characterized by the fitness function, i.e, the optimization objective, which is the Sharpe ratio of the spread portfolio in our case. In particular, after initiating the random individuals in the first generation, GP will calculate their associated Sharpe ratios. Then, to produce new individuals for the next generation, the individuals are randomly selected as parent individuals, with the selection probabilistically biased in favor of the relatively fit individuals with greater Sharpe ratios. Next, the parent individuals are combined by genetic operators, such as crossover and mutation, to create new offspring individuals.

Figure 1 illustrates how the crossover and mutation operators work. As suggested by the green and red box in Panel A to D, the parent individuals in Panels A and B are combined by the crossover operator, and the resulting offspring individuals are shown in Panels C and D. The offspring individuals can also be produced by the mutation operators. For example, the characteristics of X_2 and the constant number of 1 in the green box in Panel A can mutate to X_3 and 2 in Panel E, respectively. Also, the mutation operator can also work on the function node and the whole subtree. For example, the subtree of X_3, shown in the red box in Panel B, can mutate to another subtree of $|sin(X_1)|$ in Panel F.

After applying these genetic operators to produce offsprings, GP will evaluate the fitness of these offspring and parent individuals, and those with greater Sharpe ratios will survive as individuals in the next generation. Afterward, successive generations are iteratively generated in the same way, until the generation number exceeds a pre-defined max generation *Gen*.

Moreover, following Zhang and Bhattacharyya (2004) and Bhowan, Johnston, Zhang and Yao (2012), we adopt an ensemble approach in training our GP model to improve the model robustness and to mitigate overfitting. In particular, since GP has the advantage of parallel computing (Winschel and Krätzig, 2010, and Polachek, Das, and Thamma-Apiroam, 2015), we independently estimate GP for five times, and get $5 \times Pop$ individuals (or models) in total, as each time GP generates *Pop* individuals. Because of the stochastic nature of GP, this helps search for the global optima rather than being accidentally trapped by a local optimum. Finally, we take the average of the top M models with the highest training sample Sharpe ratio as the final model. Although M is set to 5, we have also examined alternative values of 3 and 10 as robustness check.

2.2.4. Hyperparameter tuning

There are two important hyperparameters that control the optimization process of the GP. The first is *Population* (*Pop*), defined as the number of individuals that GP will generate in each generation. The second is *Generation* (*Gen*), used to determine the maximum generation that the evolution will iterate. Clearly, the pair <*Pop*, *Gen*> characterize the searching depth for GP, and have influence on model performance. Since there is no theoretical criterion for the selection of the pair, we follow the most common approach in the literature and select the hyperparameters in a validation sample. The validation sample can be interpreted as a simulated OOS sample to learn about model complexity and hence to mitigate overfitting.

In our paper, the parameter values for Pop are 100, 200, and 400, and those for Gen are 10, 20, and 40.[3] Hence, there are 9 hyperparameter combinations for GP. For a given $< Pop, Gen >$, we use the training sample to estimate the GP model, and use the average of the top M (M=5) model as the model, denoted as $G_{<Pop,Gen>}$. We then evaluate the performance of the 9 models in the validation sample. The optimal model $G^*_{<Pop^*,Gen^*>}$ is the one that earns the highest Sharpe ratio for the spread portfolio in the validation sample. Last, we use the out-of-sample subsample, which is not used for model estimating nor parameter tuning, to examine the OOS performance of the optimal GP model.

2.3. Other methods

For easier comparison, we briefly introduce below other machine learning methods, i.e., those used by Gu, Kelly, and Xiu (2020).

2.3.1. Ridge

Ridge regression imposes an l_2 norm in the standard regression model,

$$\hat{\beta}_{Ridge}(\lambda) = \arg\min_{\beta}\{\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(r_{i,t} - \beta_0 - \sum_{j=1}^{P} X_{i,t-1,j}\beta_j)^2 + \lambda\sum_{j=1}^{P}\beta_j^2\}, \tag{3}$$

where the parameter penalization helps to prevent coefficients from becoming unduly large in magnitude.

2.3.2. Lasso

Lasso regression imposes the l_1 norm,

$$\hat{\beta}_{Lasso}(\lambda) = \arg\min_{\beta}\{\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(r_{i,t} - \beta_0 - \sum_{j=1}^{P} X_{i,t-1,j}\beta_j)^2 + \lambda\sum_{j=1}^{P}|\beta_j|\}, \tag{4}$$

[3] We do not use too large parameters because the GP is computationally extensive. For example, in our applications, it takes about 24 hours to just estimate the model once under the parameter of $< 400, 40 >$. Nevertheless, the chosen values are adequate in robustness checks.

where the parameter penalization helps to force coefficients on some regressors to exactly zero, thereby selecting the most useful variables.

2.3.3. Enet

The elastic net (Enet) model imposes both l_1 and l_2 norms,

$$\hat{\beta}_{Enet}(\lambda, \rho) = \arg\min_{\beta}\{\frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}(r_{i,t} - \beta_0 - \sum_{j=1}^{P} X_{i,t-1,j}\beta_j)^2 + \lambda\sum_{j=1}^{P}(\rho\beta_j^2 + (1-\rho)|\beta_j|)\}. \quad (5)$$

It is clear that $\rho = 1$ corresponds to the Ridge, and $\rho = 0$ corresponds to the Lasso. In our paper, we set $\rho = 0.5$, allowing for the associated Enet takes the advantages of both shrinkage and selection. The hyperparameter λ, in Ridge, Lasso or Enet, is determined by validation sample.

2.4. Dimension reduction models

2.4.1. PCR

Principal components regression (PCR) performs dimension reduction by zeros out coefficients on low variance components. It consists of two steps. In the first step, principal components analysis (PCA) combines the P regressors into a small set of K components ($K \leq P$), which are linear combinations that best preserve the covariance structure among the regressors. Mathematically, the k^{th} PCA component direction v_m solves:

$$\begin{aligned} \underset{v}{\text{maximize}} \quad & Var(Xv) \\ \text{subject to} \quad & ||v|| = 1, \\ & Cov(Xv, Xv_l) = 0, \\ & l = 1, \cdots, k-1. \end{aligned} \quad (6)$$

In the second step, regressions of stock return on the leading components are run to predict future returns.

2.4.2. PLS

Partial least square (PLS) regression performs dimension reduction by directly exploiting co-variation of regressors with the forecast target. In the optimization form, the k^{th} PLS components solves :

$$\begin{aligned} \underset{v}{\text{maximize}} \quad & Cov^2(r, Xv) \\ \text{subject to} \quad & ||v|| = 1, \\ & Cov(Xv, Xv_l) = 0, , \\ & l = 1, \cdots, k-1. \end{aligned} \tag{7}$$

Then, a regression, similar to the PCR case, is run to determine the expected stock returns.

2.5. Neural Networks

Following Gu, Kelly, and Xiu (2020), we construct the neural networks for our study in the same way. We consider the architectures with up to five hidden layers. The The shallowest neural network, denoted as NN1, has a single hidden layer of 32 neurons, NN2 has two hidden layers with 32 and 16 neurons, respectively; NN3 has three hidden layer with 32, 16, and 8 neurons, respectively; NN4 has four hidden layer with 32, 16, 8, and 4 neurons, respectively; and NN5 has four hidden layer with 32, 16, 8, 4, and 2 neurons, respectively. The nonlinear activation function is also the same rectified linear unit (ReLU) function for all nodes, defined as

$$ReLU(x) = \begin{cases} 0 & \text{if x} \leq 0 \\ x & \text{otherwise,} \end{cases}$$

Moreover, we also employ the stochastic gradient descent (SGD) to estimate the neural network weight parameters to minimize the mean squared errors. We denote the expected return generated by NN_l $(l = 1, 2, 3, 4, 5)$ for stock i in month t as $ER^{i,t}_{NN_l}$.

3. Main results

In our GP applications below, we split the full sample, from 1945:01 to 2019:12, into three subsamples. The training subsample from 1945:01 to 1980:12 is used to train the machine learning models. The validation subsample from 1981:01 to 1990:12 is used to choose the hyperparameters in these models.[4] The out-of-sample (OOS) subsample from 1991:01 to 2019:12 is used to evaluate the models' predictive performance.

3.1. Spread portfolios

Table 1 reports the OOS performance of the value-weighted decile spread portfolios sorted by the expected return of various models. It is interesting that there are not much differences in the linear models, whose annualized Sharpe ratios range from 0.68 to 0.81. Consistent with Gu, Kelly, and Xiu (2020), the neural networks tend to outperform the linear machine learning methods, achieving the highest annualized Sharpe ratio of 0.96. In contrast, the GP earns the best annualized Sharpe ratio up to 1.32, more than 37% greater than the next best level of 0.96.

In terms of average returns, the GP also performs the best, with a monthly return of 1.71%, while the next largest average return, achieved by NN2, is only 1.22%. The linear models also have lower returns at about 1.00%. Moreover, in terms of skewness, the GP enjoys a positive skewness of 1.17, while the skewness is of the linear models are lower than 0.5. However, NN2 has the largest skewness of 1.50, but it is too volatile and does not even have the highest Sharpe ration among the neural networks models.

Table 2 reports the sub-sample results before and after 2003, a year when Green, Hand, and Zhang (2017) detect a major structural break for predicting the cross-sectional returns. Panel A shows that during the pre-2003 sub-period, the GP yields the highest spread return (2.93%) and the greatest Sharpe ratio (1.89). Interestingly, the linear models perform almost as well as the average of the neural network models. Hence, during this "easier" to predict periods as identified by Green, Hand, and Zhang (2017), existing machine learning methods, linear or nonlinear, do not seem to make much differences. This is because, though NN5 does the best, it is ex ante difficult to select NN5 out of all the models. Nevertheless, the GP still stands out and performs the best as expected.

Panel B reveals a much different pattern. In this "difficult" to predict period, all linear models fail to generate significant average returns on the spread portfolios, and even three of the 5 neural networks models fail. In contrast, the GP earns an economically and statistically significant monthly

[4] Following Gu, Kelly, and Xiu (2020), we do not choose cross-validation to maintain the temporal ordering of the data for prediction.

average return of 0.72%. In terms of Sharpe ratio, it has the highest of 0.77, exceeding the next best level of 0.55, achieved by the best neural network model NN4, by 40%.

In short, empirically both in the subperiods and in the entire out-of-sample period, the GP achieves what it is designed for, to maximize the Sharpe ratio. This is one of the most important measures investors or fund managers rely upon in assessing a portfolio strategy.

3.2. Controlling for other models

Since GP and other models exploit different predictive information from the same characteristic set, it is of interest to examine which of them can provide incremental predictive power beyond the use of the other. Consider, for example, how to measure the incremental predictive power of the GP conditional on any other model. A simple approach is to regress the expected stock return generated by the GP on those generated by a given other model, and then sort the residuals into decile portfolios to see whether the new long-short spread portfolio can earn significant profits. Clearly, if the predictive power of the GP is subsumed by the given model, we should not be able to observe any profitable pattern in the resulting spread portfolio.

Panel A of Table 3 reports the results. After controlling for the expected return generated by any of the other models, the GP still produces highly significant spread returns in every single case. The results clearly indicate that the GP has certain unique predictability which cannot be replaced by any of the other models.

Conversely, we also examine the predictive power of any other model after controlling that of the GP. Panel B of the table shows that, after controlling for the GP, none of the other machine learning models can produce significant spread returns. The results suggest that the predictability of all the other models are subsumed by the GP.

3.3. Information uncertainty

To understand under what conditions where the GP and other methods differ, we focus on information uncertainty, and, following Zhang (2006), use idiosyncratic volatility (IVOL) to proxy for it. In particular, we construct an IVOL index, defined as the average of IVOL of individual stocks in each month, to reflect the information uncertainty at the market level. The greater the

IVOL index, the greater the information uncertainty across stocks.

