Deep Learning from Implied Volatility Surfaces

Bryan Kelly, Boris Kuznetsov, Semyon Malamud, and Teng Andrea Xu*

This version: September 15, 2023

Abstract

We develop a novel methodology for extracting information from option implied volatility (IV) surfaces for the cross-section of stock returns, using image recognition techniques from machine learning (ML). The predictive information we identify is essentially uncorrelated with most of the existing option-implied characteristics, delivers a higher Sharpe ratio, and has a significant alpha relative to a battery of standard and option-implied factors. We show the *virtue of ensemble complexity*: Best results are achieved with a large ensemble of ML models, with the out-of-sample performance increasing in the ensemble size, saturating when the number of model parameters significantly exceeds the number of observations. We introduce *principal linear features*, an analog of principal components for ML and use them to show *IV feature complexity:* A low-rank rotation of the IV surface cannot explain the model performance. Our results are robust to short-sale constraints and transaction costs.

*Bryan Kelly is at Yale School of Management, AQR Capital Management, and NBER; www.bryankellyacademic.org. Boris Kuznetsov is at the Swiss Finance Institute, EPFL. Semyon Malamud is at the Swiss Finance Institute, EPFL, and CEPR and is a consultant to AQR. Teng Andrea Xu is at EPFL. We are grateful for the helpful comments from Fabio Trojani. Semyon Malamud gratefully acknowledges the financial support of the Swiss Finance Institute and the Swiss National Science Foundation, Grant 100018_192692. AQR Capital Management is a global investment management firm that may or may not apply similar investment techniques or methods of analysis as described herein. The views expressed here are those of the authors and not necessarily those of AQR. This work was supported by a grant from the Swiss National Supercomputing Centre (CSCS) under project ID sm81.

1 Introduction

The Option Implied Volatility (IV) surface contains information about state-contingent risk premia and the probability distribution of returns over multiple horizons. Stock and option traders systematically use the shape of this surface, including its slope, skewness, and other geometric features, to infer market expectations and risk attitudes and make trading decisions. The option pricing theory provides a basis for such analysis: local properties of this surface indeed contain information about the underlying market dynamics. For example, the surface variation along the moneyness dimension can be used to uncover the Arrow-Debreu state prices (Breeden and Litzenberger, 1978a), while the surface slope along the maturity dimension contains information about volatility by the Dupire formula (Davis, 2011). In the language of machine learning, universal *local* features (non-linear transformations that depend on neighboring strikes and maturities) of the IV surface can be used to extract useful information about the stochastic structure of returns.

Despite the underlying theory's elegance, empirically computing the theory-driven features is associated with often unsurmountable econometric difficulties: theoretical formulas rely heavily on the continuity of strike and maturity dimensions, while, in reality, both strikes and maturities are discrete, living on a sparse grid. Furthermore, high bid-ask spreads due to often extreme illiquidity of options markets[1] introduce large amounts of noise into the estimation of equity implied volatilities.

In this paper, we leverage the progress in deep learning for computer vision and image recognition to construct powerful, non-linear, local features of IV surfaces. To this end, we exploit a particular neural network architecture called convolutional neural networks (CNNs). Thanks to their use of convolutional layers, CNNs are designed to capture spatial patterns and relationships between pixels in images. These layers apply filters to local regions of the input image, allowing the network to learn and identify patterns at various levels of abstraction (e.g., edges, shapes, and more complex features). The behavior of CNNs closely resembles the way in which humans interpret images. As a result, CNNs have the potential to detect patterns on the IV surface that are similar to those recognized by professional traders

[1] See (Glebkin et al., 2023).

during visual inspection. Furthermore, the locality of CNNs makes them perfect instruments for learning non-linear filters extracting the volatility and risk premia information from the noisy IV surface.[2]

We train several CNN architectures of increasing depth (complexity) on the standard OptionMetrics dataset of IV surfaces for several thousand stocks to predict stock returns over a one-month horizon. Since CNNs (like any neural networks) are trained by gradient descent, they are sensitive to (random) weight initialization: Depending on where the gradient descent starts, it may converge to a weight vector corresponding to a different local minimum.[3] A classic approach for dealing with this is to create an *ensemble* of neural networks corresponding to different weight initializations and then combine them. See, (Lakshminarayanan et al., 2017). We find that the benefits from this ensembling are huge because predictive models generated by different initializations produce portfolios with low pairwise correlations. As a result, for the most complex (4- and 5-layer) deep learning architectures, increasing the ensemble size from one to a hundred leads to an increase in the out-of-sample Sharpe ratio from 0.9 to 2.7 for the full stock universe. By contrast, for the lower complexity (1 hidden layer) CNN, the Sharpe ratio only increases from 0.80 to 1.6. As (Lakshminarayanan et al., 2017) explain, the sensitivity of predictions generated by NN to initialization captures the degree of uncertainty around these predictions. In low signal-to-noise ratio environments of financial markets, this uncertainty is very high, implying large gains from ensembling.

Since each model in the ensemble is represented by a (completely) different set of parameters, this finding implies a very large *virtue of complexity* (in the language of (Kelly et al., 2021) and (Didisheim et al., 2023)): bigger, more complex, non-linear models generate significant gains out-of-sample.[4] However, the nature of the complexity of the ensemble is different: While (Kelly et al., 2021) and (Didisheim et al., 2023) establish the virtue of complexity for one very big model with all parameters jointly trained, in this paper, we document *the virtue of ensemble complexity*. Namely, we train a large number of randomly

[2]Formally, locality means that two close points on the IV surface have similar informational content.

[3]The reason is that the dependence of CNN mean squared error on the CNN coefficients is highly non-convex. As a result, the problem is many local minima, and it is impossible to predict to which local minimum the gradient descent will converge.

[4]Table 9 shows that these models are indeed extremely complex, with the number of parameters significantly exceeding the number of observations in our panel dataset.

initialized CNNs with identical architecture and then average their predictions, and show that the out-of-sample Sharpe ratios are approximately monotone increasing in the ensemble size. A similar pattern is observed if we measure the out-of-sample performance of the model in terms of its alpha with respect to a large set of standard factors, including many standard stock characteristics as well as a large set of option-based characteristics from (Neuhierl et al., 2022).

The complexity of our CNN-based model ensemble seems to contradict the conventional principle of parsimony in economics, suggesting that predictive information (originating, e.g., from risk premia or behavioral anomalies) can be encoded into a small number of factors and stock characteristics. When applied to the IV surface, the principle of parsimony suggests that only a few key surface features might contain useful information about future stock returns. Most of these features used in the existing literature are constructed as simple, linear combinations of implied volatilities; for example, the IV level, the slope of the term structure, and the smile. To test whether such a low-dimensional structure is indeed present in our highly non-linear predictive model, we use the methodology of (Constantine et al., 2014) to identify key linear combinations of IVs with the most explanatory power. These combinations are natural analogs of principal components for our highly non-linear models, and we refer to them as *principal linear features*. Contrary to conventional wisdom, we find no evidence for a low-dimensional structure. While broadly consistent with the complexity principle of (Kelly et al., 2021) and (Didisheim et al., 2023), the nature of this phenomenon is different. Indeed, the results in (Kelly et al., 2021) and (Didisheim et al., 2023) mostly concern complexity due to non-linearities, whereas our last finding is about *feature complexity,* expressing the fact that a very large number of linear features is necessary to extract the predictive information contained in the IV surface.

As (Kelly et al., 2021; Didisheim et al., 2023) explain, this *virtue of complexity* means that, even for an already complex model, we can often find a new, nonlinear transformation of the IV surface that boosts the out-of-sample performance. This complexity is not a puzzle to be solved or evidence of data mining. Instead, it is the theoretically expected outcome of learning a non-linear, high-dimensional relationship with limited data. The only constraint we impose on the model is the principle of locality, formalized by choice of a convolutional

NN instead of a generic NN architecture utilized, say, in (Gu et al., 2020b) and (Chen et al., 2021).

Recently, several papers (see, e.g., (DeMiguel et al., 2020), (Detzel et al., 2023), and (Muravyev et al., 2022)) have argued that the performance of modern multi-factor models needs to be evaluated by accounting for transaction costs and short-sale constraints. We perform a detailed analysis of trading costs for our strategies and find that, perhaps surprisingly, the market-neutral portfolio constructed by sorting stocks based on the CNN-predicted return makes most of the money with the long leg, implying that our findings are not sensitive to short-selling costs and constraints. Transaction costs incurred by our strategy are significant: As with many machine learning models, it exhibits a high turnover (about 80% per month). Following (Jensen et al., Forthcoming) and (Didisheim et al., 2023), we study the model performance separately for several groups of stocks, created based on their market capitalization (mega, large, small, micro, and the non-micro group constructed as the set of all stocks excluding the micro-cap group). While the micro-group dominates the performance, the non-micro group also delivers a very strong Sharpe Ratio and a significant alpha. Both micro- and non-micro groups retain their alpha significance after accounting for realistic costs. It is also important to note that, in any case, the optionable stocks (i.e., stocks with options traded on them) are typically large and have a significant trading volume, implying that even the predictability we identify for micro stocks can be exploited with some meaningful arbitrage capital.[5]

The rest of this paper is organized as follows. Section 2 reviews the related literature. Section 3 describes data and methodology. Section 4 provides the necessary background on CNNs. Our main empirical results are reported in Section 5. Section 6 investigates feature importance, introduces *principal linear features*, and documents *the virtue of feature complexity*. Section 7 concludes.

[5]Recent results of (Jensen et al., 2022a) suggest that, by properly smoothing positions, it is possible to reduce turnover and preserve the bulk of the performance of machine learning models such as ours. We leave this important direction for future research.

2 Related Literature

Understanding why average returns differ across assets is a central question in finance. Over the last few decades, the search for stock characteristics that predict returns has led to the emergence of the (constantly growing) "factor zoo": A huge number of characteristics that contain information about the cross-section. See, for example, (Cochrane, 2011), (Harvey et al., 2016), (McLean and Pontiff, 2016), (Hou et al., 2020), (Feng et al., 2020), (Jensen et al., Forthcoming), and (Giglio et al., 2022) for a recent overview.