We carry out the following time-series predictive regression,

$$\Delta R_t = \beta_L Low_{t-1}^{Vol} + \beta_H High_{t-1}^{Vol} + \beta MKT_t + \epsilon_t, \tag{8}$$

where ΔR_t is the return of the GP spread minus that of other models, Low_{t-1}^{Vol} and $High_{t-1}^{Vol}$ are dummy variables indicating low- and high-IVOL periods of previous month, as classified based on the median level of the IVOL index. The parameters of interest are β_L and β_H, indicting either the low- or high-IVOL period or both matter for the GP outperformance.

Table 4 reports the results. The slope β_H is greater than β_L for all of the ten models. Moreover, β_L is insignificant for nine of the ten spreads, while β_H is significant except for the NN2 case. On average, β_H is 1.02 with a significant t-statistic of 2.43, whereas β_L is much lower at 0.31 with a weak t-statistic of only 1.20. The results suggest that the improved performance of GP over other models is mainly attributed to the high-IVOL periods, during which the information uncertainty level is high. From an investment perspective, it is more difficult and hence more important to predict returns more accurately with greater information uncertainty. The GP appears to help exactly to do it in comparison with other methods.

3.4. GP factor

In this subsection, we consider a factor formed based on the GP and compare it with various well known factor models in the literature.

Following Fama and French's (1993) factor formulation approach, we construct a GP factor (GPF) based on a 2 × 3 double sorting on size and ER_{GP}. The factors for the comparison are: the CAPM, the Fama and French (1993) 3-factor model (FF-3), Fama and French (2015) 5-factor model (FF-5), Hou, Xue, and Zhang (2015) 4-factor model (HXZ-4), Stambaugh and Yuan (2016) mispricing-factor model (SY-4), and Daniel, Hirshleifer, and Sun (2020) behavioral-factor model model (DHS-3), with data from their websites.

Table 5 reports the results.[5] The GPF earns the greatest monthly average return of 1.20%, almost doubling the next best factor of 0.69%. Its annualized Sharpe ratio, 1.75, is also the

[5] We use the earliest sample ending month of the data, SY-4, as our last period, 2016. The results are similar if we use a different end period for other available data.

maximum, almost doubling the next best too, 0.86. It has large skewness of 0.85, indicating a desirable positive return pattern, whereas its kurtosis is about the average, with tails neither too fat or skinny. Panel B provides the correlation matrix of the factors. It shows that the GPF has low correlation with the well known factors.

Although the GPF other factors and has little correlation with each one of them, it does not rule out the hypothesis that a portfolio of other factors can replicate the performance of the GP. To test this hypothesis, we carry out six spanning tests: Wald test under conditional homoskedasticity, Wald test under independent and identically distributed (IID) elliptical distribution, Wald test under conditional heteroskedasticity, Bekerart-Urias spanning test with errors-in-variables (EIV) adjustment, Bekerart-Urias spanning test without the EIV adjustment and DeSantis spanning test (see Kan and Zhou, 2012).

Panel A of Table 6 provides the results for the spanning tests. The spanning hypothesis is strongly rejected, indicating that the GPF can add substantial investment value to existing factor models. Barillas and Shanken (2017) show that investment value is related to model comparison. If a new factor can add substantial Sharpe ratio to an existing factor model, an extended model by adding the factor must outperform the existing model in explaining asset returns, irrespective of the test assets. Along this line, we conduct the Sharpe ratio test to compare the Sharpe ratios(Sh^2) of the various models with and without the GPF.

Panel B of Table 6 reports the results. It is apparent that adding the GP factor substantially improves the Sh^2 for all of other models. For example, the Sh^2 for CAPM increases significantly from 0.026 to 0.265, where the significance level is computed based on a studentized bootstrap procedure due to Ledoit and Wolf (2008). The virtually zero p-values cross the models suggest that the GP factor can improve the pricing ability of existing models substantially.

3.5. Risk-adjusted performances

Table 7 reports the alphas of the spread portfolios of the machine learning methods under different factor models. The first 6 rows show that all of the 11 spread portfolios earn highly significant alphas with respect to all the well known existing factor models: the CAPM, FF-3, FF-5, XHZ-4, SY-4, and DHS-3, indicating that existing factor models cannot explain the predicted returns of the machine learning methods. In fact, the magnitude of the alphas are much larger than almost all of those classic anomalies in the literature (see, e.g., Hou, Xue, and Zhang (2015)).

In contrast, as shown by the last row, all the alphas become insignificant relative to the extended CAPM with the GPF as the added factor. Indeed, the largest alpha is now only 30 basis points, while the average alphas exceed 1% previously. The results are similar if the GPF is added to any other factor models, suggesting that the GP factor improves substantially the pricing ability of existing models.

4. Robustness

4.1. Alternative characteristics

In this subsection, we examine the performance of the machine learning methods when applied to another 15 characteristics, the typical one used by Lewellen (2015). Different to the characteristic set used in the main results which relies heavily on technical indicators, this new data set are mainly fundamental variables: size, book-to-market ratio, the growth in split-adjusted shares outstanding from month -36 to month -1, accrual, ROA, annual growth of total asset, dividend yield, the growth in split-adjusted shares outstanding from month -12 to month -1, market beta, the return from month -12 to month -2, the return from month -36 to month -13, return volatility, turnover, debt-to-price ratio, and sales-to-price ratio.[6] Since this characteristic set uses the accounting data from the Compustat, the sample period is much shorter and starts in January 1976. Hence, we set below the training sample from 1976:01 to 1995:12, the validation sample from 1996:01 to 2000:12, and the OOS sample from 2001:01 to 2019:12.

Table 8 reports the OOS performance of the spread portfolios. It is important to note that none of the linear models can generate significant returns, although they still have positive returns and still outperform substantially the OLS model (unreported). The neural network models, however, do yield significant gains in 3 out of 5 cases. In contrast, GP still performs the best, earning the greatest significant return of 0.99%, improving the next best one by about 60%. Its Sharpe ratio is the largest, 0.74, as expected, which improves the nest best one by about 70%. In contrast to

[6] The detailed constructions of these variables are provided by Lewellen (2015) and are also available in the Online Appendix.

the previous set of characteristics, the new one has less predictability on the cross-section of the stock returns. In this case, the GP outperforms other methods even more in terms of percentage improvement.

4.2. International markets

In this subsection, we examine the performance of the GP in the major international stock markets. As emphasized by Schwert (2003), the use of alternative data sets is one way to mitigate the concern of data-snooping. For brevity, we focus on other G7 countries: the UK, Canada, Japan, Italy, France, and Germany.

There is one unique feature in our applications to the international markets. Instead of re-estimating the machine learning models in each market, we directly apply all of them estimated in the US directly to other markets. Since the data in other markets are not used for neither model estimation nor parameter tuning, they offer a perfect setting to examine the OOS performance.

Table 9 reports the results. There are two notable patterns. First, the GP substantially outperforms other machine learning methods, achieving the largest Sharpe ratio in all the 6 markets. For example, in UK, it has a Sharpe ratio of 1.09 with an average monthly return of 1.69%. Although the Sharpe ratio from NN2 is high, but the average across the other methods is about 30% lower than the GP. The result is echoed by the average cross the markets, reported in Panel G. The second pattern is that linear models perform well in the international markets relative to the nonlinear neural networks. This differs from the US market where the latter dominates the former. The pattern is interesting and puzzling, and is a subject of future research.

In short, GP performs well not only in the US, but also internationally in other G7 markets, even with the same model estimated in the US. The strong performance of the GP internationally indicates that the method captures salient features of the market and is robust to alternative data sets.

4.3. Alternative parameters

For the main results, the GP model is estimated under the hyperparameters $< Pop, Gen >=< 200, 40 >$, which is determined by the validation sample. We now further examine the robustness under alternative parameters.

4.3.1. In-sample performance evolution

Consider alternative parameters for *Pop*: 100, 200, and 400, and that for *Gen*: 10, 20, and 40. There are a total of 9 sets of the hyperparameters. For a given $< Pop, Gen >$, we independently estimate GP for 5 times, and get $5\times Pop$ models (individuals) in total. We use the average of the top M models with the highest Sharpe ratios in the training sample as the final model. Note that $M = 5$ in our main results, and here we also consider alternative values of 3 and 10.

Table 10 reports the results with the alternative $< Pop, Gen >$'s and M's. There are a few interesting facts. First, since $< Pop, Gen >$ characterizes the searching depths for GP, the Sharpe ratio in the training sample increases with $< Pop, Gen >$. For example, for *Pop*=100 and $M = 5$, , the annualized Sharpe ratio grows from 2.04 to 2.85 as *Gen* increases from 10 to 40. Second, while the training sample Sharpe ratio generally increases with *Pop*, the effect is weaker. For example, for *Gen*=10 and $M = 5$, the Sharpe ratio increases from 2.04 to only 2.28 as *Pop* increases from 100 to 400. In particular, for a deeper *Gen* of 40, the Sharpe ratios are almost flat among different *Pop*. In general, the in-sample performance increases with $< Pop, Gen >$, but is more sensitive with respect to *Gen*.

Third, by comparing the validation Sharpe ratios of various parameters in Panel A, we find that the parameter $< Pop, Gen >$ of $< 200, 40 >$ achieves the largest validation sample Sharpe ratio of 2.66, supporting our earlier parameter choice. This choice also achieves the best OOS performance: the spread portfolios earns the largest annualized Sharpe ratio of 1.32, as shown earlier in Table 1. Forth, although the objective of GP is to maximize the spread portfolio's Sharpe ratio, we also report the average return of the spread portfolios for other parameters. In general, the spread return exhibits similar patterns as the Sharpe ratios. For example, the training sample return increases with $< Pop, Gen >$. The largest validation return is also achieved at $< Pop, Gen >=< 200, 40 >$. Fifth, Panel B and Panel C show similar patterns to Panel A, indicating that the performance is robust to M. Overall, the results are economically not too far apart even though the parameter values are substantially different.

4.3.2. Sharpe ratio evolution

To understand further the performances under the alternative parameters, we examine now how the Sharpe ratio changes in the GP algorithm as the population grows.

Figure 2 presents the plots of the Sharpe ratios. Consider Figure A. Since the *Pop* is 100, the max number in the X-axis is 100 for the individual. The blue curve plots the training sample Sharpe ratios averaged over the five estimations. The green curve and the red line one and those for the validation sample and OOS sample, respectively. Since we sort the individuals (models) by their training sample Sharpe ratio, the blue curve shows a monotonic increasing pattern. It is clear that OOS performance is weaken in comparison with in-sample and validation. However, it does share the same pattern, indicating that greater in-sample Sharpe ratios tend to generate stronger predictability in the OOS sample.

As *Gen* increases, the green and red line become less volatile. For example, in comparison Figures B and C, the green and red line are much flatter, suggesting that the solution converges to a stable OOS performance. In the Online Appendix, we provide detailed results to show that the model performance volatility decreases with *Gen*.

5. What drives GP's performance?

In this section, we explore the reasons why the GP can outperform the other machine learning methods.

5.1. Objective function

An obvious question is whether the objective function plays a role in the performance. To examine this, instead of maximizing the Sharpe ratio as we did before, we now consider the objective of minimizing the conventional mean squared error (MSE) of the predicted returns. We denote this model as GP_{MSE}, and denote the previous GP model of maximizing the Sharpe ratio as GP_{SR}.