Many papers in this literature focus on predicting asset returns using complex, non-linear models; see (Moritz and Zimmermann, 2016), (Chinco et al., 2019), (Han et al., 2019), (Chen et al., 2019), (Bryzgalova et al., 2020), (Liu et al., 2020), (Gu et al., 2020b), (Kozak et al., 2020), (Freyberger et al., 2020), (Avramov et al., 2021), (Guijarro-Ordonez et al., 2021), (Leippold et al., 2022), (Kelly et al., 2021), and (Didisheim et al., 2023). Given the ever-growing complexity of these models (both in terms of the number of characteristics and the degree of non-linearity of the predictive relationships), several papers develop techniques to "shrink" the cross-section and find a sparse representation of the expected returns, either through a form of dimensionality reduction (e.g., by exploiting principal components, as in (Kelly et al., 2020), (Kozak et al., 2018), (Kozak et al., 2020), (Lettau and Pelger, 2020), and (Giglio and Xiu, 2021)), or by imposing sparsity directly in the space of characteristics (see, e.g., (Gu et al., 2020b), (Freyberger et al., 2020), and (Bryzgalova et al., 2023)). While the evidence is mixed, recent findings of (Didisheim et al., 2023) suggest that complexity is there to stay, and there might be no feasible way to find a sparse representation of expected returns. Namely, as (Didisheim et al., 2023) show, the factor zoo is simply a statistical phenomenon originating from the small data problem: We do not have enough data to find the right low-dimensional representation of expected returns, even if it exists; hence, our best bet is to build the most complex model without imposing a sparse prior.[6]

Most of the above-mentioned papers focus on stock characteristics that are either purely price-based (such as momentum; see, (Carhart et al., 1997)) or depend on company fundamentals such as the book-to-market ratio. By contrast, our paper focuses exclusively on the

[6]More generally, several recent papers have questioned the principle of sparsity in economic modeling. See, e.g., (Giannone et al., 2021).

predictive information contained in the implied volatility surface, motivated by the idea that derivative prices provide an interesting lens to uncover rich information about the underlying assets and associated risks. The early contributions by (Breeden and Litzenberger, 1978b), (Banz and Miller, 1978) show how Arrow-Debreu state prices can be recovered from the option prices. State prices contain information about risk premia and (subjective) physical probabilities as market participants anticipate. Under technical conditions, some information about these physical probabilities can be recovered; see, e.g., (Ross, 2015), (Borovička et al., 2016), and (Jensen et al., 2019). The idea of extracting useful forward-looking information from both individual equity and index options has been exploited in many papers. For example, (Bali and Hovakimian, 2009) show that the difference between implied (i.e., risk-neutral) and physical volatility, as well as the difference between the implied volatilities of near-the-money call and put options are both strong predictors of stock returns; (Cremers and Weinbaum, 2010) use deviations from put-call parity to predict stock returns; (Johnson and So, 2012) show that the ratio of the volume between options and stocks predicts stock returns at the one-week horizon; (Chordia et al., 2020) reach a similar conclusion when examining the (signed) order flow in equity index options; (An et al., 2014) find that stocks with large increases in call (put) implied volatilities over the previous month tend to have high (low) future returns; (Andersen et al., 2015), (Bollerslev and Todorov, 2014; Bollerslev et al., 2015), (Lin and Todorov, 2019), (Begin et al., 2020), and (Han et al., 2020) show how the jump (tail) risk extracted from options prices predicts future stock returns; (Baltussen et al., 2018) show that the volatility of implied volatility has predictive power for future stock returns, and (Dew-Becker and Giglio, 2020) show how to estimate cross-sectional uncertainty from option prices.

Many option-implied predictors build on a potential risk-return relationship between the risk-neutral variance, variance risk-premium (VRP), and stock returns; see, e.g., (Bollerslev et al., 2009), (Feunou et al., 2018), (Martin and Wagner, 2019), (Kilic and Shaliastovich, 2019), (Feunou et al., 2019), (Tang, 2019), (Kadan and Tang, 2020), (Pederzoli, 2020), and (Duarte et al., 2022).

Several other papers propose characteristics of the IV surface that are related to the shape of the implied volatility smirk or corresponding risk-neutral skewness portfolios; see,

e.g., (Xing et al., 2010), (Yan, 2011), (Conrad et al., 2013), (Stilger et al., 2017), (Jones et al., 2018) and (Bali and Murray, 2019), and (Schneider et al., 2020). While many of these papers find that IV-surface-based skewness measures contain predictive information, the nature of this information seems extremely sensitive to precise details of factor construction. For example, (Conrad et al., 2013) document a negative relation between ex-ante risk-neutral skewness and asset returns, while (Stilger et al., 2017) finds a positive relation.[7] These findings suggest that the nature of predictive information might be extremely complex, making it difficult to pin down the precise underlying economic mechanism.

The paragraphs above suggest that, even in the smaller world of option-implied characteristics, we are clearly facing the problem of an ever-growing factor zoo, with new predictive relationships constantly discovered by academic researchers. Two recent papers attempt to bring order into this option-implied factor zoo by combining the informational contents of the multiple factors identified in the existing literature. (Bali et al., 2022) use five of those factors (based on the findings of (Bali and Hovakimian, 2009), (Cremers and Weinbaum, 2010), (Xing et al., 2010), and (An et al., 2014)). They provide evidence that a linear tangency portfolio of these factors is not spanned by standard stock characteristics-based managed portfolios. However, in a more recent paper, (Neuhierl et al., 2022) provide evidence that the factors in (Bali et al., 2022) are spanned by fundamental stock characteristics when more of those characteristics are included in the model. (Neuhierl et al., 2022) then argue that machine learning methods can deal with the IV-surface-based factor zoo and extract useful information unspanned by standard factors. They consider 17 options-based characteristics, including five of (Bali et al., 2022). Then, they apply the adaptive group LASSO methodology of (Freyberger et al., 2020) to build non-linear predictive models based on these 17 characteristics. They provide strong evidence that only 4 out of 17 option characteristics contain information about future stock returns not spanned by a large set of more than 60 stock characteristics. These four economically important characteristics are all related to the shape of the IV smirk.

While (Neuhierl et al., 2022) do develop a complex, non-linear, machine learning model,

[7] See, also, (Jiang et al., 2020) for similar results based on physical measures of skewness; and (Kozhan et al., 2013), (Schneider and Trojani, 2019), and (Orlowski et al., 2020) for a studies of aggregate measures of skewness.

they utilize relatively small (only 17) ready, pre-built characteristics of the IV surface, motivated by sparsity considerations and the idea that a few linear features of the IV surface summarize its predictive information. In this paper, we follow a different approach. First, we completely abstract from the existing set of characteristics such as smirk, skew, slope, etc.; instead, we take an agnostic approach and let the machine learning model decide which features of the IV surface are useful for predicting stock returns. Second, we do not try to impose any form of sparsity on the model. Instead, we build a model with an exorbitant number of features and parameters following the principle of the virtue of complexity ((Kelly et al., 2021; Didisheim et al., 2023)) combined with the *virtue of ensemble complexity* introduced in this paper. As explained above, we find evidence for both virtues: There is a large amount of predictive information in the IV surface, and we need a large ensemble of highly complex models to leverage this information efficiently. Furthermore, our model exhibits a very high *feature complexity:* A small number of linear IV features cannot span our model's predictive content.

We complete this literature review by noting that implied volatility is closely related to the physical volatility (in fact, in the idealized Black-Scholes continuous time setting, the two should be identical) and, hence, any signal about the level of the IV surface is closely related to the physical idiosyncratic stock volatility. The latter contains predictive information about stock returns and belongs to the large family of "low-risk anomalies." See, (Ang et al., 2006), (Harvey and Siddique, 2000), (Boyer et al., 2010), (Frazzini and Pedersen, 2014), (Amaya et al., 2015) and (Schneider et al., 2020). In particular, (Schneider et al., 2020) argue that these low-risk anomalies reflect compensation for co-skewness risk. It is possible that the more complex predictive signals extracted from the IV surface by our model might also reflect a form of risk premium related to some jump, tail, or high-order moment risk. Understanding the connection of our non-linear signals with risk premia is an important direction for future research.

3 Data and methodology

3.1 Data

Options data are from OptionMetrics IvyDB. The historical IvyDB Volatility_Surface dataset contains interpolated option implied volatility surfaces for a large set of firms for each trading day from January 1996 to December 2021. For interpolation, a proprietary kernel smoothing algorithm is applied by OptionMetrics across both moneynesses (defined in terms of option δ) and expiry (defined in terms of days-to-maturity τ) grids, allowing us to abstract from option cycles and varying strikes. Thus, for each trading day, δ-moneyness grid goes from -1 to +1 in equidistant 0.05 steps (puts have negative δ, calls have positive δ), and expiry grid ranges from 10 to 730 days-to-maturity.

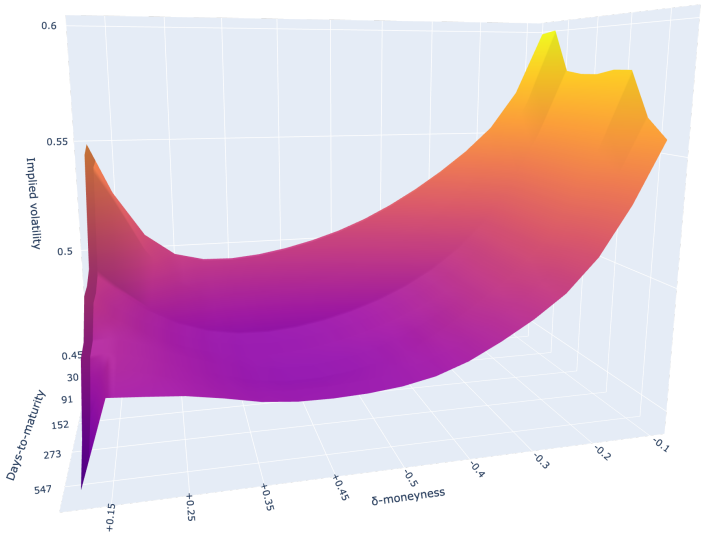


Figure 1: Transformed implied volatility surface of TSLA stock, 01/07/2016.

We hypothesize that in-the-money calls contain the least forward-looking information and only keep $\delta \in [-0.5, 0.5]$; then, we drop all 10-days-to-maturity values due to lack of data before 2010[8]; lastly, we re-stack volatility surfaces at-the-money (so that δ goes from 0.1 to 0.5, then goes from -0.5 to -0.1) – this way, the combination of put and call IV components results in a more seamless transition. Thus, we end up with implied volatility surface images of size 10×18. Figure 1 shows a typical example of an implied volatility surface from Option

[8]This is likely because so-called "weeklies" (options with weekly expiration cycles) were only introduced in 2005 and gained popularity after the Global Financial Crisis.

Metrics IvyDB that we work with. We denote $IV_{i,t} = (IV_{i,t}(\delta,\tau))_{\delta\in\Delta,\tau\in\mathcal{T}}, \in \mathbb{R}^{|\Delta|\times|\mathcal{T}|}$ the interpolated implied volatility surface of a stock $i \in N$ on the last day of the month $t \in T$. Here, $\delta \in \Delta$, where Δ is the grid of available δ-levels, and $\tau \in \mathcal{T}$, where $\mathcal{T}$ is the grid of available times to maturity.

We use daily stock data from CRSP to construct monthly total stock returns for all NYSE, AMEX, and NASDAQ firms. CRSP sample is aligned to our OptionsMetrics sample and lasts from January 1996 to December 2021. The number of stocks with available stock and options data is shown in Figure 2. The total number of stocks in our sample is close to 25 thousand, with the average number of stocks at any point in time exceeding three thousand.[9] We denote $R_{i,t+1}$ the total return on stock i from the last business day of the month t to the last business day of the month $t+1$.

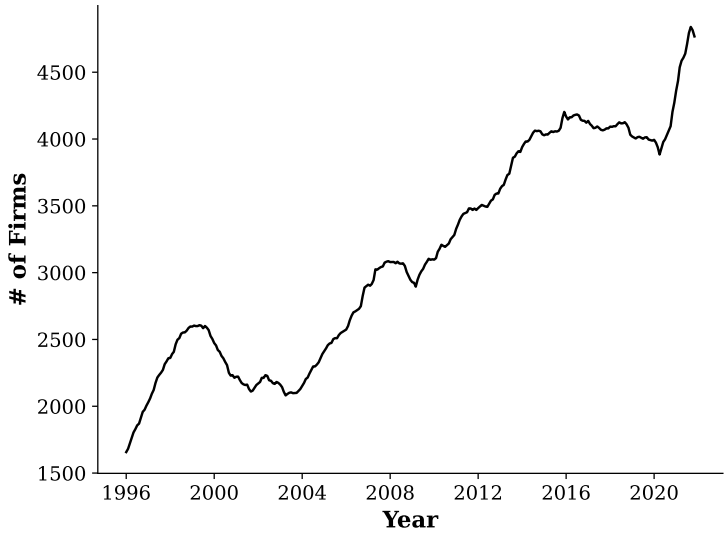


Figure 2: Evolution of our post-processed dataset size regarding the number of unique firms.