Table 11 compares the performance of the two models. The spread portfolio of GP_{MSE} yields an average monthly return of 1.44% with an annualized Sharpe ratio of 1.04. The performance is comparable to and slightly better than that of the neural network (NN2) in Table 1, which earns a mean return of 1.22% and a Sharpe ratio of 0.92. However, it is important to note that GP_{MSE} is dominated by GP_{SR}. Since the Sharpe ratio is a comprehensive metric which considers the trade-off between return and risk, the results show that it does do better in terms of both return and volatility. Indeed, compared with GP_{MSE}, GP_{SR} not only earns a higher spread return of 1.71%, but also produces a lower volatility of 4.47%. As a result, GP_{SR} yields a greater Sharpe ratio of 1.32, about 30% larger than that of GP_{MSE}. In addition, GP_{SR} also earns a much higher positive skewness of 1.17, while that for GP_{MSE} is only 0.37.

As an alternative way to compare GP_{MSE} with GP_{SR}, we regress the expected returns generated by the two GP models on each other, and then examine the performance of the resulting spread portfolio sorted by the residuals. The right panel of Table 11 reports the results. Controlling for GP_{SR}, GP^{ω}_{MSE} generates a negligible spread return of 0.27% with a weak t-statistic of only 0.83, indicating that the predictability of GP_{MSE} is subsumed by GP_{SR}. In contrast, controlling for GP_{MSE}, GP^{ω}_{GP} still earns a persistent spread return of 0.91% with a significant t-statistic of 5.69, suggesting that GP_{SR} contains additional predictability uncorrelated to GP_{MSE}.

In short, compared with the conventional MSE-based models, the reason for the economic gains of using our proposed GP model arises from maximizing the spread portfolio's Sharpe ratio directly. By considering both return and risk, the metric produces much higher Sharpe ratio and outperforms the MSE-based models of the GP and other machine learning methods.

5.2. Linearity vs nonlinearity

It is well known that the standard MSE estimator of the parameters is efficient if the data are normally and independent and identically distributed. In this case, there is likely little difference between MSE minimization and Sharpe ratio maximization. However, when the true data have nonlinearity (see, e.g., Freyberger, Neuhierl, and Weber, 2020), the difference will likely be large. We show that this is indeed the case via simulations.

Consider the linear case first. Following Freyberger, Neuhierl, and Weber (2020), we simulate data from a linear model with a set of fixed predictors:

1. Assume the "true" predictor set Z consists of $Size$, $SREV$, MOM, and $LREV$.

2. Regress the stock return R on the assumed predictor set Z in a panel regression, pooled over the entire sample from 1945 to 2019. Then, decompose R into the fitted part ($\hat{R}_{i,t}$) and the residual ($\epsilon_{i,t}$).

3. Generate returns according to $\tilde{R}_{i,t} = \hat{R}_{i,t} + \tilde{\epsilon}_{i,t}$, where $\tilde{\epsilon}_{i,t}$ is resampled with replacement from the empirical residuals in step 2. To generate the residuals in a particular month t, we first draw a random time period, say month s, from which we sample the residuals. Moreover, to ensure we sample from the distribution with zero means, we re-center the original residuals each month.

4. Based on the simulated return $\tilde{R}_{i,t}$ from step 3 and the predictor set for investment use, Q, consisting of $Size$, $SREV$, MOM, and $LREV$, we estimate the GP and other benchmark models, and examine their OOS performance.

   Note that Q is the same as Z from step 1, and this is equivalent to assuming the true predictors are known to investors.

5. Redo steps 3-4 for 500 times.

For the nonlinear case, the simulation procedure is similar, except that we add the interaction terms in the true predictor set. That is, Z now consists of 10 variables: $Size$, $SREV$, MOM, and $LREV$, as well as 6 pairwise interaction terms of these four variables. Suppose the true data process is generated by this new predictor set. We then estimate the coefficients in a panel regression. In particular, we scale the slopes on the interaction terms to make them comparable to those of the four original predictors.[7]

Note that in the linear simulation, the true predictor set Z is the same as the indicator set Q, which is the input data for training the models. In this case, the linear model is the true model and hence is expected to perform well. In the nonlinear simulation, however, the true predictor set Z include the nonlinear interaction effects, while we still use the same indicator set Q for forecasting. Since GP captures nonlinearity, we expect that GP will show its strength in the nonlinear simulation.

[7] We multiply the slopes on the interaction terms by 8, and also get qualitatively robust results under alternative values.

Table 12 reports the average OOS statistics in the linear and nonlinear simulations. In particular, we also consider a special benchmark model, the fitted return $\hat{R}_{i,t}$ from step 2 in the simulation procedure. $\hat{R}_{i,t}$, by construction, contains *all* the predictability in the simulated return $\tilde{R}_{i,t}$, and hence, it can be interpreted as the optimal model.

The left panel reports the Sharpe ratios. In the linear simulation, the model of $\hat{R}_{i,t}$ produce a Sharpe ratio of 1.33. GP earns a Sharpe ratio of 1.24, which is only slightly less than the optimal model. Meanwhile, consistent with our prediction, all the linear models performs well in the linear simulation, and yield Sharpe ratios around 1.30, very close to that of $\hat{R}_{i,t}$. It is not surprising to see the good performance of the linear models, because, in the linear simulation, these models are the true models and hence are expected to achieve similar performance with $\hat{R}_{i,t}$.

In the nonlinear simulation, as we include the nonlinear interaction terms, which increases the overall predictability, the Sharpe ratio of the optimal model $\hat{R}_{i,t}$ increases substantially by 153%, from 1.33 to 3.36. It is important to note that the Sharpe ratio of GP also grow significantly by 148%, from 1.24 to 3.08, which is close to that of the $\hat{R}_{i,t}$. On the contrary, although linear models also produce better performance in the nonlinear case, the resulting Sharpe ratio of around 2.05 is much lower than that for GP and $\hat{R}_{i,t}$, and the rowth rate of 50% is also much smaller than that for GP.

The right panel reports the mean returns. In the linear case, the return of the linear models are very close to that of $\hat{R}_{i,t}$. Since the objective of our GP is to maximize the Sharpe ratio rather than the return, GP earns a little bit lower OOS returns than other linear models in the linear case. But in the nonlinear case, GP yields higher return than other linear models.

In short, consistent with our prediction, while linear models perform well in the linear simulation, GP outperforms linear models in the nonlinear simulation. This evidence suggests that the ability to exploit nonlinear predictability is another source for GP's good performance.

5.3. Bootstrap with different sample size

In this subsection, we carry out bootstrap analysis to compare the performance of GP and NN under different sample sizes.

We choose two different sample size. In the first simulation, in each month t, we resample stocks so that the stock number in the simulated data is only *half* of the actual stock number. In the second simulation, we do the same but double the stock number in the simulatiton each month. Then, based on the simulated data, we estimate GP and NN models, and examine their OOS performance.

Table 13 reports the average OOS statistics for the spread portfolios of GP and NN.[8] The performance of GP is robust to the sample size. For example, in both sample sizes, GP earns an annualized Sharpe ratio of about 1.00 and a mean return of 1.60%. In contrast, the performance of NN is sensitive to the sample size. In particular, when we reduce the sample size to half, the OOS performances of NN become substantially worse. As the sample size increases, they become better. In short, GP has stable performances under reasonable sample sizes, while NN is more sensitive to it. This provides another reason (besides objective functions) why, although both are nonlinear models, the GP has better performances previously than the NNs.

6. Conclusion

In this paper, we propose to maximize the Sharpe ratio of a portfolio via genetic programming (GP), one of the machine learning tools applied here the first time for the study of the cross-section of stock returns. Our approach directly optimizes the Sharpe ratio by searching a function that maps from the stock characteristics to the expected stock returns in a large functional space. We find that the performance of the GP spread portfolio in the cross-section outperforms substantially the usual MSE-based models, such as ridge, lasso, Enet, PCR, and PLS. It also outperforms significantly the more powerful neural networks by subsuming their predictability. While existing factor models fail to explain the performance of the MSE-based machines learning methods, a single factor based on the GP fully captures all their spread portfolios. The performance of the GP is robust to alternative parameters, different characteristics, and international data sets. We find further that the good performance of the GP is due to its economic objection optimization, and it is less sensitive to sample size than the neural networks.

Our empirical evidence suggests that it is important to apply machine learning tools to maximize economic objectives, beyond the scope of the traditional model fitting. Since the Sharpe ratio is one

[8] The time for estimating GP model increases with sample size. To save time, the parameter $< Pop, Gen >$ for GP in this analysis is $< 100, 10 >$. We repeat the simulation for 10 times, and the table reports the average statistics.

of the most important performance measure of a trading strategy, the present framework can be applied in many areas to maximize the Sharpe ratio. It will not only be useful for a fund managers to improve investment performance in various asset classes, but also be useful for researchers to identify potentially the largest anomalies in currencies, corporate bonds or commodities. These are interesting issues for future research.

References

Allen, F., Karjalainen, R.1999. Using genetic algorithms to find technical trading rules. Journal of Financial Economics 51, 245-271

Barillas, F., Shanken, J., 2017. Which alpha?. Review of Financial Studies 30, 1316-1338.

Bhowan, U., Johnston, M., Zhang, M., Yao, X., 2012. Evolving diverse ensembles using genetic programming for classification with unbalanced data. IEEE Transactions on Evolutionary Computation, 17, 368-386.

Brogaard, J., Zareei A., 2018. Machine learning and the stock market. Workingpaper.

Chinco, A., Clark-Joseph, A.D., Ye, M., 2019. Sparse signals in the cross-section of returns. Journal of Finance 74, 449-492.

Daniel, K., Hirshleifer, D., Sun, L., 2020. Short-and long-horizon behavioral factors. Review of Financial Studies 33, 1673-1736

Darwin, C., 2004. On the origin of species, 1859. Routledge.

DeBondt, W.F.M., Thaler, R., 1985. Does the stock market overreact? Journal of Finance 40, 783-805.

Dempster, M.A. and Jones, C.M., 2001. A real-time adaptive trading system using genetic programming. Quantitative Finance, 1, 397-413.

Dunis, C.L., Laws, J., Middleton, P.W., Karathanasopoulos, A., 2015. Trading and hedging the corn/ethanol crush spread using time-varying leverage and nonlinear models. The European Journal of Finance, 21, 352-375.

Fama, E.F., French, K.R., 1993. Common risk factors in the returns on stocks and bonds. Journal of financial economics 33, 3-56.

Fama, E.F., French, K.R., 2015. A five-factor asset pricing model. Journal of financial economics 116, 1-22.

Feng, G., Giglio, S., Xiu, D., 2020. Taming the factor zoo: A test of new factors. Journal of Finance, 75, 1327-1370.

Filippou, I., Rapach, D., Taylor, M.P., Zhou, G., 2020. Exchange Rate Prediction with Machine Learning and a Smart Carry Portfolio. Available at SSRN 3455713.

Freyberger. J., Neuhierl A., Weber, M., 2020. Dissecting characteristics nonparametrically, Review of Financial Studies, 33, 2326-2377.

Green, J., Hand, J.R., Zhang, X.F., 2017. The characteristics that provide independent information about average us monthly stock returns. The Review of Financial Studies, 30, 4389-4436.

Gu, S., Kelly, B., Xiu, D., 2020. Empirical asset pricing via machine learning. Review of Financial Studies, 33(5), 2223-2273.

Guo, X., Lin, H., Wu, C., Zhou, G., 2019. Investor Sentiment and the Cross-Section of Corporate Bond Returns. Available at SSRN 3223846.

Han, Y., He, A., Rapach, D., Zhou, G., 2020. Firm characteristics and expected stock returns. Available at SSRN 3185335.

Han, Y., Zhou, G., Zhu, Y. 2016. A trend factor: Any economic gains from using information over investment horizons?. Journal of Financial Economics 1222, 352-375.

Hou, K., Xue, C., Zhang, L., 2015. Digesting anomalies: An investment approach. Review of Financial Studies 28, 650-705.

Jegadeesh, N., Titman, S., 1993. Returns to buying winners and selling losers: implications for stock market efficiency. Journal of Finance 48, 65-91.

Kan, R., Zhou, G., 2012. Tests of mean-variance spanning. Annals of Economics and Finance 13, 139-187.

Koza, J.R., 1992. Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge

Kozak, S., Nagel, S., Santosh, S., 2020. Shrinking the cross-section. Journal of Financial Economics, 135, 271-292.

LeBaron, B., Arthur, W.B., Palmer, R., 1999. Time series properties of an artificial stock market. Journal of Economic Dynamics and control, 23, 1487-1516.

Ledoit, O., Wolf, M., 2008. Robust performance hypothesis testing with the Sharpe ratio. Journal of Empirical Finance 15, 850-859.

Lehmann, B.N., 1990. Fads, martingales and market efficiency. Quarterly Journal of Economics 105, 1-28

Lewellen, J. 2015. The Cross-section of Expected Stock Returns. Critical Finance Review 4, 1-44.

Lo, A.W., MacKinlay, A.C., 1990. When are contrarian profits due to stock market overreaction? Review of Financial Studies 3, 175-205.

Neely, C., Weller, P., Dittmar, R., 1997. Is technical analysis in the foreign exchange market profitable? A genetic programming approach. Journal of financial and Quantitative Analysis, 32, 405-426.

Newey, W.K., West, K. D., 1987. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. Econometrica 55, 703-708

Nordhaus, W.D., 2001. The progress of computing. Available at SSRN 285167.

Polachek, S.W., Das, T., Thamma-Apiroam, R., 2015. Micro-and macroeconomic implications of heterogeneity in the production of human capital. Journal of Political Economy, 123, 1410-1455.

Ready, M.J., 2002. Profits from technical trading rules. Financial Management, 43-61.

Schwert, G.W., 2003. Anomalies and market efficiency. In: Constantinides, G.M., Harris, M., Stulz, R.M. (Eds.), Handbook of the Economics of Finance, 1. Elsevier, Amsterdam, Netherlands, pp. 939-974. chap. 15.

Stambaugh, R.F., Yuan, Y., 2016. Mispricing factors. The Review of Financial Studies 30, 1270-1315.

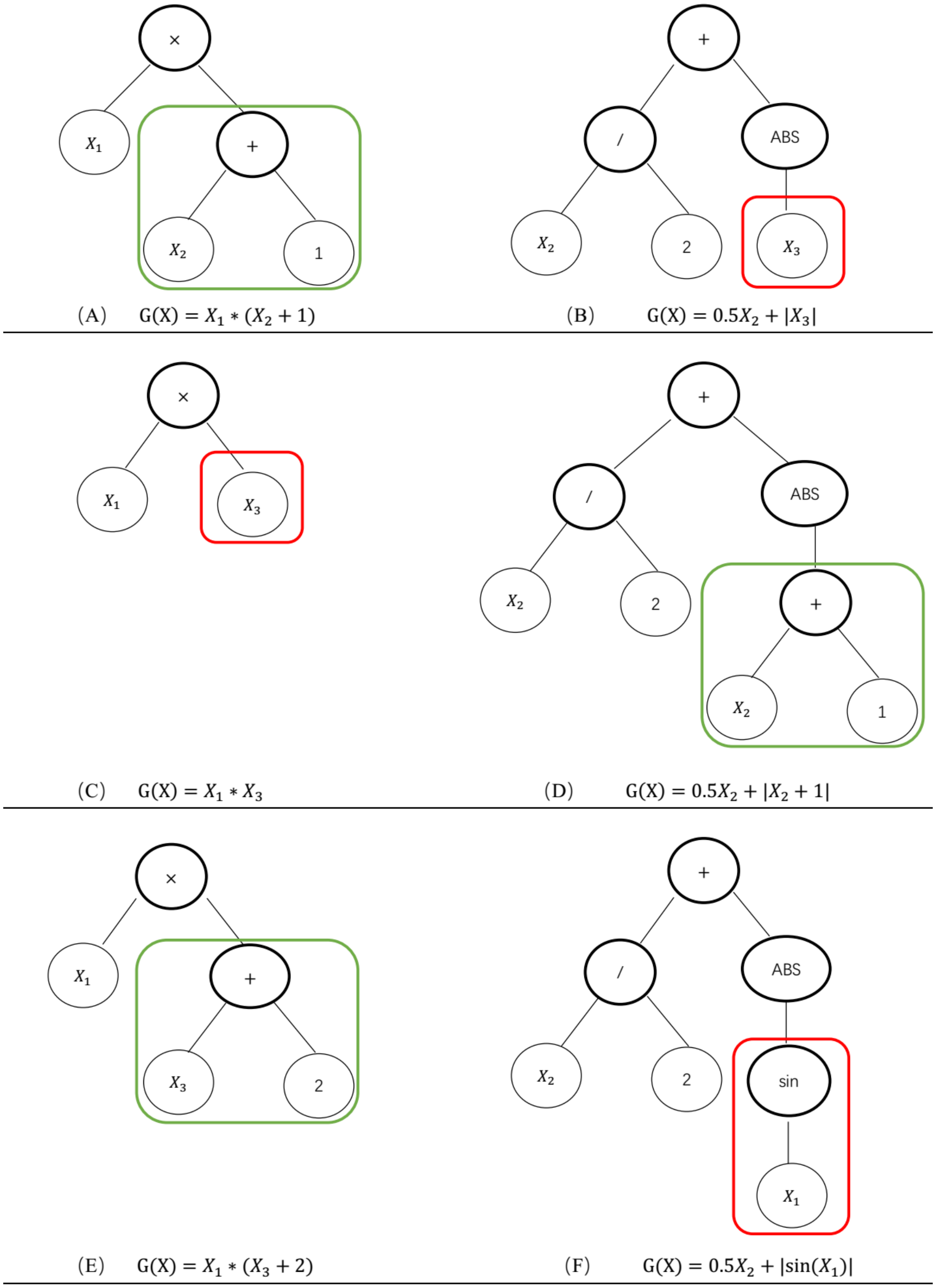
Winschel, V., Krätzig, M., 2010. Solving, estimating, and selecting nonlinear dynamic models without the curse of dimensionality. Econometrica, 78, 803-821.

Zhang, X.F., 2006. Information uncertainty and stock returns. Journal of Finance, 61, 105-137.

Zhang, Y., Bhattacharyya, S., 2004. Genetic programming in classifying large-scale data: an ensemble method. Information Sciences, 163, 85-101.

Figure 1: **Tree-structured representation and genetic operators**

This figure illustrates the tree-structured individuals and the genetic operators of crossover and mutation. The parents individuals in Panel A and B are combined by the crossover operator, and the resulting offspring individuals are shown in Panel C and D. The offspring individual in Panel E (F) is produced by the mutation operator from the individual in Panel A (B).



(A) $G(X) = X_1 * (X_2 + 1)$

(B) $G(X) = 0.5X_2 + |X_3|$

(C) $G(X) = X_1 * X_3$

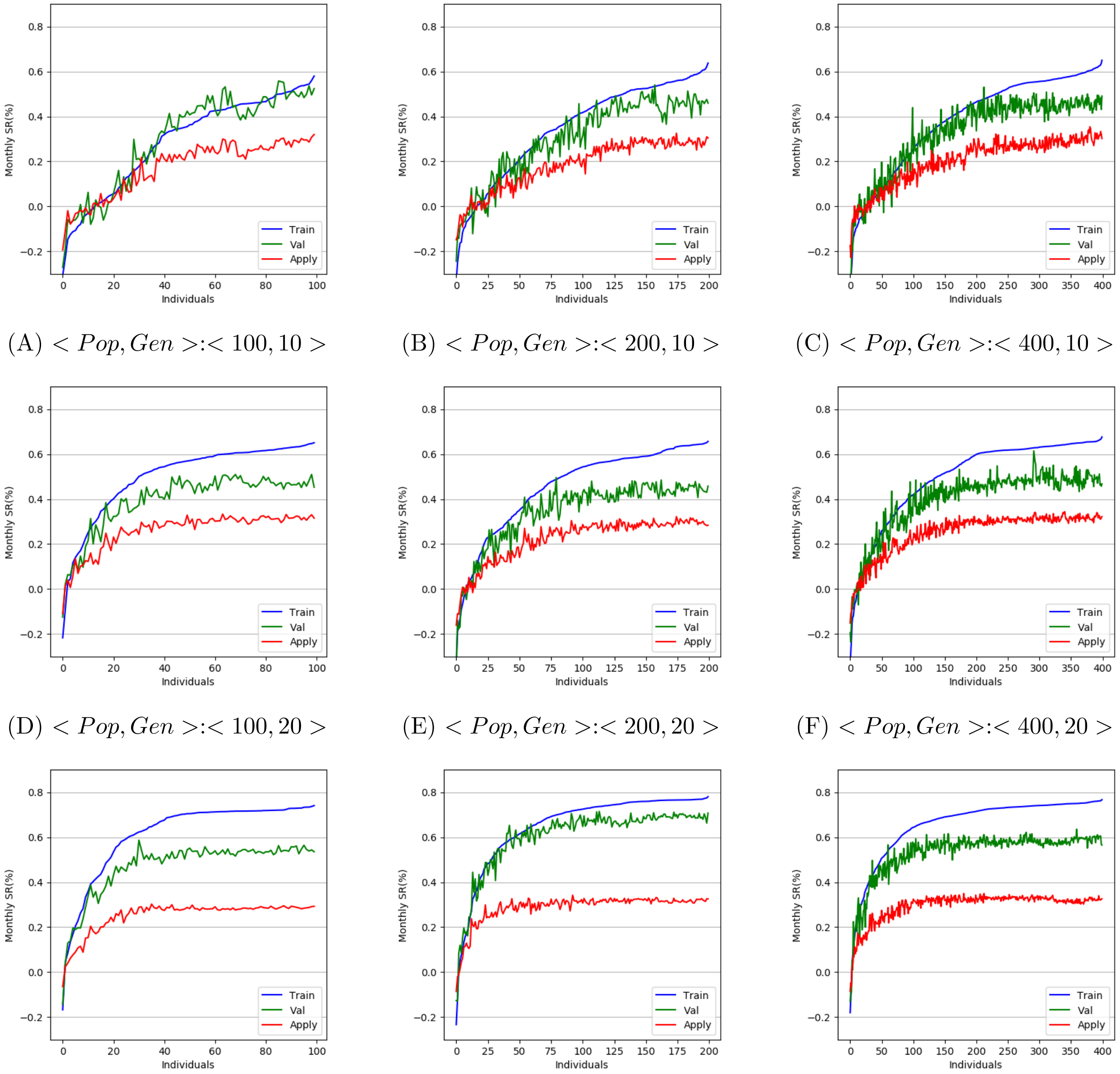
(D) $G(X) = 0.5X_2 + |X_2 + 1|$

(E) $G(X) = X_1 * (X_3 + 2)$

(F) $G(X) = 0.5X_2 + |\sin(X_1)|$

Figure 2: **GP's performance under various hyperparameters**

This figure shows the monthly Sharpe ratio of the spread portfolios generated by GP under various parameters. For a given set of the parameter $< Pop, Gen >$, we independently estimate GP model using training sample for five times and get Pop individuals each time. We sort the individuals within each time by their associated Sharpe ratio in the training sample. The blue (green, or red) lines show the Sharpe ratios of the individuals average over the five estimations in the training (validation, or OOS) sample.