Monthly returns are constructed as follows. First, for each stock i and business day d, daily total gross returns are computed as $(1 + r_{i,d})$ or as $(1 + r_{i,d})(1 + r_{i,d}^{\text{delist}})$ if the stock i delists on day d. Here, $r_{i,d}$ is the total daily return from the CRSP Daily Stock file, and $r_{i,d=delist}$ is the delisting return from the Delisting Information CRSP file. Missing daily returns[10] are replaced with zeros. Monthly total gross returns for a month t, denoted by

[9] See the section A of the Appendix for a detailed description of CRSP and OM datasets, the dataset linking procedure, and some additional statistics.

[10] 1.34% of the total number of return observations.

$R_{i,t}$, are cumulative products of daily gross returns,

$$R_{i,t} \;=\; \prod_{d\in t}(1+r_{i,d})(1+r_{i,d}^{\text{delist}}\mathbf{1}_{d=delist}) \;-\; 1\,. \tag{1}$$

Everywhere in this paper, *we focus on predicting monthly returns* $R_{i,t+1}$ using only implied volatility surfaces *on the last business day of the preceding month* t, which we denote by $IV_{i,t}$.[11]

3.2 Ensembles of Randomly Initialized Neural Nets

Given a family of non-linear functions, $f(x;w)$, indexed by a (high-dimensional) vector of parameters w (e.g., neural weights for the case of neural networks), we can optimize w with the objective of predicting returns $R_{i,t+1}$ by plugging the month-end $IV_{i,t}$ into f and minimizing an error on the training data: Given a look-back horizon T, we can try selecting a parameter vector w that minimizes the in-sample prediction error based on the last T periods of the data:

$$w_{*,t} \;=\; \arg\min_{w}\ell(w),\;\; \ell(w) \;=\; \sum_{\theta=t-T}^{t}\sum_{i=1}^{N_\theta}(R_{i,\theta+1}-f(IV_{i,\theta};w))^2\,, \tag{2}$$

where N_θ is the number of stocks in our data set at time θ. The standard way of finding w_* is by using gradient descent: Given a learning rate η, one can randomly initialize w_0 and then implement the algorithm of gradient descent by iteratively computing

$$\hat{w}_j(w_0) \;=\; \hat{w}_{j-1} \;-\; \eta\,\nabla_w\ell(\hat{w}_{j-1}) \tag{3}$$

for $j\geq 1$, where $\nabla_w\ell$ is the gradient of the loss $\ell(w)$ with respect to w.

By definition, the path of $\hat{w}_j(w_0)$ during the gradient descent (as j increases from one to ∞) depends on the initialization w_0. However, standard results imply that, as the number

[11] One may wonder whether information about the daily dynamics of implied volatility surfaces in the preceding month t contains predictive information about $R_{i,t+1}$ and whether we are neglecting this information by keeping only month-end $IV_{i,t}$. We have tried to incorporate additional information about daily IV dynamics in the preceding month in our predictive models, and these experiments suggested that lagged IV information during the month does not add predictive power.

j of gradient steps increases while the learning rate η converges to zero, $\hat{w}_j(w_0)$ converges to a *local extremum* $\hat{w}_\infty(w_0) = \lim_{j\to\infty,\eta\to 0} \hat{w}_j(w_0)$, satisfying $\nabla_w \ell(\hat{w}_\infty(w_0)) = 0$. In fact, as recent research shows, these local extrema are typically global minima for neural nets. See, e.g., (Soudry and Carmon, 2016). In simple, linear regression problems, the loss function $\ell(w)$ is convex in w and, hence, has a unique extremum which is also a global minimum; hence, for convex problems, $\hat{w}_\infty$ is independent of w_0. By contrast, for realistic, non-linear machine-learning models (such that CNNs in our paper), $\ell(w)$ is not convex and has a tremendous number of (global or local) minima. See, e.g., (Lakshminarayanan et al., 2017; Li et al., 2018; Yao et al., 2020; Fort et al., 2019). As a result, $\hat{w}_\infty(w_0)$ depends on the initial weights, w_0, in a highly complex fashion. In particular, for K different initializations $w_0(k), k = 1, \cdots, K$, we will typically end up with K different weight vectors $\hat{w}_\infty(w_0(k))$. This leads to a *randomly initialized ensemble* of K models, $\{f(x; \hat{w}_\infty(w_0(k)))\}$, indexed by $k = 1, \cdots, K$. We then build the ensemble prediction by taking a simple average across k :

$$\widehat{R}_{i,t}^{\text{ens}} = \frac{1}{K} \sum_{k=1}^{K} f(IV_{i,t}; \hat{w}_\infty(w_0(k))) \,. \tag{4}$$

Our findings present results for CNN ensembles of different sizes, with K ranging from 1 to 100. We observe a significant enhancement in portfolio performance as the ensemble size increases up to around 50 models, after which it saturates.

Given these predictions at each month-end, all stocks are sorted into decile, equal-weighted portfolios to construct a long-short spread portfolio, "H-L", that is long the upper decile and short the lower decile. Denote $N_{\widehat{R}_{i,t}^{\text{ens}} > Q_t}$ the number of stocks in the upper decile at time t, and $N_{\widehat{R}_{i,t}^{\text{ens}} < q_t}$ the number of stocks in the lower decile at time t, the returns on "H-L" are given by:

$$R_{t+1}^{H-L} = \sum_{i=1}^{N_t} R_{i,t+1}\, w_{i,t} \;,\; w_{i,t} = \frac{\mathbf{1}_{\widehat{R}_{i,t}^{\text{ens}} > Q_t}}{N_{\widehat{R}_{i,t}^{\text{ens}} > Q_t}} - \frac{\mathbf{1}_{\widehat{R}_{i,t}^{\text{ens}} < q_t}}{N_{\widehat{R}_{i,t}^{\text{ens}} < q_t}}, \tag{5}$$

where Q_t and q_t are the 90% and 10% percentiles of the distribution of $\widehat{R}_{i,t}$ at time t.

The algorithm described above depends on two key objects: The look-back window T in (2) used for training the model and the family of non-linear functions $f(x; w)$. For the

lookback window, we follow the approach of (Gu et al., 2020b) and use an expanding window (allowing us to use all available data at time t). [12]

We consider three different families of functions $f_\ell(x; w)$, $\ell \in \{CNN1, CNN4, CNN5\}$ in our analysis. Each family is a convolutional neural network (CNN, see Section 4 for details), with $CNNi$ equipped with i hidden convolutional layers, $i = 1, 4, 5$. The number of layers of the network defines its complexity, expressive power, and ability to approximate non-linear functions. The complexity of a CNN model (defined as the number of parameters divided by the sample size, see (Kelly et al., 2021)) increases exponentially with depth. See Table 9 for details. Intuitively, we expect shallow CNNs to learn fewer and simpler features, whereas more complex CNNs will detect a greater variety of features. While such complex models are tremendously over-parametrized and severely overfit the data in-sample, their performance out-of-sample might be better due to the virtue of complexity. See (Kelly et al., 2021), (Didisheim et al., 2023), and Belkin (2021).

4 Convolutional Neural Networks

This section provides a concise overview of the rationale behind Convolutional Neural Networks (CNNs), the architecture of the proposed models, and the training methodology employed.[13] Additionally, we briefly discuss the discrete convolution operation and model optimization via stochastic gradient descent. Please refer to Appendix B for rigorous mathematical definitions.

4.1 CNN Architecture

Convolutional Neural Networks (CNNs) continue to be the state-of-the-art approach for image classification (see (Shankar et al., 2020)). We outline some of the key advantages of utilizing CNNs.

1. Spatial Hierarchy and Local Connectivity: CNNs are designed to capture spatial pat-

[12] We also report that using a rolling window of five years, which allows us to account for potential non-stationarity, leads to a performance drop.

[13] See, also, (Jiang et al., 2022).

terns and relationships between pixels in images, thanks to their use of convolutional layers. These layers apply filters to local regions of the input image, allowing the network to learn and identify patterns at various levels of abstraction (e.g., edges, shapes, and more complex features). In contrast, Deep Neural Networks (DNNs) treat all input features independently, losing the spatial information and relationships between neighboring pixels. This makes it difficult for DNNs to learn and recognize complex image patterns effectively.

2. Translational and Rotational Invariance: CNNs inherently possess translation and rotational invariance, meaning that they can recognize patterns and features regardless of their position in the image. Conversely, as mentioned earlier, DNNs treat each input independently, resulting in the loss of information.

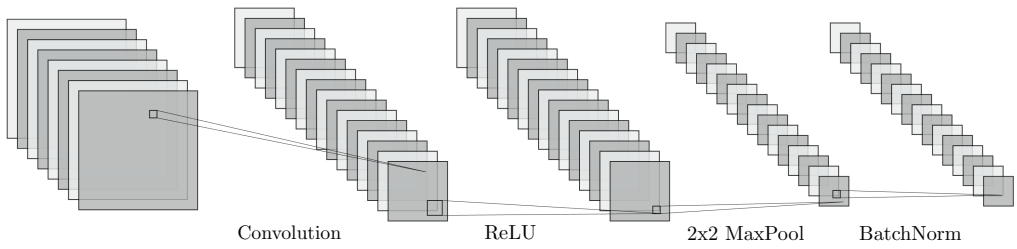


Figure 3: The figure above shows a building block of the CNN model consisting of a convolutional layer with a 3×3 filter, a ReLU layer, 2×2 max-pooling, and batch normalization layers. Note the max-pooling layer shrinks the height and width of the input by half and keeps the same depth.

We now describe a Convolutional Neural Network (CNN) as a sequence of operations to transform raw images, implied volatility surfaces in this case, into a prediction. See Appendix

Figure 4: The figure above describes our CNN4 architecture.

B for details.

A CNN core building block consists of three operations: convolution, activation, and pooling. In addition, we also use a batch normalization layer at the end of each block. A batch normalization layer normalizes the building core's output reducing the so-called *internal covariance shift* (acting as a regularizer) from one building core to the next (Ioffe and Szegedy, 2015). The core CNN building block is shown in Fig. 3. A convolution layer applies filters to the input data, capturing local spatial patterns and generating feature maps as output. An activation function is a nonlinear function applied elementwise to the output of a layer. Max-pooling is a downsampling operation in a CNN that reduces the spatial dimensions of a feature map by selecting the maximum value within a defined neighborhood, helping to retain the most prominent features while reducing computational complexity and achieving translation invariance. Global Average Pooling is a pooling operation that computes the average value of each feature map across its entire spatial extent, effectively "vectorizing" the feature map. The final CNN layer is a single fully connected node that targets the next month's stock gross return. The CNNi architectures described above are constructed by stacking i such building blocks. For example, Figure 4 shows the CNN4 architecture.[14] See Appendix B for details and precise mathematical definitions of convolutions, MaxPool, and Global Average Pooling layers.

[14]In particular, we use an increasing number of convolution filters in each block, i.e., 16, 32, 64, and 128.

4.2 Training the CNN

Our initial training uses the first seven years of observations (1996-2002) for the expanding window case. We call this the *warm-up period*. In this period, the CNN is trained for ten epochs. Then, we apply the trained model to the features of the subsequent month (unseen to the model) to make predictions. Afterward, we include this new month in the training set to retrain the model using the same procedure. We call this the *transfer-learning period*. In this period, the CNN is trained for five epochs only, as the CNN has mostly learned the market dynamics during the *warm-up period*, and only requires *fine-tuning* to adjust to an additional one month of observations.

We employ the same regularization methods as in (Gu et al., 2020a) to mitigate overfitting and facilitate efficient computation. We utilize the Xavier initializer for weight assignment in each layer, as proposed by (Glorot and Bengio, 2010), which accelerates convergence by producing initial weight values that align the prediction variance with the label scale. For loss function optimization, we combine stochastic gradient descent with the Adam algorithm (Kingma and Ba, 2014), setting the initial learning rate at 1×10^{-3} and using a batch size of 512.