(A) $< Pop, Gen >:< 100, 10 >$

(B) $< Pop, Gen >:< 200, 10 >$

(C) $< Pop, Gen >:< 400, 10 >$

(D) $< Pop, Gen >:< 100, 20 >$

(E) $< Pop, Gen >:< 200, 20 >$

(F) $< Pop, Gen >:< 400, 20 >$

(G) $< Pop, Gen >:< 100, 40 >$

(H) $< Pop, Gen >:< 200, 40 >$

(I) $< Pop, Gen >:< 400, 40 >$

Table 1

Spread portfolios

The table reports the summary statistics for the decile spread portfolios generated by the GP and other models. For each model, we report the average monthly return in percentage points, the Newey-west (1987) robust t-statistic, the annualized Sharpe ratio (*Sharpe*) and the skewness (*Skew*). The sample period is from 1991:01 to 2019:12.

	GP	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
Low	0.08	0.58	0.54	0.51	0.61	0.57	0.71	0.68	0.67	0.43	0.37
2	0.57	0.67	0.69	0.77	0.75	0.70	0.80	0.76	0.91	0.68	0.67
3	0.57	0.90	0.90	0.81	0.81	0.89	0.86	0.92	0.87	0.78	0.76
4	0.50	0.93	1.00	1.00	0.98	0.95	0.92	0.96	1.04	0.86	0.87
5	0.74	1.14	1.12	1.10	1.18	1.13	1.16	1.09	1.17	0.85	0.91
6	1.06	1.12	1.14	1.16	1.18	1.13	1.08	1.30	1.03	0.92	0.85
7	1.09	1.30	1.41	1.36	1.21	1.29	1.12	1.29	1.16	1.17	0.94
8	1.53	1.42	1.38	1.40	1.51	1.41	1.16	1.38	1.40	1.18	1.08
9	1.49	1.58	1.46	1.52	1.46	1.61	1.31	1.41	1.36	1.33	1.42
High	1.79	1.64	1.67	1.61	1.53	1.61	1.66	1.90	1.80	1.56	1.47
H-L	**1.71*****	1.06***	1.13***	1.10***	0.92***	1.04***	0.95***	1.22***	1.13***	1.12***	1.10***
t-stat	**7.12**	3.99	4.35	4.27	3.66	3.92	4.07	4.93	4.22	5.07	5.15
Sharpe	**1.32**	0.74	0.81	0.79	0.68	0.73	0.76	0.92	0.78	0.94	0.96
Skew	**1.17**	0.45	0.24	0.34	0.45	0.48	0.10	1.50	0.99	0.38	0.58

Table 2

Subperiod performance

The table reports the summary statistics for the decile spread portfolios generated by the GP and other models over two subperiods. For each model, we report the average monthly return in percentage points, the Newey-west (1987) robust t-statistic, the annualized Sharpe ratio (*Sharpe*) and the skewness (*Skew*). The sample period in Panel A is from 1991:01 to 2003:12, and in Panel B is from 2004:01 to 2019:12.

	GP	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
Panel A: 1991:01-2003:12											
H-L	**2.93*****	2.03***	2.10***	1.93***	1.74***	2.01***	1.60***	2.25***	1.88***	1.85***	2.04***
t-stat	**6.78**	4.59	4.95	4.49	4.28	4.56	4.10	5.45	4.01	4.79	5.80
Sharpe	**1.89**	1.28	1.38	1.25	1.19	1.27	1.14	1.52	1.12	1.33	1.61
Skew	**1.08**	0.49	0.24	0.34	0.43	0.55	0.25	1.71	0.94	0.12	0.42
Panel B: 2004:01-2019:12											
H-L	**0.72*****	0.27	0.34	0.42	0.25	0.24	0.42	0.38	0.53*	0.53**	0.33
t-stat	**3.06**	0.87	1.10	1.40	0.82	0.80	1.53	1.34	1.77	2.19	1.34
Sharpe	**0.77**	0.22	0.28	0.35	0.21	0.20	0.38	0.34	0.44	0.55	0.34
Skew	**-0.02**	0.02	-0.04	0.04	0.28	0.01	-0.41	0.78	0.64	0.42	0.48

Table 3

Spread portfolios controlling for other models

This table reports the summary statistics for the decile spread portfolios of each model controlling for one of the other models. Panel A provides the results for the GP controlling for one of the other models, and Panel B provides the results for other models controlling for the GP. The sample period is from 1991:01 to 2019:12.

	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
Panel A: GP, controlling for other models										
Low	0.66	0.65	0.64	0.60	0.66	0.55	0.61	0.65	0.58	0.55
2	0.96	1.01	0.99	0.95	0.95	0.84	0.83	0.88	0.92	0.86
3	1.10	1.06	1.14	1.07	1.13	0.98	1.09	0.96	0.87	0.88
4	1.11	1.21	0.98	1.41	1.07	0.99	0.99	0.98	1.01	1.08
5	1.15	1.19	1.26	1.09	1.17	1.22	1.16	1.17	1.16	1.10
6	1.04	1.13	1.11	1.13	0.93	1.18	1.11	1.23	1.10	1.07
7	1.55	1.27	1.42	1.32	1.57	1.13	1.29	1.35	1.03	1.23
8	1.20	1.35	1.35	1.12	1.25	1.29	1.37	1.25	1.33	1.29
9	1.43	1.74	1.54	1.53	1.38	1.38	1.35	1.31	1.38	1.28
High	1.28	1.26	1.30	1.30	1.29	1.29	1.23	1.25	1.39	1.40
H-L	0.62***	0.62***	0.65***	0.70***	0.62***	0.74***	0.62***	0.60***	0.81***	0.85***
t-stat	4.27	4.13	4.43	4.47	4.30	4.90	4.09	4.20	5.58	5.70
Panel B: Other models, controlling for GP										
Low	0.97	1.04	1.05	1.03	0.96	0.91	0.82	0.61	0.84	0.94
2	1.08	1.17	1.29	1.34	1.27	1.18	0.99	1.13	1.18	1.26
3	1.26	1.23	1.37	1.28	1.25	0.83	1.01	1.22	1.13	1.39
4	1.25	1.33	1.30	1.30	1.16	1.32	0.94	1.18	1.23	1.29
5	1.14	1.33	1.28	1.20	1.23	1.13	1.00	1.21	1.38	1.28
6	1.37	1.27	1.25	1.05	1.37	1.13	0.97	1.32	1.12	1.20
7	1.05	1.06	1.04	1.02	1.08	1.19	0.98	1.13	1.04	1.17
8	1.00	1.02	1.05	1.07	1.02	1.10	0.94	0.98	1.08	1.14
9	0.87	0.87	0.85	0.94	0.87	1.03	0.95	1.00	0.96	0.97
High	0.96	0.96	0.96	0.95	0.96	0.90	0.97	0.88	0.92	0.90
H-L	0.00	-0.08	-0.09	-0.07	0.00	-0.01	0.15	0.27	0.09	-0.04
t-stat	-0.01	-0.32	-0.38	-0.31	-0.02	-0.03	0.68	1.15	0.28	-0.11

Table 4

Performance under information uncertainty

This table reports the β_L and β_H and their t-stats for the regression:

$$\Delta R_t = \beta_L Low_{t-1}^{Vol} + \beta_H High_{t-1}^{Vol} + \beta MKT_t + \epsilon_t,$$

where ΔR_t is the spread portfolio return of the GP minus the spread of one of the other models, and Low_{t-1}^{Vol} and $High_{t-1}^{Vol}$ are dummy variables indicating high- and low-IVOL periods, as classified based on the median level of the IVOL index, which is defined as the cross-sectional mean of the IVOL of individual stocks. The last row "Average" reports the statistics average over the 10 models. The sample period is from 1991:01 to 2019:12.

	β_L	t-stat	β_H	t-stat
Ridge	0.26	0.94	1.05***	2.75
Lasso	0.15	0.50	1.02**	2.51
Enet	0.10	0.36	1.13***	2.67
PCR	0.30	1.08	1.28***	2.80
PLS	0.27	0.96	1.08***	2.80
NN1	0.23	0.98	1.29***	3.01
NN2	0.41	1.60	0.57	1.57
NN3	0.36	1.55	0.79*	1.70
NN4	0.35	1.41	0.83**	2.31
NN5	0.50**	2.18	0.73*	1.91
Average	0.31	1.20	1.02	2.43

Table 5

Comparison with existing factors

This table provides the summary statistics of the GP factor (GPF) and the well known factors. Panel A reports the average monthly return (*Mean*) (%), the Newey-west (1987) robust t-statistics, the standard deviation (*Std.dev.*) (%) the annual Sharpe ratio (*Sharpe*), the skewness (*Skew*), and kurtosis (*Kurt*). Panel B reports the correlation matrix. The sample period is from 1991:01 to 2016:12.

	GPF	Mkt	SMB	HML	RMW	CMA	IA	ROE	MGMT	PERF	PEAD	FIN
Panel A: Summary statistics												
Mean	1.20***	0.69**	0.21	0.30	0.34*	0.26*	0.28**	0.44***	0.53***	0.64**	0.51***	0.56**
t-stat	(6.65)	(2.56)	(1.30)	(1.38)	(1.78)	(1.86)	(2.25)	(2.67)	(2.80)	(2.19)	(4.04)	(2.07)
Std. dev.	2.37	4.24	3.25	3.04	2.71	2.08	1.99	2.80	2.96	4.47	2.06	4.44
Sharpe	1.75	0.56	0.22	0.34	0.44	0.44	0.49	0.54	0.62	0.50	0.86	0.44
Skew	0.85	-0.67	0.74	0.16	-0.41	0.60	0.32	-0.72	0.46	0.02	0.30	-0.03
Kurt	5.93	4.34	11.17	5.42	12.95	5.43	5.09	7.48	5.53	6.28	7.32	8.36
Panel B: Correlation matrix												
GPF	1.00	0.12	0.11	-0.15	-0.11	-0.10	-0.14	-0.06	-0.06	0.08	0.07	-0.13
Mkt	0.12	1.00	0.22	-0.16	-0.46	-0.36	-0.32	-0.45	-0.45	-0.45	-0.12	-0.54
SMB	0.11	0.22	1.00	-0.28	-0.55	-0.14	-0.25	-0.45	-0.42	-0.11	0.11	-0.57
HML	-0.15	-0.16	-0.28	1.00	0.38	0.66	0.68	0.14	0.67	-0.23	-0.25	0.64
RMW	-0.11	-0.46	-0.55	0.38	1.00	0.25	0.33	0.73	0.50	0.42	-0.08	0.76
CMA	-0.10	-0.36	-0.14	0.66	0.25	1.00	0.91	0.14	0.74	0.05	-0.10	0.59
IA	-0.14	-0.32	-0.25	0.68	0.33	0.91	1.00	0.20	0.76	0.00	-0.17	0.67
ROE	-0.06	-0.45	-0.45	0.14	0.73	0.14	0.20	1.00	0.34	0.66	0.21	0.55
MGMT	-0.06	-0.45	-0.42	0.67	0.50	0.74	0.76	0.34	1.00	0.13	-0.08	0.81
PERF	0.08	-0.45	-0.11	-0.23	0.42	0.05	0.00	0.66	0.13	1.00	0.43	0.24
PEAD	0.07	-0.12	0.11	-0.25	-0.08	-0.10	-0.17	0.21	-0.08	0.43	1.00	-0.11
FIN	-0.13	-0.54	-0.57	0.64	0.76	0.59	0.67	0.55	0.81	0.24	-0.11	1.00

Table 6

Spanning test and Sharpe ratio test

Panel A reports six spanning tests of whether the GP factor can be spanned by various factor models: W, the Wald test under conditional homoskedasticity; W_e, the Wald test under the IID elliptical; W_a the Wald test under the conditional heteroskedasticity; J_1, the Bekaert-Urias test with the Errors-inVariables (EIV) adjustment; J_2 is the Bekaert-Urias test without the EIV adjustment, and $J3$, the DeSantis test. The p-values are in brackets. Panel B reports the results of the Sharpe ratio test. "*Original*" reports the squared monthly Sharpe ratios (Sh^2) of a model. "*With GPF*" reports the squared monthly Sharpe ratios for a model plus the GP factor. "$\Delta(Sh^2)$" reports the Sh^2 difference of the two models. The bootstrap p-value, for the null hypothesis of no difference, is reported in brackets, following Ledoit and Wolf (2008) with a repetition number of 4999. The sample period is from 1991:01 to 2016:12.

Panel A: Spanning test

	W	W_e	W_a	J_1	J_2	J_3
CAPM	895.85***	451.07***	645.15***	69.84***	69.08***	452.33***
	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]
FF-3	217.31***	114.03***	155.78***	63.64***	71.72***	137.74***
	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]
FF-5	95.58***	59.19***	73.41 ***	64.19***	71.74***	90.15***
	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]
HXZ-4	102.35***	61.80***	80.62***	62.86***	71.70***	84.06***
	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]
SY-4	68.65***	46.09***	63.01***	42.58***	47.11***	53.81***
	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]
DHS-3	98.73***	58.60***	73.41***	55.08***	60.63***	85.44***
	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]	[0.00]

Panel B: Sh^2 in the Sharpe ratio test

	Original	With GPF	$\Delta(Sh^2)$	p-value
CAPM	0.026	0.265	0.239***	[0.00]
FF-3	0.046	0.304	0.258***	[0.00]
FF-5	0.137	0.390	0.253***	[0.00]
HXZ-4	0.147	0.403	0.256***	[0.00]
SY-4	0.210	0.408	0.198***	[0.00]
DHS-3	0.200	0.438	0.238***	[0.00]

Table 7

Risk-adjusted returns

The table reports the risk-adjusted returns of the spread portfolios generated by the GP and other methods. Newey-west (1987) robust t-statistics are reported in parentheses. The sample period is from 1991:01 to 2016:12.

	GP	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
CAPM	1.69***	0.95***	1.00***	0.97***	0.86**	0.93***	1.02***	1.25***	0.98***	1.07***	1.07***
	(5.78)	(2.94)	(2.91)	(2.96)	(2.36)	(2.85)	(4.07)	(3.90)	(3.45)	(4.34)	(3.60)
FF-3	1.77***	0.84***	0.90***	0.86***	0.69***	0.82***	0.94***	1.12***	0.93***	1.03***	1.01***
	(5.39)	(3.50)	(3.37)	(3.54)	(2.77)	(3.36)	(4.62)	(4.49)	(4.23)	(4.55)	(3.86)
FF-5	1.86***	0.88***	0.95***	0.89***	0.67***	0.86***	0.81***	1.12***	1.14***	0.95***	0.97***
	(4.76)	(3.53)	(3.40)	(3.56)	(2.65)	(3.37)	(3.70)	(3.83)	(4.91)	(4.15)	(3.86)
HXZ-4	1.80***	0.75***	0.77***	0.73***	0.55**	0.72***	0.51**	0.96***	1.02***	0.81***	0.86***
	(4.97)	(3.01)	(2.73)	(2.89)	(2.13)	(2.84)	(2.06)	(3.84)	(4.23)	(3.62)	(3.08)
SY-4	1.56***	0.70***	0.79**	0.72***	0.55**	0.68**	0.42*	0.87***	0.98***	0.89***	0.70***
	(5.33)	(2.68)	(2.57)	(2.69)	(2.02)	(2.55)	(1.67)	(3.56)	(4.36)	(3.73)	(2.77)
DHS-3	1.67***	1.34***	1.33***	1.33***	1.12***	1.32***	1.00***	1.48***	1.50***	1.23***	1.13***
	(6.26)	(4.59)	(4.27)	(4.64)	(3.63)	(4.48)	(3.39)	(4.93)	(5.94)	(4.57)	(3.62)
CAPM+GPF	-0.01	-0.17	-0.05	-0.06	-0.06	-0.19	0.30	0.15	-0.07	0.04	-0.05
	(-0.06)	(-0.64)	(-0.16)	(-0.20)	(-0.20)	(-0.70)	(1.43)	(0.59)	(-0.33)	(0.20)	(-0.24)

Table 8

Alternative characteristic set

This table reports the performance of the decile spread portfolios based on the alternative characteristic set. For each spread portfolio, we report the average monthly return in percentage points, the Newey-west (1987) robust t-statistic, the annualized Sharpe ratio ($Sharpe$), the skewness ($Skew$), and the maximum drawdown (MDD) in percentage. The sample period is from 2001:01 to 2019:12.

	GP	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
Low	0.12	0.35	0.17	0.23	0.21	0.15	0.11	0.27	0.24	0.24	0.22
2	0.61	0.29	0.37	0.34	0.28	0.30	0.60	0.38	0.51	0.39	0.72
3	0.57	0.78	0.60	0.70	0.76	0.71	0.61	0.83	0.71	0.79	0.64
4	0.72	0.65	0.82	0.77	0.73	0.79	0.86	0.75	0.86	0.78	0.65
5	0.88	0.77	0.88	0.77	0.78	0.73	0.83	0.77	0.77	0.81	0.73
6	1.00	0.95	0.78	0.98	0.78	0.59	0.80	0.92	0.68	0.76	0.75
7	0.80	0.76	0.83	0.78	0.88	1.03	0.71	0.62	0.95	0.67	0.81
8	0.95	0.81	0.65	0.69	0.71	0.80	0.94	0.76	0.91	0.71	1.05
9	1.00	0.83	0.90	0.86	0.87	0.85	0.77	0.83	0.65	0.94	0.71
High	1.11	0.61	0.67	0.62	0.61	0.68	0.65	0.87	0.85	0.80	0.81
H-L	0.99***	0.26	0.51	0.39	0.40	0.53	0.54	0.61*	0.61*	0.56*	0.58
t-stat	3.29	0.60	1.14	0.89	0.94	1.26	1.45	1.76	1.91	1.66	1.57
Sharpe	0.74	0.13	0.26	0.20	0.21	0.28	0.32	0.40	0.43	0.37	0.35
Skew	0.91	0.40	0.43	0.44	0.48	0.21	0.71	0.55	0.34	0.21	0.55

Table 9

International evidence

The table reports the performance of the decile spread portfolios in other G7 markets. For each spread portfolio, we report the average monthly return in percentage points, the Newey-west (1987) robust t-statistic, the annualized Sharpe ratio (*Sharpe*). Panel A to F report the statistics for each of the six markets, whereas Panel G reports the average over the six markets. The sample period is from 1991:01 to 2019:12.

	GP	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
Panel A: UK											
Mean	1.69	1.29	1.34	1.43	1.31	1.29	1.33	1.66	1.30	1.01	1.13
t-stat	5.77	3.74	3.89	4.34	3.69	3.75	4.18	5.23	3.86	2.87	3.63
Sharpe	1.09	0.71	0.73	0.82	0.70	0.71	0.79	0.99	0.73	0.54	0.69
Panel B: Canada											
Mean	2.05	1.85	1.79	1.80	1.23	1.79	1.63	1.55	0.90	0.89	0.85
t-stat	4.62	3.43	3.35	3.34	2.39	3.32	3.11	3.32	1.83	1.72	1.65
Sharpe	0.86	0.64	0.63	0.62	0.45	0.62	0.58	0.62	0.34	0.32	0.31
Panel C: Germany											
Mean	2.11	1.36	1.60	1.41	1.40	1.39	0.82	0.75	0.65	0.80	1.40
t-stat	6.52	3.54	4.01	3.53	3.62	3.64	2.49	2.03	1.71	2.01	3.95
Sharpe	1.22	0.66	0.75	0.66	0.68	0.68	0.46	0.38	0.32	0.38	0.74
Panel D: Japan											
Mean	1.55	1.45	1.60	1.50	1.64	1.43	0.98	1.23	1.14	1.22	1.52
t-stat	6.15	5.11	5.78	5.31	5.15	5.07	4.76	5.47	4.95	5.49	5.41
Sharpe	1.30	1.02	1.13	1.06	1.10	1.01	0.83	1.05	0.90	1.07	1.11
Panel E: Italy											
Mean	1.23	1.35	1.09	1.15	1.11	1.24	1.08	0.89	1.34	1.31	1.11
t-stat	3.84	3.70	3.09	3.22	3.09	3.42	3.08	2.55	3.47	3.65	3.01
Sharpe	0.71	0.69	0.57	0.60	0.58	0.64	0.57	0.47	0.64	0.68	0.56
Panel F: France											
Mean	2.08	1.66	1.68	1.66	1.75	1.72	1.00	1.37	1.60	1.44	1.21
t-stat	6.77	4.77	5.10	4.92	4.96	4.94	2.91	4.13	4.61	4.25	3.53
Sharpe	1.26	0.89	0.95	0.92	0.92	0.92	0.54	0.77	0.86	0.79	0.66
Panel G: Average statistics over the markets											
Mean	1.78	1.49	1.52	1.49	1.40	1.48	1.14	1.24	1.16	1.11	1.20
t-stat	5.61	4.05	4.20	4.11	3.82	4.02	3.42	3.79	3.40	3.33	3.53
Sharpe	1.07	0.77	0.79	0.78	0.74	0.76	0.63	0.71	0.63	0.63	0.68

Table 10

Performance under alternative parameters

The table reports the annualized Sharpe ratio and average return of the spread portfolios generated by GP under alternative hyperparameters $< Pop, Gen >$ and M. Panel A, B, and C reports the results for $M = 5$, 3, and 10, respectively. The training sample is from 1945:01 to 1980:12. The validation sample is from 1981:01 to 1990:12. The OOS sample is from 1991:01 to 2019:12.

	Sharpe ratio									Mean return								
	Train			Validation			OOS			Train			Validation			OOS		
Gen\Pop	100	200	400	100	200	400	100	200	400	100	200	400	100	200	400	100	200	400
Panel A: Average of Top 5 Models																		
10	2.04	2.30	2.28	1.69	1.68	1.59	1.07	0.93	1.06	1.77	2.15	2.16	1.46	1.67	1.63	1.38	1.37	1.59
20	2.38	2.39	2.44	1.85	1.49	1.68	1.22	0.92	1.14	1.91	2.19	2.24	1.58	1.46	1.62	1.45	1.33	1.66
40	2.85	2.96	2.84	1.86	2.66	2.07	1.11	1.32	1.01	2.29	2.22	2.38	1.68	2.28	1.94	1.56	1.71	1.41
Panel B: Average of Top 3 Models																		
10	2.07	2.34	2.30	1.81	1.71	1.62	1.15	0.92	1.00	1.72	2.20	2.15	1.49	1.69	1.68	1.37	1.36	1.54
20	2.39	2.40	2.45	1.87	1.48	1.64	1.21	0.90	1.13	1.93	2.19	2.25	1.61	1.44	1.59	1.43	1.29	1.66
40	2.87	2.96	2.85	1.76	2.63	2.05	1.14	1.27	1.02	2.31	2.22	2.37	1.63	2.26	1.92	1.62	1.69	1.41
Panel C: Average of Top 10 Models																		
10	2.00	2.26	2.24	1.79	1.54	1.55	1.03	0.95	1.06	1.71	2.13	2.13	1.48	1.56	1.59	1.34	1.39	1.54
20	2.36	2.37	2.43	1.85	1.49	1.67	1.21	0.93	1.13	1.90	2.17	2.22	1.58	1.49	1.63	1.44	1.34	1.63
40	2.82	2.96	2.82	1.85	2.68	2.05	1.11	1.27	0.99	2.30	2.21	2.38	1.69	2.28	1.94	1.58	1.68	1.39

Table 11

Comparison with different objective functions

This table reports the summary statistics for the decile portfolios generated by GP under two objectives, i.e., to maximize the resulting spread portfolio's Sharpe ratio (GP_{SR}) and to minimize the conventional mean squared error (GP_{MSE}). We also report the results for the two methods controlling for each other. GP^{ω}_{SR} reports the results for GP_{SR} controlling for GP_{MSE}. Each month, the expected return under GP_{SR} is regressed in a cross-section regression on that under GP_{MSE}. Stocks are then sorted by the resulting residual into ten decile portfolios. GP^{ω}_{MSE} reports the results for GP_{MSE} controlling for GP_{SR}. The sample period is from 1991:01 to 2019:12.