5 CNN Portfolio Performance

In this section, we investigate the out-of-sample performance of CNN-based return forecasts. Given a CNN architecture, we randomly initialize its weights w_0 using 100 different random seeds and train each of these NNs using gradient descent, as described in Section 4, thus creating an ensemble of CNNs. As described in Section 3.2, we construct a long-short spread portfolio, "H-L", that is long the upper decile and short the lower decile of predicted stock returns, with a monthly holding period, see (5). Then, we compute the predictions (4) gradually increasing the ensemble size K from 1 to 100 and report the corresponding out-of-sample Sharpe ratios in Figure 13, and summarize their distribution in Table 2. Following (Jensen et al., Forthcoming) and (Didisheim et al., 2023), we perform the analysis separately for different size groups of stocks: mega (largest 20% of stocks based on NYSE monthly breakpoints), large (between 80% and 50% percentile of NYSE breakpoints), small

(between 50% and 20% percentile of NYSE breakpoints), and micro (smallest 20% of stocks). Additionally, we report the results for the non-micro group that comprises all stocks with a market cap larger than the 20% NYSE percentile.

Figure 13 clearly illustrates the *virtue of ensemble complexity*: Sharpe ratios are monotone increasing in the ensemble size and, for more complex models (CNN4 and CNN5), saturate only around $K = 40$. While a single run of CNN1, CNN4, and CNN5 achieves an annualized Sharpe Ratio of 0.80, 1.48, and 0.88, respectively, the portfolios built using ensembles of $K = 100$ randomly initialized CNN1, CNN4, and CNN5 (see (4) for the definition), achieve out-of-sample Sharpe ratios of 1.64, 2.72, and 2.50, respectively. Figure 14 shows the cumulative performance of our strategy over time and compares it with the three strongest option characteristics-based factors from (Neuhierl et al., 2022). As one can see, the complex model clearly outperforms these factors. Sharpe ratios for 100-ensemble CNN models per market cap segment are reported in Table 17.

We now investigate to what extent standard and option-based characteristics span the performance of CNN-based factors. As Table 3 shows, ensemble CNN portfolios deliver a highly significant alpha against a variety of option-based factor portfolios from (Neuhierl et al., 2022) as well as the standard Fama-French factors. Complexity also has a very strong impact on the regression R^2 : While for the lower-complexity CNN1, the R^2 is about 66%, it drops all the way to 36% for CNN4 and CNN5 models, suggesting that more complex models are able to pick up highly non-linear predictive patterns in the IV surface that are not accessible to simpler models.

Perhaps surprisingly, our most complex CNN4 and CNN5 models have only minimal and marginally significant exposure to option characteristics-based factors. It has some exposure to long-term reversal while being significantly *negatively* exposed to short-term reversal, suggesting that the predictability we identify is not driven by standard short-term statistical arbitrage. Instead, exposure to long-term reversal suggests some link between our signals and fundamental stock valuations. The low complexity CNN1 model does pick up significant exposure to the Skew factor, but as complexity increases, this exposure vanishes, confirming our intuition that complex models identify highly non-linear patterns that are not spanned by standard, linear characteristics of the IV surface.

The H-L strategy (5) is non-linear and exploits the power of the relationship between CNN-based predictions and future returns in the tails of the prediction distribution. We test the robustness of the above results by running simple, cross-sectional (Fama-MacBeth) regressions of returns on CNN-based predicted returns, using multiple characteristics as controls. Table 10 strongly supports our findings: Stock returns are indeed highly significantly related to CNN-based predictions, even after controlling for a large number of standard stock characteristics, including reversal, momentum, idiosyncratic volatility, as well as standard option-based characteristics.

We find that smaller-capitalization stocks largely drive the model performance, see Tables 4-8, nonetheless, CNN models still generate significant alpha for the not-micro group. We also find that it exhibits the same *virtue of the ensemble complexity*[15]. Notably, the results from the Fama-Macbeth regressions per-segment reveal that the statistical power of CNN forecasts is mainly concentrated in the small-cap stock group. This is highlighted in Tables 11 to 15. Value-weighted[16] portfolios built using CNN4 forecasts generate alphas that are statistically significant at a 10% level, see Table 38.

5.1 Simpler Models

The above findings clearly indicate that CNN-based models are able to extract important predictive information from the IV surface. But do we really need CNNs? Is it possible to extract the same information using simpler models?

We start by investigating whether it is possible to use all IVs in a linear fashion to predict returns. To this end, we generate predictions based on the simple ridge *panel* regression model with an expanding window by building the prediction $\hat{R}_t^{ridge}$:

$$\hat{R}_t^{ridge} = IV_t(IV_{0:t-1}^\top IV_{0:t-1} + zI)^{-1} IV_{0:t-1}^\top R_{0:t-1}, \tag{6}$$

for each period in time $t = 0, \ldots, T$ with a ridge penalty z. Here, $IV_t \in \mathbb{R}^{N_t \times |\Delta| \times |\mathcal{T}|}$ represents the panel of implied volatility surfaces for N_t stocks at time t; $IV_{0:t-1} \in \mathbb{R}^{N_{0:t-1} \times |\Delta| \times |\mathcal{T}|}$ and

[15] These results are not included and are available from the authors upon request.

[16] Weights are set in proportion to stock market capitalization, with market caps winsorized at 80% NYSE percentile as described in (Jensen et al., Forthcoming).

$R_{0:t-1} \in \mathbb{R}^{N_{0:t-1}}$ denote the expanding window of implied volatility and true gross returns, respectively, covering the time period $t = 0, \ldots, t-1$. This is a "kitchen sink" approach, whereby we are completely agnostic about the nature of predictors that the model generates. Table 18 reports the corresponding H-L strategy. As one can see, even such a simple strategy generates significant alpha relative to existing option-based factors. However, a per-group analysis, see Tables 19-23, indicates that, in fact, only the micro-group delivers significant alpha, in stark contrast to the case of CNN, see Tables 4-8.

We now turn to a simpler non-linear model, referred to as NN1 in the sequel: A single-hidden layer perceptron (a fully connected neural network). We use a relatively wide hidden layer of 128 neurons, implying a modest complexity (see Table 9). As for the CNNs, we build an ensemble of 100 randomly initialized NN1 models and find evidence for the same virtue of ensemble complexity as for the CNN models. Thus, we build our final NN1 model by averaging across 100 seeds and then build the corresponding H-L portfolios. Table 24 studies whether the NN1-based factor absorbs any significant fraction of the alpha generated by the CNN models. Comparing with Table 3, we see that both alphas significance and magnitudes are preserved: While NN1 does absorb about 50% of the CNN1 alpha, it absorbs only 25% of the alpha of more complex CNN4 and CNN5 models. At the same time, Table 25 shows that the NN1 model does not exhibit any alpha relative to the simple CNN1 model. These findings have two important implications for us. First, they imply a very strong form of the virtue of complexity, manifesting itself in the ability of CNN4 and CNN5 models to identify non-linear features that the simpler models cannot capture. Second, ignoring the geometry of the IV surface and the key *locality principle* used by CNNs for feature construction leads to highly inferior performance of the simple NN1 model. Just naively building bigger models suggested by the complexity principle of Kelly et al. (2021) and Didisheim et al. (2023) is not enough: One needs to exploit the economic structure of the data and build neural architectures that optimally exploit this structure.

5.2 Long only

Short selling might be extremely costly for some stocks, especially less liquid ones. Many anomalies have been criticized for being difficult to implement for this reason. See (Muravyev

et al., 2022). In this section, we investigate the long leg of our strategy, defined as

$$R^H_{t+1} \;=\; \sum_{i=1}^{N_t} R_{i,t+1}\, w_{i,t} \;,\; w_{i,t} \;=\; \frac{\mathbf{1}_{\widehat{R}^{\text{ens}}_{i,t} > Q_t}}{N_{\widehat{R}^{\text{ens}}_{i,t} > Q_t}} \,, \tag{7}$$

where Q_t is the 90% percentile of the predicted returns, and $N_{\widehat{R}^{\text{ens}}_{i,t} > Q_t}$ is the number of stocks in the upper decile at time t. The performance of this long-only strategy is reported in Figure 15 (cumulative returns plot) and Table 16 (regressions statistics) for all stocks. To emphasize the power of our results, we set an extremely high bar for our long-only strategy and compute the alphas with respect to the benchmark factors (including all option-based factors) that are long-short. As one can see, the performance is robust, and alphas are highly significant. Compared to Table 3, complex models lose about half of their alpha and gain (not surprisingly) a huge exposure to the market portfolio.

5.3 Transaction and Short-Sale Costs

Many stock market anomalies and factors have been criticized for generating very high turnover and hence, their performance being extremely sensitive to transaction costs. Many characteristics-based portfolios generate negative performance after costs. See, for example, (Detzel et al., 2023). As (Jensen et al., 2022a) show, efficiently exploiting machine-learning-based strategies requires optimizing portfolio positions to reduce turnover optimally. While it is possible to use the methodology of (Jensen et al., 2022a) and incorporate costs and turnover directly into our optimization algorithm, in this paper, we purposely follow a simpler approach and evaluate the performance of (5) and its long-only version (7) after costs, without applying position smoothing techniques.

We apply linear transaction fees [17] and short-sale costs to all our portfolios (CNN1, CNN4, CNN5, NN1, and Ridge) and, for a fair comparison, to all option-based factors from (Neuhierl et al., 2022) against which we benchmark our models. We do not apply fees to the factors that are not option-based (e.g., momentum, reversal, Fama-French factors), which makes our analysis even more conservative. Following (Crego et al., 2023), we apply the

[17] Restricting the analysis to linear transaction costs abstracts from important price impact considerations, see (Jensen et al., 2022a).

method that accounts for transaction costs at the portfolio level, which is described below. At a given time t, denote $w_t \in R^{N_t}$ the weight vector with coordinates

$$w_{i,t} = \frac{\mathbf{1}_{\widehat{R}_{i,t}^{\text{ens}} > Q_t}}{N_{\widehat{R}_{i,t}^{\text{ens}} > Q_t}} - \frac{\mathbf{1}_{\widehat{R}_{i,t}^{\text{ens}} < q_t}}{N_{\widehat{R}_{i,t}^{\text{ens}} < q_t}}, \tag{8}$$

i.e., our portfolios (CNN1, CNN4, CNN5, NN1, and Ridge) are equal-weighted, and, at a given t, all positive (negative) weights sum up to 1 (-1). We abuse the notation and use $N_t \cup N_{t+1}$ to denote the set of stocks available for trading at times t and $t+1$. Then, the unit cost (per dollar of investment) of rebalancing and short-selling at time $t+1$ is given by

$$\begin{aligned} \bar{f}_{t+1} &= \sum_{i \in N_t \cup N_{t+1}} (f_i |w_{i,t+1} - \frac{1 + r_{i,t+1}}{1 + r_{t+1}} w_{i,t}| + (-w_{i,t})^+ \theta_i) \\ &= \theta_i + \sum_{i \in N_t \cup N_{t+1}} f_i |w_{i,t+1} - \frac{1 + r_{i,t+1}}{1 + r_{t+1}} w_{i,t}|, \end{aligned} \tag{9}$$

where r_{t+1} stands for the total return on the long-short portfolio with weights defined in (8) and $x^+ = \max(x, 0)$. Notice that we used that weights set at time t change from t to $t+1$ relative to the total portfolio value by $\frac{1+r_{i,t+1}}{1+r_{t+1}}$. This reflects the fact that the security value in the portfolio compounds from t to $t+1$, and the portfolio value compounds from t to $t+1$ as a whole. For simplicity, we assume that short-sale costs are paid out at the beginning of each holding period (this is why we use $w_{i,t}$ in the $(-w_{i,t})^+ \theta_i$) term). We also used that $\sum_{i \in N_t \cup N_{t+1}} (-w_{i,t})^+ = 1$ by definition (8). Now, let I_t denote the total investment value of the portfolio at time t. The investment value evolves from $t+1$ to $t+2$ as follows:

$$I_{t+2} = I_{t+1}(1 - \bar{f}_{t+1}) r_{t+2}, \tag{10}$$

where r_{t+2} is pre-fee gross return on the portfolio of stocks between $t+1$ and $t+2$, and I_{t+1} is pre-$(t+1)$-rebalancing investment value. Regrouping, we get the net-of-fee net return on the portfolio

$$\frac{I_{t+2}}{I_{t+1}} - 1 = (1 - \bar{f}_{t+1}) r_{t+2} - 1. \tag{11}$$

This simple identity allows us to deduct linear transaction costs without keeping track of the investment value but by discounting net-of-fee returns only.