	Original		Controlling for each other	
	GP_{MSE}	GP_{SR}	GP^{ω}_{MSE}	GP^{ω}_{SR}
Low	0.07	0.08	0.66	0.64
2	0.40	0.57	1.31	0.87
3	0.58	0.57	1.29	1.11
4	0.79	0.50	1.26	1.14
5	0.80	0.74	1.22	1.15
6	0.95	1.06	1.25	1.19
7	1.20	1.09	1.05	1.11
8	1.22	1.53	0.97	1.38
9	1.53	1.49	0.92	1.32
High	1.50	1.79	0.92	1.55
H-L	1.44***	1.71***	0.27	0.91***
t-stat	5.59	7.12	0.83	5.69
Std. dev.	4.79	4.47	5.72	2.84
Sharpe	1.04	1.32	0.16	1.11
Skew	0.37	1.17	-0.69	0.52

Table 12

Simulation: Linear vs nonlinear

This table reports the OOS performances of various models in the linear and nonlinear simulations. $\hat{R}_{i,t}$ is the fitted return from step 2 in the simulation procedure. The simulation procedure is discussed in section 5.2.

	Annual SR		Mean Rt	
	Linear	Nonlinear	Linear	Nonlinear
$\hat{R}_{i,t}$	1.33	3.36	1.26	3.49
GP	1.24	3.08	1.08	2.50
Ridge	1.30	2.03	1.24	1.97
Lasso	1.32	2.05	1.27	2.03
Enet	1.32	2.07	1.25	2.01
PCR	1.30	2.03	1.24	1.97
PLS	1.30	2.03	1.25	1.97

Table 13

Bootstrap with various sample size

This table reports the OOS performances of various models in bootstrap with different sample size. For "Half", in each month t, we resample stocks so that the stock number in the simulated data is only *half* of the actual stock number. For "Double", we do the same but double the stock number in the simulated data each month. Then, based on the simulated data, we estimate GP and NN models, and examine the spread portfolio in the OOS sample.

	Annualied SR		Mean Rt	
	Half	Double	Half	Double
GP	1.01	1.07	1.65	1.61
NN1	0.25	0.74	0.30	1.10
NN2	0.47	0.86	0.58	1.27
NN3	0.46	0.72	0.58	1.04
NN4	0.49	0.62	0.56	0.86
NN5	0.20	0.90	0.20	1.32

Online Appendix

A. Terminologies in GP

This section introduces the basic flow chart and some terminologies in GP.

A.1 The basic flow chart for GP

The evolution within a single run of the genetic programming can be summarized as follows:

Step 1: *Initialization.* Create an initial random population and evaluate the fitness of each individual.

Step 2: *Selection.* Select parent individuals from the current population, with the selection probabilities biased in favor of the relative fit individuals.

Step 3: *Transformation.* Apply crossover and mutation operators to the selected parents to create offspring.

Step 4: *Evaluation.* Evaluate the fitness of the offspring.

Step 5: *Selection.* Selecting the survivor individuals for the next generation.

Step 6: *Iteration.* Repeat step 2-6 until the termination criterion is satisfied.

Figure A.1 illustrates the basic framework for genetic programming. Some related terminologies are briefly discussed in the next subsections.

A.2 Program structure and encoding

The solution *individuals* are computer programs represented as tree structures, which is build of two types of basic primitives, *terminals* and *functions*.

Generally, the *terminal* node provides the inputs to the GP program, including the input data used to train the model and some random constants supplied to the GP program. The *function set* is a predefined function set which may be application-specific and the range of the functions is very broad. For example, the arithmetic functions, such as PLUS, MINUS, NULTIPLY, DIVIDE, and boolean functions, such as AND, OR.

Figure A.2 shows a example of a tree structured individual. This program has a depth of 2 and it consists of the two function nodes, "-" and "*times*", and three terminal nodes, C_1, C_2, and C_3, which are the input data. The entire tree can be also interpreted as a function, which computes C_1 - $C_2 \times C_3$.

A.3 Genetic operators

An initialized population usually perform poorly in fitness. Evolution proceeds by transforming the initial population by the use of the genetic operators. In machine learning terms, these are the search operators. The two principal GP genetic operators are *crossover* and *mutation*.

The *crossover* operator combines the genetic material of two parents individuals by swapping a part of one parent with a part of the other. Tree-based crossover is described graphically in Figure A.3. and proceeds as follows. First, choose two individuals as parents based on the selection policy. Second, select a random subtree in each parent. Third, swap the selected subtrees between the two parents. The resulting individuals are the children.

Mutation operator operates on only one individual. When an individual has been selected for mutation, the mutation operator select a point randomly and replaces the existing subtree at that point with a new randomly generated subtree.

A.4 Fitness and selection

Intuitively, the fitness function is a mapping between the genetic individuals and a metric evaluating its performance in solving the original problem. Fitness function is one of the most significant ingredient of GP, as it defines the environment for the evolution in the sense that it gives feedback to the learning algorithm regarding which individual should have a higher probability of being allowed to create offspring and which individuals should have a higher probability of surviving in the new generation.

Fitness functions are very problem-specific. In an optimization problem, the fitness function simply computes the value of the objective function. For example, assume the individual in Figure A.2 is generated for a regression problem to minimize the mean squared error (MSE). Then, we can use the sample data to calculate the the mean squared error as the fitness for the program shown in Figure A.2.

B. Definition for the alternative characteristics

Here, we provide the detailed definition for the 15 characteristics of Lewellen (2015).

$LogSize_{-1}$ Log market value of equity at the end of the prior month;

$LogB/M_{-1}$: Log book value of equity minus log market value of equity at the end of the prior month;

$Return_{-2,-12}$: Stock return from month -12 to month -2;

$LogIssues_{-1,-36}$ Log growth in split-adjusted shares outstanding from month -36 to month -1;

$Accruals_{Yr-1}$ Change in non-cash net working capital minus depreciation in the prior fiscal year,The Cross-section of Expected Stock Returns;

ROA_{Yr-1} Income before extraordinary items divided by average total assets in the prior fiscal year;

$LogAG_{Yr-1}$ Log growth in total assets in the prior fiscal year,

$DY_{-1,-12}$: Dividends per share over the prior 12 months divided by price at the end of the prior month,

$LogReturn_{-13,-36}$: Log stock return from month -36 to month -13,

$LogIssues_{-1,-12}$: Log growth in split-adjusted shares outstanding from month -12 to month -1,

$Beta_{-1,-36}$: Market beta estimated from weekly returns from month -36 to month -1,

$StdDev_{-1,-12}$: Monthly standard deviation, estimated from daily returns from month -12 to month -1,

$Turnover_{-1,-12}$: Average monthly turnover (shares traded/shares outstanding) from month -12 to month -1,

$Debt/Price_{Yr-1}$: Short-term plus long-term debt divided by market value at the end of the prior month,

$Sales/Price_{Yr-1}$: Sales in the prior fiscal year divided by market value at the end of the prior month

C. Further robustness under alternative parameters

Table A.1 compares the GP under alternative parameters with other competitive models. We carry out the cross-sectional regression to regress the expected returns of model-A on that of model-B, and then examine the spread portfolios formed on the resulting residuals. Panel A shows that GP under various parameters still generate strong spread returns after controlling for other models. On the contrary, in Panel B, none of linear models generate significant returns and only few of the NN models produce weak returns once controlling for GP. This results confirm that GP under alternative parameters persistently dominates other models by subsuming their predictability.

D. Detailed results about model volatility

Table A.2 shows that the volatility of GP's performances decreases with *Gen*. For example, in terms of the volatility of the Sharpe ratio of the top 5 models in Panel A, for *Pop*=200, when *Gen* increase from 10 to 40, the training sample volatility decrease from 1.37 to 0.39, and the OOS sample volatility also decreases from 3.06 to 1.05. This evidence indicates that as the models evolve in the direction guided by the objective of maximizing Sharpe ratio, they become less-diversified and tend to converge. Similar patterns are presented for other Panels. From the perspective of evolutionism, greater volatility indicates a greater species diversity. In other words, a lower volatility indicates that there is little room for the evolution.

Figure A.1: **A basic framework for genetic programming**

This figure illustrates the flow charts for a basic framework of genetic programming: *Step* 1: *Initialization*. Create an initial random population and evaluate the fitness of each individual. *Step* 2: *Selection*. Select parent individuals from the current population, with the selection probabilities biased in favor of the relative fit individuals. *Step* 3: *Transformation*. Apply crossover and mutation operators to the selected parents to create offspring. *Step* 4: *Evaluation*. Evaluate the fitness of the offspring. *Step* 5: *Selection*. Selecting the survivor individuals for the next generation. *Step* 6: *Iteration*. Repeat step 2-6 until the termination criterion is satisfied.

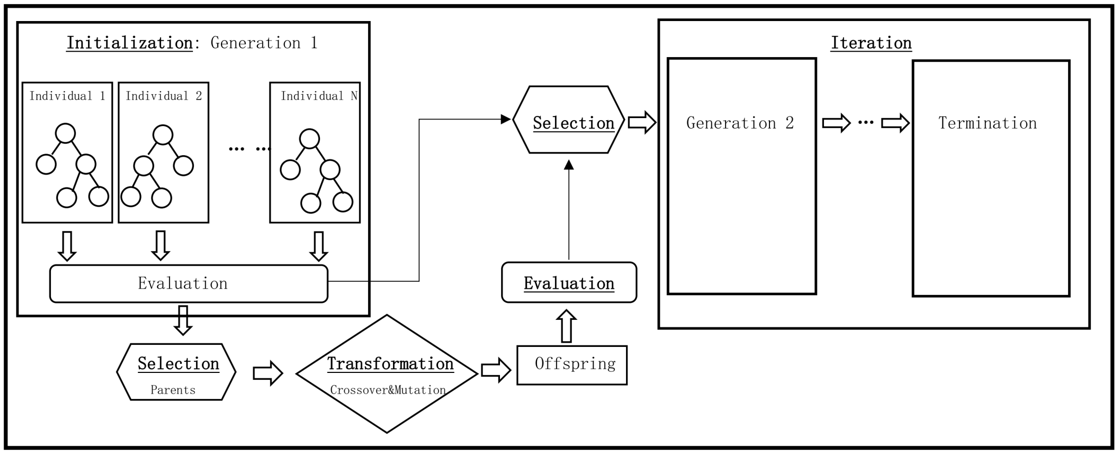


Figure A.2: **An example of a tree structures individual**

This figure illustrates an example of a tree structured individual. This program has a depth of 2 and it consists of the two function nodes, "-" and "*times*", and three terminal nodes, C_1, C_2, and C_3, which are the input data. This program computes C_1 - $C_2 \times C_3$.