We set transaction costs f_i for a stock i in the micro-cap segment to be two times the cost level for other segments so that

$$f_i = (1 + \mathbf{1}_{i \in microcap}) f_{base} \tag{12}$$

and we investigate two settings: $f_{base} = 0.001$ (ten basis points (bps)) and $f_{base} = 0.002$ (twenty bps). To account for costly short-selling, we assume a fixed monthly cost of

$$\theta_i = f_i. \tag{13}$$

For $f_{base} = 0.002$, this corresponds to 40 bps (= 4.8% per annum) short-sale cost for micro-stocks and 20 bps (=2.5% per annum) short-sale cost for non-micro stocks.[18]

Again, to ensure a fair comparison, we apply the same fee structure to all option-based factor portfolios from Neuhierl et al. (2022), and we do not apply fees to other factors, which adds to the challenge for CNN-based portfolios to maintain their superiority.

Tables 26 to 37 present the performance of long-short portfolios net of linear transaction costs and short-sale fixed monthly costs, at two different cost levels: 10 bps (and 20 bps for the micro-cap segment) and 20 bps (and 40 bps for the micro-cap segment). In both cases, the micro-cap segment shows high and significant net-of-fee excess returns for CNNs, and, notably, CNN4 also exhibits consistent excess performance in the not-micro-cap segment at 5% (10%) significance level for the 10 (20) bps cost level. On the other hand, the performance of NN1 and the ridge model diminishes, indicating that it's not solely the micro-cap segment that drives excess returns for CNNs. Overall, CNN4 withstands tests with conservative linear transaction and short-sale costs, even outside the micro-cap segment.

[18] These estimates are very conservative. According to (D'avolio, 2002), most stocks have short-selling costs below 1% per annum.

6 Principal Linear Features

Despite the abundance of data and the recent emergence of the "factor zoo" with numerous characteristics identified as return predictors, predictive relationships in economic and financial data are commonly believed to be sparse. While characteristic-based sparsity (the hypothesis that only a few characteristics matter for economic relationships) is likely an illusion (Giannone et al., 2021), many papers argue that there exists some form of *"linear feature sparsity"*, usually formulated in terms of the principal components (Kozak et al., 2020). Namely, only a few top principle components are believed to be responsible for most of the predictable variation in returns.

Literature on the predictive content of the IV surface largely relies on the idea of linear feature sparsity and studies only a small number of signals represented as linear transformations of the IV surface, for instance, CIV, PIV, IVS_{atm}, IVS_{otm}, and related option characteristics described above are all examples of linear transformations of the IV surface. It is thus natural to ask a more general question of whether it is possible to use some non-parametric statistical techniques to extract relevant, linear combinations of the IV surface with the most predictive content.[19] Under the linear feature sparsity hypothesis, we expect that the true dependence

$$f(IV_{i,t}) \;=\; E[R_{i,t+1}|IV_{i,t}] \tag{14}$$

of expected returns on implied volatilities is given by a *low-rank function*, as formalized in the following definition.

Definition 1 *Let $\mathcal{S} \subseteq \mathbb{R}^d$. A function $f : \mathcal{S} \to \mathbb{R}$ is said to have rank r on $\mathcal{S}$ if there exists a low-rank matrix $M \in \mathbb{R}^{r\times d}$, $r < d$, and a function $g : M\mathcal{S} \to \mathbb{R}$ such that $f(x) = g(Mx)$ $\forall x \in \mathcal{S}$.*

[19] One common way of detecting sparsity in a non-linear model is by measuring feature importance. If most features get a low importance score, then one can conclude that the model is effectively sparse. The standard way of measuring feature importance is based on the Shapley value. See, e.g., (Lundberg and Lee, 2017). This approach is designed to measure the importance of a given feature that is expected to contain predictive information. This is the case for standard stock characteristics such as momentum, value, reversal, etc. (see, e.g., (Jensen et al., 2022a)). However, in the case of the IV surface, the features are individual IVs. Hence, it is unlikely that a given IV for a specific (moneyness, maturity) combination contains distinct predictive information, and removing just one point on the IV surface will likely not impact model performance.

In the context of our machine learning task, $\hat{x} = Mx \in \mathbb{R}^r$ is the r-dimensional vector of *linear features*, i.e., those linear transformations of x that matter for the function f. To determine whether a given function is low-rank, we will use the following observation from (Constantine et al., 2014) and (Radhakrishnan et al., 2022):

Lemma 1 *Let $\mathcal{S} \subset \mathbb{R}^d$ be an open subset and $f(x) : \mathcal{S} \to \mathbb{R}$ be a real analytic function such that $\nabla f(x)$ is bounded on $\mathcal{S}$. Let also X be a random vector taking values in $\mathcal{S}$, such that its density $p(x) : \mathcal{S} \to \mathbb{R}_+$ is Lebesgue-almost surely positive on $\mathcal{S}$. Define*

$$M_* \;=\; E[\nabla f(X) \nabla f(X)'] \;\in\; \mathbb{R}^{d \times d}. \tag{15}$$

Suppose that $\operatorname{rank} M_* = r$, *and let $M_* = UDU'$ be its eigenvalue decomposition, where $D = \operatorname{diag}(\lambda_1, \cdots, \lambda_r)$ is the vector of non-zero eigenvalues of M_* and $U \in \mathbb{R}^{d \times r}$ is the matrix of corresponding eigenvectors. Then, for Lebesgue almost every x_0, $g(y) = f((I - UU')x_0 + Uy)$ satisfies $f(x) = g(U'x) \forall x \in \mathcal{S}$. Conversely, any function $f : \mathcal{S} \to \mathbb{R}$ of rank r has* $\operatorname{rank} E[\nabla f(X) \nabla f(X)'] \leq r$. *Furthermore, if $f(x) = g(Mx)$ where $M \in \mathbb{R}^{r \times d}$ has rank exactly r and $g(y)$ is real analytic on $M\mathcal{S}$ and the functions $\nabla_{y_i} g(y)$, $i = 1, \cdots, r$, are linearly independent, then,* $\operatorname{rank} E[\nabla f(X) \nabla f(X)'] = r$.

This lemma motivates the following definition:

Definition 2 *We refer to the eigenvectors of the gradient outer product matrix (15) as the principal linear features.*

By definition, principal linear features of f are the directions along which the function varies the most.

We now show how principal linear features can be used to capture a large fraction of variation of the f function. To state the result, we will make use of the Poincare inequality (Leoni, 2017): We say that a probability measure $p(x)dx$ on $\Omega \subset \mathbb{R}^d$ satisfies the Poincare inequality on Ω if there exists a constant C such that

$$\operatorname{Var}[f(x)] \;\leq\; C\, E[\|\nabla f(x)\|^2] \tag{16}$$

for any continuously differentiable function f. Here, we use this inequality for Gaussian measure (Brascamp and Lieb, 1976): If $p(x)$ is a Gaussian measure with the covariance matrix Σ on $\Omega = \mathbb{R}^n$, then $C = \lambda_1(\Sigma)$, where λ_1 is the largest eigenvalue of Σ. It is possible to prove the following result using subtle properties of the Gaussian measures.

Lemma 2 *Let $M_* = UDU'$ be the eigenvalue decomposition of M_* and U_p the matrix whose columns are the top p eigenvectors of M_*. Suppose that the data $x \sim N(\mu, \Sigma)$ is normally distributed. Let $f_p(y) : \mathbb{R}^p \to \mathbb{R}$ be defined via $f_p(y) = E[f(x)|U_p'x = y]$. Then,*

$$E[(f(x) - f_p(U_p'x))^2] \;\leq\; \lambda_1(\Sigma)\,\Lambda(M_*)\,, \tag{17}$$

where

$$\Lambda_{-p}(M_*) \;=\; \sum_{i=p+1}^{d} \lambda_i(M_*)\,. \tag{18}$$

Lemma 2 shows that $f_p(U_p'x)$ represents a good approximation to $f(x)$ as long as the residual variation $\Lambda_{-p}(M_*)$ is sufficiently small. The linear feature sparsity hypothesis is then simply a claim about the quality of the approximation of Lemma 2. The smaller the residual variation $\Lambda_{-p}(M_*)$, the more accurate the approximation is. To validate our analysis, we verify that $\Lambda_{-p}(M_*)$ is reasonably small for each CNNi model and each p of interest.

Algorithm 1 describes the procedure of building the low-rank counterparts of the ensemble function $f^{ens}(x) = \frac{1}{K}\sum_{k=1}^{K} f(x; \hat{w}_\infty(w_o(k)))$. Using p-predictions obtained with $f_p^{ens}(x)$ counterparts, we build long-short portfolios and compare them by the Sharpe ratio.

Figure 5 shows the out-of-sample Sharpe ratios of this strategy as a function of p. We find a striking *virtue of feature complexity:* The out-of-sample performance is monotone increasing for $p > 50$, and the increase is slow. Even for $p = 100$, the Sharpe ratio is still significantly below the full model (corresponding to $p = 180$ linear features). Even for the low complexity CNN1 model, $p = 100$ recovers only about half of the full model Sharpe ratio, and the effect is even stronger for more complex models. It means that we need more than 100 linear features of the IV surface to capture the predictive relationships identified by the CNN models.

Algorithm 1 Linear Feature Sparsity

Require: Full sample of $IV_{i,t} \in \mathbb{R}^{18\times 10}$ for each stock $i \in N_t$ available at time t, and ensembles of CNNs $f_t^{ens}(\cdot)$ trained until t for all $t \in T$.

Compute the total number of $IV_{i,t}$ observations in the sample $N = \sum_t^T N_t$,
Compute average outer gradient product $M = \frac{1}{N}\sum_{i,t} \nabla f_T^{ens}(IV_{i,t}) \nabla f_T^{ens}(IV_{i,t})'$,
Perform eigenvalue decomposition of $M = UDU'$,
for $p \in [1, \ldots, 180]$ **do**
    Pick eigenvectors U_p corresponding to p largest eigenvalues, and denote U_{-p} the remainder,
    Let $\text{vec}(x)$ be the vertical stack of flattened $IV_{i,t}$ for each i, t, i.e. $\text{vec}(x) \in \mathbb{R}^{N\times 180\times 1}$,
    Compute $\mu_y = \frac{1}{N}\sum_{i,t} U_p \text{vec}(x)$,
    Compute $\mu_z = \frac{1}{N}\sum_{i,t} U_{-p} \text{vec}(x)$,
    Compute $\Sigma_x = \text{Cov}(\text{vec}(x), \text{vec}(x)) \in \mathbb{R}^{180\times 180}$,
    Compute $\Sigma_z = U'_{-p}\Sigma_x U_{-p}$,
    Compute $\Sigma_{zy} = U'_{-p}\Sigma_x U_p$,
    Compute $\Sigma_y = U'_p \Sigma_x U_p$,
    Compute $\hat{\Sigma}_z = \Sigma_z - \Sigma_{zy}\Sigma_y^{-1}\Sigma_{yz}$,
    Sample $\epsilon \sim \mathcal{N}(0,1) \in \mathbb{R}^{n\times(180-p)}$,
    for i, t **do**
        Compute $\text{vec}(y_{i,t}) = U'_p \text{vec}(x_{i,t})$, where
        Sample N_ε realisations of $\text{vec}(z|y_{i,t,k}) = \mu_z + \Sigma_{zy}\Sigma_y^{-1}(\text{vec}(y_{i,t}) - \mu_y) + \hat{\Sigma}_z^{1/2}\varepsilon_k$, $k = 1, ..., N_\varepsilon$,
        Estimate $f_{p,t}^{ens}(y_{i,t}) = \mathbb{E}\big[f_t^{ens}(z + U_p y_{i,t})|y_{i,t}\big] = \frac{1}{N_\varepsilon}\sum_{k=1}^{N_\varepsilon} f_t^{ens}(z|y_{i,t,k} + U_p y_{i,t})$.
    end for
end for
We end up with N new $f_{p,t}^{ens}(y_{i,t})$ for p-predictions (for each p of interest) for each i, t by which new p-portfolios are sorted.

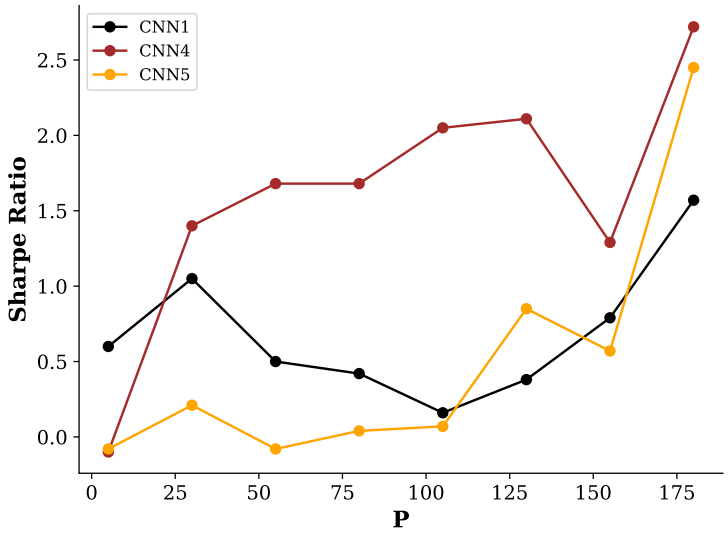


Figure 5: The figures above show the Sharpe Ratio of our H-L strategy (5) as a function of P, the number of principal features, based on the function $f_P(x)$ constructed using Algorithm 1. The experiment is run separately for each of the CNN1, CNN4, and CNN5 models.

7 Conclusion

The remarkable growth of the factor zoo (Feng et al., 2020), (Bryzgalova et al., 2023) over the last few years has been accompanied by the development of machine learning methods for asset pricing Gu et al. (2020b). As Kelly et al. (2021) and Didisheim et al. (2023) explain, this is no coincidence: Factor zoo is a natural consequence of *complexity*: A highly non-linear predictive relationship between returns and characteristics. The most naive and direct way of exploiting this complexity is to build large, unstructured non-linear models such as simple, fully connected neural networks of Gu et al. (2020b) or the random feature models of Kelly et al. (2021) and Didisheim et al. (2023). While this approach works well with unstructured stock characteristics, it is unsuited for structured data, such as the IV surface. To deal with such data, one needs to develop tools and ML algorithms that exploit the data structure optimally. In this paper, we take a step in this direction and propose Convolutional Neural Networks (CNN) architecture designed specifically to extract features of the IV surface that respect locality, as economic theory requires. We show that CNNs can successfully identify highly complex non-linear relationships that cannot be learned with naive, fully-connected networks. Importantly, we find that consistent with the existing evidence for image data

Lakshminarayanan et al. (2017), the loss landscape of the CNN is extremely non-convex and is characterized by a very large number of local minima. All those minima contain information about returns. Exploiting them requires using an ensemble of CNNs, and we document a very large *virtue of ensemble complexity*. Gaining insights into the incremental information offered by the model as it converges to different local minima for other return prediction problems (including even simpler ones, with the fully connected networks of Gu et al. (2020b)) is an important direction for future research.

Conventional wisdom based on the numerous manually constructed option characteristics suggests that a few linear features of the IV surface (e.g., level, slope, skew, and convexity) should fully summarize its predictive content. To test this "linear feature sparsity hypothesis," we introduce a novel object in financial machine learning, the *gradient outer product*, whose eigenvectors, the *principal linear features*, are natural analogs of principal components for machine learning (Radhakrishnan et al., 2022). We find no evidence for linear feature sparsity and show that a very large number (more than 100) of linear features are necessary to explain the predictive content of IV, manifesting a very high *feature complexity*. Investigating principal linear features for other ML models and datasets might bring interesting novel insights into the different notions of sparsity in return prediction.

References

Amaya, D., P. Christoffersen, K. Jacobs, and A. Vasquez, 2015, Does realized skewness predict the cross-section of equity returns?', *Journal of Financial Economics* 118, 135–167.

An, B.-J., A. Ang, T.G. Bali, and N. Cakici, 2014, The joint cross section of stocks and options', *The Journal of Finance* 69, 2279–2337.

Andersen, T.G., N. Fusari, and V. Todorov, 2015, The risk premia embedded in index options', *Journal of Financial Economics* 117, 558–584.

Ang, A., J. Chen, and Y. Xing, 2006, Downside risk', *Review of Financial Studies* 19, 1191–1239.

Attwell, David, and Simon B Laughlin, 2001, An energy budget for signaling in the grey matter of the brain, *Journal of Cerebral Blood Flow & Metabolism* 21, 1133–1145.

Avramov, Doron, Si Cheng, and Lior Metzker, 2021, Machine learning versus economic restrictions: Evidence from stock return predictability, *Available at SSRN 3450322* .

Bali, T., and S. Murray, 2019, In search for a factor model for optionable stocks.

Bali, T.G., and A. Hovakimian, 2009, Volatility spreads and expected stock returns', *Management Science* 55, 1797–1812.

Bali, Turan G, Fousseni Chabi-Yo, and Scott Murray, 2022, A factor model for stock returns based on option prices, *Available at SSRN 4071995* .

Baltussen, G., S. Bekkum, and B. Grient, 2018, Unknown unknowns: uncertainty about risk and stock returns', *Journal of Financial and Quantitative Analysis* 53, 1615–1651.

Banz, R.W., and M.H. Miller, 1978, Prices for state-contingent claims: Some estimates and applications', *Journal of Business* 653–672.

Begin, J.-F., C. Dorion, and G. Gauthier, 2020, Idiosyncratic jump risk matters: Evidence from equity returns and options', *Review of Financial Studies* 33, 155–211.

Belkin, Mikhail, 2021, Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation, *Acta Numerica* 30, 203–248.

Bollerslev, T., and V. Todorov, 2014, Time-varying jump tails', *Journal of Econometrics* 183, 168–180.

Bollerslev, T., V. Todorov, and L. Xu, 2015, Tail risk premia and return predictability', *Journal of Financial Economics* 118, 113–134.

Bollerslev, Tim, George Tauchen, and Hao Zhou, 2009, Expected stock returns and variance risk premia, *The Review of Financial Studies* 22, 4463–4492.

Borovička, J., L.P. Hansen, and J.A. Scheinkman, 2016, Misspecified recovery', *The Journal of Finance* 71, 2493–2544.

Boyer, B., T. Mitton, and K. Vorking, 2010, Expected idiosyncratic skewness', *Review of Financial Studies* 23, 169–202.

Brascamp, Herm Jan, and Elliott H Lieb, 1976, On extensions of the brunn-minkowski and prékopa-leindler theorems, including inequalities for log concave functions, and with an application to the diffusion equation, *Journal of functional analysis* 22, 366–389.

Breeden, Douglas T, and Robert H Litzenberger, 1978a, Prices of state-contingent claims implicit in option prices, *Journal of business* 621–651.

Breeden, D.T., and R.H. Litzenberger, 1978b, Prices of state-contingent claims implicit in option prices', *The Journal of Business* 51, 621–651.

Bryzgalova, Svetlana, Jiantao Huang, and Christian Julliard, 2023, Bayesian solutions for the factor zoo: We just ran two quadrillion models, *The Journal of Finance* 78, 487–557.

Bryzgalova, Svetlana, Markus Pelger, and Jason Zhu, 2020, Forest through the trees: Building cross-sections of stock returns, *Available at SSRN 3493458* .

Carhart, M.M., P. Carr, and D.B. Madan, 1997, On persistence in mutual fund performance', *The Journal of Finance* 52, 57–82, Towards the theory of volatility trading, in R. Jarrow, ed., 'Risk book on.

Chen, Luyang, Markus Pelger, and Jason Zhu, 2019, Deep learning in asset pricing, *arXiv preprint arXiv:1904.00745* .

Chen, Qihui, Nikolai Roussanov, and Xiaoliang Wang, 2021, Semiparametric conditional factor models: Estimation and inference, *arXiv preprint arXiv:2112.07121* .

Chinco, Alex, Adam D Clark-Joseph, and Mao Ye, 2019, Sparse signals in the cross-section of returns, *The Journal of Finance* 74, 449–492.

Chordia, T., A. Kurov, D. Muravyev, and A. Subrahmanyam, 2020, Index option trading activity and market returns', management science forthcoming.

Cochrane, John H., 2011, Presidential address: Discount rates, *The Journal of Finance* 66, 1047–1108.

Conrad, J., R.F. Dittmar, and E. Ghysels, 2013, Ex ante skewness and expected stock returns', *The Journal of Finance* 68, 85–124.

Constantine, Paul G, Eric Dow, and Qiqi Wang, 2014, Active subspace methods in theory and practice: applications to kriging surfaces, *SIAM Journal on Scientific Computing* 36, A1500–A1524.

Crego, Julio, Jens Soerlie Kvaerner, and Marc Stam, 2023, Machine learning and expected returns, *Available at SSRN 4345646* .

Cremers, Martijn, and David Weinbaum, 2010, Deviations from put-call parity and stock return predictability, *The Journal of Financial and Quantitative Analysis* 45, 335–367.

Davis, Larry S, 1975, A survey of edge detection techniques, *Computer graphics and image processing* 4, 248–270.

Davis, MARK HA, 2011, The dupire formula, *Imperial College London, Finite Difference Methods Course material* .

Dayan, Peter, and Laurence F Abbott, 2005, *Theoretical neuroscience: computational and mathematical modeling of neural systems* (MIT press).

DeMiguel, Victor, Alberto Martin-Utrera, Francisco J Nogales, and Raman Uppal, 2020, A transaction-cost perspective on the multitude of firm characteristics, *The Review of Financial Studies* 33, 2180–2222.

Detzel, Andrew, Robert Novy-Marx, and Mihail Velikov, 2023, Model comparison with transaction costs, *The Journal of Finance* .

Dew-Becker, I., and S. Giglio, 2020, Cross-sectional uncertainty and the business cycle: evidence from 40 years.

Didisheim, Antoine, Shikun Ke, Bryan T Kelly, and Semyon Malamud, 2023, Complexity in factor pricing models, *Swiss Finance Institute Research Paper* .

Duarte, J., C. Jones, and H. Wang, 2022, Very noisy option prices and inference regarding the volatility risk premium.

D'avolio, Gene, 2002, The market for borrowing stock, *Journal of financial economics* 66, 271–306.

Feng, Guanhao, Stefano Giglio, and Dacheng Xiu, 2020, Taming the factor zoo: A test of new factors, *The Journal of Finance* 75, 1327–1370.

Feunou, B., R.L. Aliouchkin, R. Tedongap, and L. Xu, 2019, Loss uncertainty, gain uncertainty, and expected stock returns.

Feunou, B., M.R. Jahan-Parvar, and C. Okou, 2018, Downside variance risk premium', *Journal of Financial Econometrics* 16, 341–383.