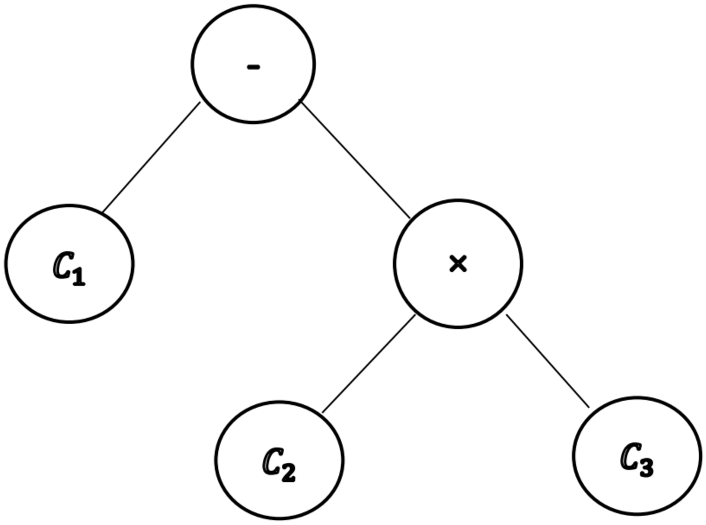


Figure A.3: **An example of crossover operator**

This figure illustrates how the crossover operator works.

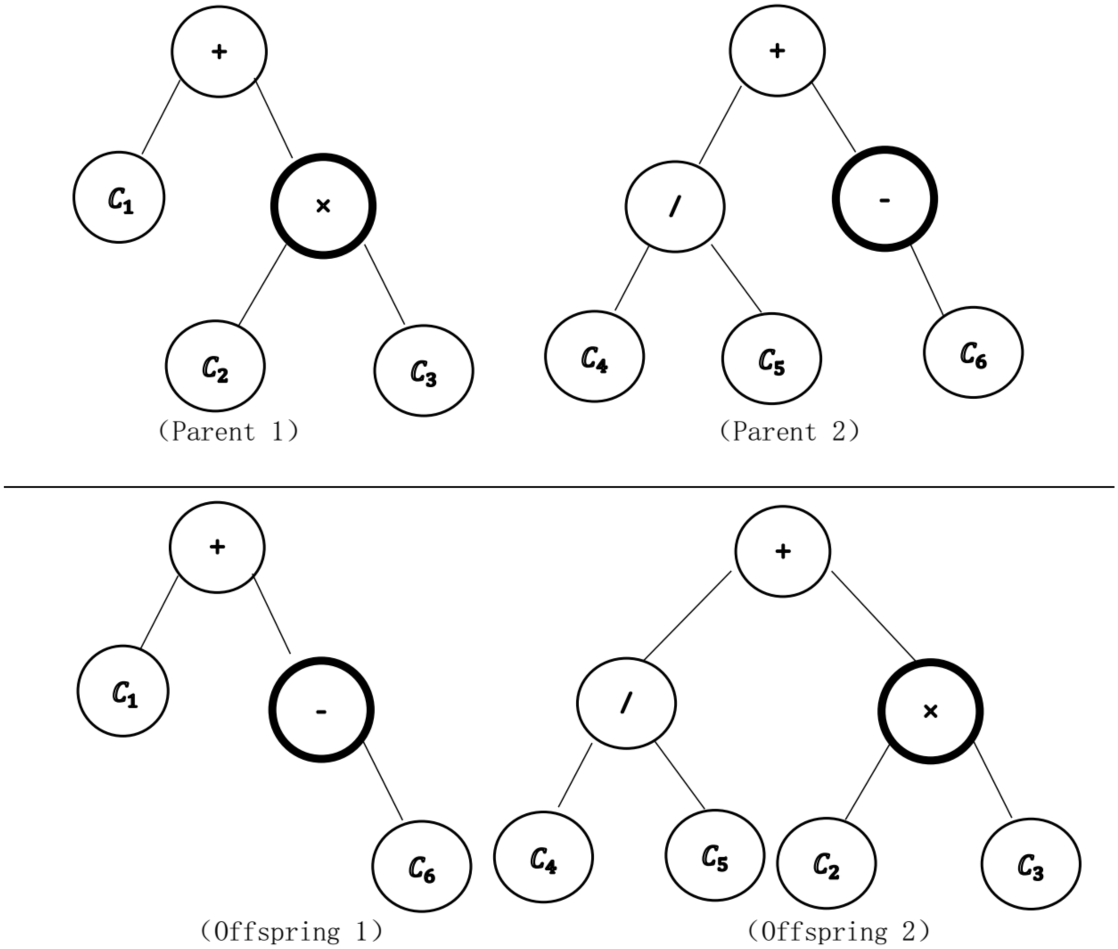


Table A.1

Spread portfolios controlling for other models: Under various GP parameters

This table reports the summary statistics for the decile spread portfolios of each model controlling for other models. Panel A reports the results for GP with various parameters of $< Pop, Gen >$ controlling for other benchmark models. Each month, the expected return on individual stocks generated by GP is regressed on that generated by another benchmark model in a cross-section regression. Stocks are then sorted by the associated residuals into ten decile portfolios. Similarly, Panel B reports the results for other models controlling for GP. The sample period is from 1991:01 to 2019:12.

$< Pop, Gen >$	Ridge	Lasso	Enet	PCR	PLS	NN1	NN2	NN3	NN4	NN5
Panel A: GP, controlling for other models										
$< 100, 10 >$	0.72***	0.74***	0.74***	0.76***	0.72***	0.83***	0.65***	0.69***	0.77***	0.79***
	(4.11)	(4.15)	(4.14)	(4.23)	(4.09)	(5.68)	(3.95)	(4.4)	(5)	(5.04)
$< 100, 20 >$	0.31**	0.32**	0.31**	0.32**	0.31**	0.46***	0.40***	0.30**	0.53***	0.6***
	(2.22)	(2.30)	(2.23)	(2.28)	(2.21)	(3.60)	(2.98)	(2.28)	(3.95)	(4.3)
$< 100, 40 >$	0.69***	0.70***	0.71***	0.78***	0.68***	0.76***	0.65***	0.76***	0.78***	0.74***
	(3.87)	(3.83)	(3.93)	(4.25)	(3.83)	(4.28)	(3.69)	(4.51)	(5.18)	(4.82)
$< 200, 10 >$	0.38**	0.40**	0.39**	0.39**	0.37**	0.49***	0.42***	0.34**	0.58***	0.6***
	(2.27)	(2.32)	(2.33)	(2.29)	(2.22)	(2.90)	(2.63)	(2.10)	(3.36)	(3.45)
$< 200, 20 >$	0.39***	0.42***	0.40***	0.39***	0.39***	0.54***	0.41***	0.42***	0.59***	0.68***
	(2.53)	(2.65)	(2.55)	(2.49)	(2.49)	(4.03)	(2.99)	(2.9)	(3.98)	(4.32)
$< 400, 10 >$	0.72***	0.69***	0.69***	0.69***	0.72***	0.76***	0.67***	0.69***	0.8***	0.79***
	(3.74)	(3.54)	(3.63)	(3.60)	(3.74)	(4.68)	(4.00)	(4.09)	(4.83)	(4.68)
$< 400, 20 >$	0.58***	0.58***	0.58***	0.6***	0.58***	0.67***	0.52***	0.64***	0.66***	0.66***
	(3.68)	(3.82)	(3.67)	(3.85)	(3.64)	(4.66)	(3.45)	(4.20)	(4.58)	(4.43)
$< 400, 40 >$	0.52***	0.64***	0.59***	0.56***	0.53***	0.65***	0.50***	0.44***	0.54***	0.56***
	(3.62)	(4.27)	(4.05)	(3.86)	(3.65)	(4.28)	(3.10)	(3.18)	(3.63)	(3.74)
Panel B: Other models controlling for GP										
$< 100, 10 >$	-0.04	-0.02	-0.10	-0.07	-0.04	-0.09	0.07	0.38**	0.03	-0.14
	(-0.17)	(-0.08)	(-0.42)	(-0.3)	(-0.17)	(-0.47)	(0.29)	(2.05)	(0.08)	(-0.38)
$< 100, 20 >$	0.02	-0.13	-0.13	-0.17	0.02	-0.01	-0.04	0.17	0.00	-0.19
	(0.09)	(-0.63)	(-0.59)	(-0.82)	(0.09)	(-0.12)	(-0.24)	(1.24)	(0.01)	(-0.64)
$< 100, 40 >$	0.13	0.00	0.15	0.12	0.08	0.02	0.21	0.55***	0.36	0.89**
	(0.55)	(0.02)	(0.59)	(0.45)	(0.36)	(0.1)	(0.87)	(2.89)	(0.99)	(1.98)
$< 200, 10 >$	0.11	0.28	0.27	-0.03	0.10	0.07	0.19	0.53***	0.39	0.12
	(0.41)	(0.97)	(0.99)	(-0.12)	(0.36)	(0.38)	(0.87)	(2.6)	(1.04)	(0.3)
$< 200, 20 >$	-0.04	-0.05	-0.05	-0.07	-0.01	-0.06	0.12	0.18	-0.39	-0.66*
	(-0.17)	(-0.21)	(-0.2)	(-0.3)	(-0.07)	(-0.47)	(0.77)	(1.23)	(-1.3)	(-1.75)
$< 400, 10 >$	-0.05	-0.04	-0.07	-0.24	-0.08	-0.03	0.11	0.29	0.12	0.06
	(-0.2)	(-0.13)	(-0.29)	(-0.98)	(-0.3)	(-0.13)	(0.5)	(1.55)	(0.34)	(0.16)
$< 400, 20 >$	-0.02	0.03	-0.12	-0.24	-0.02	-0.04	0.08	0.43**	0.35	0.24
	(-0.1)	(0.11)	(-0.46)	(-0.92)	(-0.08)	(-0.25)	(0.4)	(2.36)	(0.97)	(0.85)
$< 400, 40 >$	0.31	0.23	0.27	0.2	0.25	0.14	0.35	0.55***	0.48	0.46
	(1.26)	(0.83)	(1.03)	(0.72)	(1.04)	(0.69)	(1.62)	(3.25)	(1.37)	(1.51)

Table A.2

Volatility of GP under various parameters

This table reports the volatility of the Sharpe ratio of the spread portfolios generated by GP under various parameters. For a given $< Pop, Gen >$, GP generates Pop models (individuals) in the end. We calculate the volatility of the Sharpe ratios in the training (validation, and OOS) sample for the top M (M=5,10,50) models with the highest Sharpe ratios in the training sample. We estimate GP for 5 times, and report the average of the volatility over the five estimations. Panel A, B, and C reports the results for M= 5,10 and 50, respectively. The training sample is from 1945:01 to 1980:12. The validation sample is from 1981:01 to 1990:12. The OOS sample is from 1991:01 to 2019:12.

	Train	Val	OOS	Train	Val	OOS	Train	Val	OOS
Gen\Pop		100			200			400	
Panel A: Volatility of the SR of the top 5 models									
10	1.85	3.97	3.11	1.37	3.67	3.06	1.05	6.90	2.56
20	0.54	2.71	1.74	0.47	2.64	1.20	0.56	4.18	1.53
40	0.31	1.19	0.37	0.39	3.14	1.05	0.25	2.85	1.03
Panel B: Volatility of the SR of the top 10 models									
10	2.09	7.23	3.42	1.67	6.14	4.21	1.07	5.85	2.74
20	0.76	2.68	1.64	0.47	2.82	1.28	0.67	3.99	1.91
40	0.44	1.63	0.54	0.40	2.71	1.06	0.25	2.21	1.06
Panel C: Volatility of the SR of the top 50 models									
10	5.53	9.01	4.96	2.84	6.89	3.94	1.71	5.77	4.17
20	2.08	3.88	3.00	1.99	3.96	2.72	0.66	4.35	1.90
40	0.90	2.05	0.83	0.43	2.17	1.38	0.50	2.21	1.68