Fort, Stanislav, Huiyi Hu, and Balaji Lakshminarayanan, 2019, Deep ensembles: A loss landscape perspective, *arXiv preprint arXiv:1912.02757* .

Frazzini, Andrea, and Lasse Heje Pedersen, 2014, Betting against beta, *Journal of Financial Economics* 111, 1 – 25.

Freyberger, Joachim, Andreas Neuhierl, and Michael Weber, 2020, Dissecting characteristics nonparametrically, *The Review of Financial Studies* 33, 2326–2377.

Giannone, Domenico, Michele Lenza, and Giorgio E Primiceri, 2021, Economic predictions with big data: The illusion of sparsity, *Econometrica* 89, 2409–2437.

Giglio, Stefano, Bryan Kelly, and Dacheng Xiu, 2022, Factor models, machine learning, and asset pricing, *Annual Review of Financial Economics* 14.

Giglio, Stefano, and Dacheng Xiu, 2021, Asset pricing with omitted factors, *Journal of Political Economy* 129, 1947–1990.

Glebkin, Sergei, Semyon Malamud, and Alberto Teguia, 2023, Illiquidity and higher cumulants, *The Review of Financial Studies* 36, 2131–2173.

Glorot, Xavier, and Yoshua Bengio, 2010, Understanding the difficulty of training deep feedforward neural networks, in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249–256.

Gu, S., B. Kelly, and D. Xiu, 2020a, Empirical asset pricing via machine learning', *The Review of Financial Studies* 33, 2223–2273.

Gu, Shihao, Bryan Kelly, and Dacheng Xiu, 2020b, Empirical asset pricing via machine learning, *The Review of Financial Studies* 33, 2223–2273.

Guijarro-Ordonez, Jorge, Markus Pelger, and Greg Zanotti, 2021, Deep learning statistical arbitrage, *arXiv preprint arXiv:2106.04028* .

Han, Y., F. Liu, and X. Tang, 2020, *The information content of the implied volatility surface: Can option prices predict jumps?'* (Working Paper).

Han, Yufeng, Ai He, David Rapach, and Guofu Zhou, 2019, Expected stock returns and firm characteristics: E-lasso, assessment, and implications, *SSRN* .

Harvey, C., and A. Siddique, 2000, Conditional skewness in asset pricing tests, *Journal of Finance* 55, 1263– 1295.

Harvey, Campbell R, Yan Liu, and Heqing Zhu, 2016, ... and the cross-section of expected returns, *The Review of Financial Studies* 29, 5–68.

Hou, Kewei, Chen Xue, and Lu Zhang, 2020, Replicating anomalies, *The Review of Financial Studies* 33, 2019–2133.

Ioffe, Sergey, and Christian Szegedy, 2015, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, *32nd International Conference on Machine Learning, ICML 2015* 1, 448–456.

Jensen, Christian Skov, David Lando, and Lasse Heje Pedersen, 2019, Generalized recovery, *Journal of Financial Economics* 133, 154–174.

Jensen, Theis Ingerslev, Bryan T Kelly, Semyon Malamud, and Lasse Heje Pedersen, 2022a, Machine learning and the implementable efficient frontier, *Available at SSRN 4187217* .

Jensen, Theis Ingerslev, Bryan T. Kelly, and Lasse Heje Pedersen, 2022b, Is there a replication crisis in finance?, *Journal of Finance* .

Jensen, Theis Ingerslev, Bryan T Kelly, and Lasse Heje Pedersen, Forthcoming, Is there a replication crisis in finance?, Technical report, Journal of Finance.

Jiang, Jingwen, Bryan Kelly, and Nber Dacheng Xiu, 2022, (Re-)Imag(in)ing Price Trends, Technical report.

Jiang, L., K. Wu, G. Zhou, and Y. Zhu, 2020, Stock return asymmetry: Beyond skewness, *Journal of Financial and Quantitative Analysis* 55, 357–386.

Johnson, T.L., and E.C. So, 2012, The option to stock volume ratio and future returns, *Journal of Financial Economics* 106, 262–286.

Jones, C., H. Mo, and H. Wang, 2018, Do option prices forecast aggregate stock returns?

Kadan, O., and X. Tang, 2020, A bound on expected stock returns, *Review of Financial Studies* 33, 1565–1617.

Kelly, Bryan, Semyon Malamud, and Kangying Zhou, 2021, The virtue of complexity in return prediction, *Swiss Finance Institute Research Paper* .

Kelly, Bryan, Seth Pruitt, and Yinan Su, 2020, Characteristics are covariances: A unified model of risk and return, *Journal of Financial Economics* .

Kilic, M., and I. Shaliastovich, 2019, Good and bad variance premia and expected returns, *Management Science* 65, 2522–2544.

Kingma, Diederik P, and Jimmy Ba, 2014, Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980* .

Kozak, Serhiy, Stefan Nagel, and Shrihari Santosh, 2018, Interpreting factor models, *The Journal of Finance* 73, 1183–1223.

Kozak, Serhiy, Stefan Nagel, and Shrihari Santosh, 2020, Shrinking the cross-section, *Journal of Financial Economics* 135, 271–292.

Kozhan, R., A. Neuberger, and P. Schneider, 2013, The skew risk premium in the equity index option market', *Review of Financial Studies* 26, 2174–2203.

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell, 2017, Simple and scalable predictive uncertainty estimation using deep ensembles, *Advances in neural information processing systems* 30.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner, 1998, Gradient-based learning applied to document recognition, *Proceedings of the IEEE* 86, 2278–2324.

Leippold, Markus, Qian Wang, and Wenyu Zhou, 2022, Machine learning in the chinese stock market, *Journal of Financial Economics* 145, 64–82.

Leoni, Giovanni, 2017, *A first course in Sobolev spaces* (American Mathematical Soc.).

Lettau, Martin, and Markus Pelger, 2020, Factors that fit the time series and cross-section of stock returns, *The Review of Financial Studies* 33, 2274–2325.

Li, Hao, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein, 2018, Visualizing the loss landscape of neural nets, *Advances in neural information processing systems* 31.

Lin, H., and V. Todorov, 2019, Aggregate asymmetry in idiosyncratic jump risk.

Lin, Min, Qiang Chen, and Shuicheng Yan, 2013, Network in network, *arXiv preprint arXiv:1312.4400* .

Liu, Yang, Guofu Zhou, and Yingzi Zhu, 2020, Maximizing the sharpe ratio: A genetic programming approach, *Available at SSRN 3726609* .

Lundberg, Scott M, and Su-In Lee, 2017, A unified approach to interpreting model predictions, *Advances in neural information processing systems* 30.

Martin, Ian WR, and Christian Wagner, 2019, What is the expected return on a stock?, *The Journal of Finance* 74, 1887–1929.

McLean, R David, and Jeffrey Pontiff, 2016, Does academic research destroy stock return predictability?, *The Journal of Finance* 71, 5–32.

Moritz, Benjamin, and Tom Zimmermann, 2016, Tree-based conditional portfolio sorts: The relation between past and future stock returns, *Available at SSRN 2740751* .

Muravyev, Dmitriy, Neil D Pearson, and Joshua Matthew Pollet, 2022, Anomalies and their short-sale costs, *Available at SSRN 4266059* .

Neuhierl, Andreas, Xiaoxiao Tang, Rasmus T. Varneskov, and Guofu Zhou, 2022, Option characteristics as cross-sectional predictors, *SSRN Electronic Journal* .

Orlowski, P., P. Schneider, and F. Trojani, 2020, On the nature of jump risk premia, Unpublished manuscript, Swiss finance institute.

Pederzoli, P., 2020, Skewness swaps on individual stocks.

Radhakrishnan, Adityanarayanan, Daniel Beaglehole, Parthe Pandit, and Mikhail Belkin, 2022, Feature learning in neural networks and kernel machines that recursively learn features, *arXiv preprint arXiv:2212.13881* .

Ross, S., 2015, The recovery theorem', *The Journal of Finance* 70, 615–648.

Schneider, P., and F. Trojani, 2019, Divergence and the price of uncertainty', *Journal of Financial Econometrics* 17, 341–396.

Schneider, P., C. Wagner, J. Zechner, and W.F. Sharpe, 2020, Low-risk anomalies?', *The Journal of Finance* 75, 2673–2718.

Shankar, Vaishaal, Alex Fang, Wenshuo Guo, Sara Fridovich-Keil, Jonathan Ragan-Kelley, Ludwig Schmidt, and Benjamin Recht, 2020, Neural kernels without tangents, in *International Conference on Machine Learning*, 8614–8623, PMLR.

Soudry, Daniel, and Yair Carmon, 2016, No bad local minima: Data independent training error guarantees for multilayer neural networks, *arXiv preprint arXiv:1605.08361* .

Stilger, P.S., A. Kostakis, and S.-H. Poon, 2017, What does risk-neutral skewness tell us about future stock returns?', *Management Science* 63, 1814–1834.

Tang, X., 2019, Variance asymmetry managed portfolios.

Xing, Y., X. Zhang, and R. Zhao, 2010, What does the individual option volatility smirk tell us about future equity returns?', *Journal of Financial and Quantitative Analysis* 641–662.

Yan, Y., 2011, Jump risk, stock returns and slope of implied volatility smile', *Journal of Financial Economics* 99, 216–223.

Yao, Zhewei, Amir Gholami, Kurt Keutzer, and Michael W Mahoney, 2020, Pyhessian: Neural networks through the lens of the hessian, in *2020 IEEE international conference on big data (Big data)*, 581–590, IEEE.

A Data Preprocessing

We get option implied volatility surfaces from the Option Metrics IvyDB. In IvyDB, each option chain is already normalized across expiration dates and deltas[20] (11 and 34 different values, respectively). We remove all rows with 10-days-to-expiration implied volatility values due to the high number of missing values at the beginning of the time frame. For each option implied volatility surface, we end up with a 2D matrix of size 10×34.

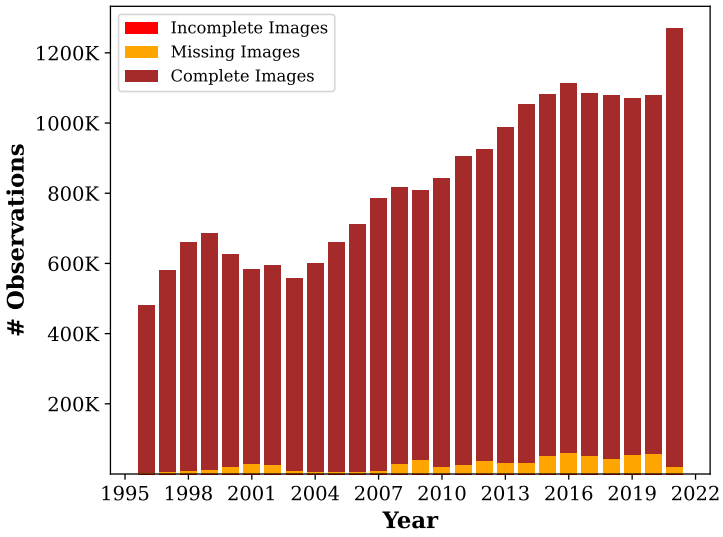


Figure 6: The figure above shows our dataset sample size. On the y-axis, we show the number of observations, while on the x-axis, we show the corresponding year. One observation is a single image obtained by stacking implied volatility surfaces for a given pair of stock and date. Incomplete Images are those observations (stock and date pair) where some implied volatility value is missing, given a particular expiration date and delta pair. Missing Images are the sample where the stock and date pair appear in the dataset but have all empty values. Complete Images are the correctly built images where the implied volatility surfaces are complete.

We link the CRSP dataset together with the CRSP delisting dataset, both available in WRDS, to take into account the delisting return as explained in subsection 3.1. As the Option Metrics database uses its own security identifier and CRSP uses the PERMNO to identify an asset uniquely, we merge these datasets thanks to a linking dataset provided by WRDS. From the linking dataset, we remove "bad" entries:

[20] In other words, each option implied volatility surface in IvyDB is interpolated across moneyness and maturities.

1. If an entry has a *score* lower than 1 is removed from our sample. A *score* of 1 means a 100% of mapping confidence.

2. Each entry has a starting date (*sdate*), ending date (*edate*), Option Metrics ID (*secid*), and the PERMNO. We double-sort the dataset using *sdate* and PERMNO. If two consecutive entries have the same PERMNO, but the first (preceding) row has *edate* higher than the second (following) row *sdate*, then the PERMNO is removed from our sample as well.

B More about Convolutional Neural Networks

B.1 The Convolution Function

In this section, we present the convolution operation in continuous time and subsequently expound upon its counterpart in the discrete domain. Convolution is a fundamental concept in mathematics and is used in many areas of science and engineering to analyze and manipulate signals and images. The operation involves taking a smaller function, called a kernel or filter, and sliding it over a larger function, called the input, to compute the area of overlap at each point. This process can be thought of as extracting local features from the input function and creating a third function that captures the interactions between the two functions. Formally, if we define the kernel function with $g(t)$ and the signal as $f(t)$, then the convolution function is defined as

$$(f * g)(t) \coloneqq \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau. \tag{19}$$

One can imagine this operation as "sliding" (to the right) the kernel function while the signal function stays still. At some point, the kernel function $g(t)$ will start overlapping with the signal function $f(t)$. The area of overlap will be less or bigger depending on the "shape" of the two functions. Equation 19 defines exactly the area where the two functions intersect. Figure 19 shows an example of convolution in continuous time.

Table 1: The table below shows our preprocessed Option Metrics statistics. Namely, Incomplete Images are those observations (stock and date pair) where some implied volatility value is missing, given a particular expiration date and delta pair. Missing Images are the sample where the stock and date pair appear in the dataset but have all empty values. Complete Images are the correctly built images where the implied volatility surfaces are complete.

Year	Missing (%)	Incomplete (%)	Complete (%)	Total
1996	3289 (0.68%)	0 (0.0%)	477674 (99.32%)	480963
1997	5975 (1.03%)	0 (0.0%)	574945 (98.97%)	580920
1998	8789 (1.33%)	2 (0.0%)	651291 (98.67%)	660082
1999	11665 (1.7%)	10 (0.0%)	673334 (98.3%)	685009
2000	19943 (3.18%)	15 (0.0%)	606628 (96.81%)	626586
2001	28265 (4.84%)	6 (0.0%)	555407 (95.16%)	583678
2002	27345 (4.59%)	2 (0.0%)	568126 (95.41%)	595473
2003	10304 (1.84%)	5 (0.0%)	548341 (98.15%)	558650
2004	6137 (1.02%)	6 (0.0%)	593240 (98.98%)	599383
2005	6203 (0.94%)	3 (0.0%)	655095 (99.06%)	661301
2006	5991 (0.84%)	4 (0.0%)	706582 (99.16%)	712577
2007	8783 (1.12%)	5 (0.0%)	776060 (98.88%)	784848
2008	29436 (3.6%)	8 (0.0%)	788981 (96.4%)	818425
2009	40659 (5.04%)	7 (0.0%)	766591 (94.96%)	807257
2010	20822 (2.47%)	7 (0.0%)	821281 (97.53%)	842110
2011	26609 (2.94%)	9 (0.0%)	877955 (97.06%)	904573
2012	38756 (4.19%)	6 (0.0%)	886891 (95.81%)	925653
2013	31810 (3.22%)	3 (0.0%)	956879 (96.78%)	988692
2014	32621 (3.1%)	0 (0.0%)	1020567 (96.9%)	1053188
2015	51663 (4.78%)	0 (0.0%)	1030061 (95.22%)	1081724
2016	60616 (5.45%)	3 (0.0%)	1052121 (94.55%)	1112740
2017	53694 (4.95%)	1 (0.0%)	1031601 (95.05%)	1085296
2018	44705 (4.14%)	3 (0.0%)	1034656 (95.86%)	1079364
2019	54570 (5.1%)	1 (0.0%)	1016092 (94.9%)	1070663
2020	57409 (5.33%)	2 (0.0%)	1020423 (94.67%)	1077834
2021	20653 (1.63%)	0 (0.0%)	1248461 (98.37%)	1269114

We now introduce the definition of convolution in case the signal is discrete:

$$(f * g)(t) \coloneqq \sum_{\tau=-\infty}^{\infty} f(a)g(t-a). \tag{20}$$

While in signal processing, the convolution function can be used to filter, smooth, or extract features from signals, in image processing and machine learning, an edge-detection kernel can be convolved with an image to highlight edges and contours. However, in this

case, we will have a two-dimensional signal (the image) I and a two-dimensional kernel K

$$(I * K)(i, j) = \sum_m \sum_n I(m, n)K(i - m, j - n). \tag{21}$$

Starting from the top-left corner of the image matrix and moving clockwise, the 2D kernel extracts features from the matrix. Figure 8 illustrates an example of 2D convolution applied to a 6×6 matrix using a 3×3 kernel. In this case, the kernel captures the first 3×3 block of the image matrix (highlighted in yellow) and generates a *feature mapping* output for the first cell by computing $23 + 255 + 34 - 66 - 67 - 89 = 90$. The kernel then moves by 2 cells to capture the number -154. The distance by which the kernel moves is referred to as the *stride*. Finally, the last column is dropped as the kernel wouldn't fit, and no padding was specified. Padding refers to adding additional pixels or values around the edges of the input image to increase its spatial dimensions and prevent the output feature maps from becoming too small. By adding padding, we can control the spatial resolution of the output feature maps and ensure that the features extracted by the convolutional layers are more representative of the original input. Normally, in the machine learning literature, the following forms of padding are used:

- Zero-padding, where additional zero values are added to the edges of the input data.
- Reflective padding, where the values at the edges of the input are reflected to create the additional padding.

We show in Figure 9 a more concrete example using a popular edge detection algorithm in computer vision (see (Davis, 1975)). Like the Sobel kernel, there are many others, like the Gaussian kernel and the Laplacian kernel. probably expand on the conclusion

B.2 The Activation Function

Now that we have a basic understanding of image processing and discrete convolution, we can discuss using activation functions in deep learning. These functions are non-linear and

are applied element-wise to the tensor, determining how a neuron should "fire." In our work, we exclusively use the Rectified Linear Unit (ReLU) activation function, defined as

$$z = \max\{0, x\} \tag{22}$$

The ReLU activation function is popular in the literature due to its simplicity and ability to create sparsity by setting negative values to zero. This can help reduce overfitting and improve generalization. Additionally, the ReLU function is inspired by real biological neuron models, such as the leaky integrate-and-fire model (see, e.g., (Attwell and Laughlin, 2001), (Dayan and Abbott, 2005), and (Glorot and Bengio, 2010)).

Figure 10 shows an example: By applying the ReLU function, the negative values are set to zero, leaving only the positive values. This helps highlight the important features in the input image and can improve the neural network's performance.

B.3 The Max-Pooling Function

Empirical data is often noisy, and images are no exception to this rule. When working with image data, noise can come from various sources, such as imperfect sensors, compression algorithms, or environmental factors. Moreover, the dimension of these images can grow exponentially when applying CNNs. To this end, (LeCun et al., 1998) has first introduced the max-pooling function.

The max-pooling function selects the maximum value within each pooling region, making the pooling operation less sensitive to small variations in the input and more resistant to noise. By reducing the dimensionality of the data, the function also helps lower the network's computational cost while improving performance.

Figure 11 shows an example of this function. At the top, a max-pool with a kernel size of 2×2 is applied to an input matrix of 2×2, giving out a single number; then, at the bottom, the same kernel is applied to a 4×2 matrix, giving in output a 2×1 vector.

B.4 The Batch-Normalization Function

When training CNNs, finding a local minimum convergence might be slow due to the distribution change for each layer in the deep architecture. While data normalization is a common practice in machine learning, the hidden layers' input is not normalized due to random parameter initialization and non-linearities. Because these inputs are not normalized by they are rather shifted, (Ioffe and Szegedy, 2015) call this phenomenon the *internal covariate shift*. The Batch-Normalization function accelerates convergence dramatically and acts as a regularizer, replacing the need for further heuristics techniques, e.g., Dropout. The Batch-Normalization function is applied at each mini-batch, and Algorithm 2 sums up the procedure.

Algorithm 2 Batch Normalization Algorithm

Require: Mini-batch of data $\{x_1, x_2, ..., x_m\}$, trainable parameters γ and β, and a small constant ϵ

1: $\mu = \frac{1}{m}\sum_{i=1}^{m} S_t$
2: $\sigma^2 = \frac{1}{m}\sum_{i=1}^{m}(S_t - \mu)^2$
3: $\hat{x}_i = \frac{S_t - \mu}{\sqrt{\sigma^2 + \epsilon}}$
4: $R_{t+1} = \gamma\hat{x}_i + \beta$

B.5 The Global Average Pooling Function

As we approach the end of the CNN architecture, it is crucial to transform the input to the final layer into a vector right before producing a regression or classification output.

This transformation process, often referred to as "flattening" or "vectorization", involves taking the multi-dimensional output from the previous layer (usually a feature map or activation map) and converting it into a one-dimensional vector. This is an essential step because the final layer, which is typically a fully connected layer (also known as a dense layer), expects its input data to be in the form of a vector. This flattened vector is then fed into the final layer to produce the desired output, such as class probabilities for classification tasks or continuous values for regression tasks.

Flattening the image into a one-dimensional vector can lead to several issues when using images as input to a neural network. One problem is that flattening the image discards the

Table 2: Annualized Sharpe Ratio Minimum & Maximum Values of the curves shown in Figure 13 for CNN1, CNN4 and CNN5 long-short portfolios from 2003 to 2022.

		SR_{full}	SR_{mega}	SR_{large}	SR_{small}	SR_{micro}	SR_{not_micro}
CNN1							
	Min	0.756	-0.181	-0.038	0.371	1.506	0.175
	Max	1.641	0.201	0.534	0.962	2.163	1.059
CNN4							
	Min	1.477	-0.156	0.140	0.647	1.300	0.757
	Max	2.724	0.204	0.613	1.502	2.365	1.470
CNN5							
	Min	0.881	-0.204	-0.340	0.096	1.111	0.001
	Max	2.501	0.174	0.511	1.383	2.184	1.377

spatial structure of the image, which can be important for capturing meaningful patterns in the data. In addition, flattening the image can result in a very high-dimensional input, which can increase the number of parameters in the model and make it more difficult to train.

In this work, we use the Global Average Pooling (GAP) (Lin et al., 2013), a technique that addresses these issues by summarizing the feature maps produced by a convolutional layer using an average pooling operation. Unlike flattening, GAP preserves the feature maps' spatial structure by computing each feature map's average value over its entire spatial extent. This reduces the dimensionality of the data, which can improve the model's efficiency while reducing the risk of overfitting.

Furthermore, GAP has been shown to have additional benefits, such as better resistance to adversarial attacks and better generalization performance than flattening. This is because GAP encourages the model to learn features that are robust to spatial transformations of the input, which can help the model generalize better to new data.

Figure 12 illustrates a shallow CNN consisting of a sequence of operations: a single convolutional layer with a ReLU activation function, followed by a max-pooling layer, and finally, a GAP layer before the input is fed into the dense network.

C Results

Table 3: **All stocks, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, and CNN5 portfolios on the factor model that includes CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	CNN1	CNN4	CNN5
Intercept	0.008***	0.012***	0.011***
	(0.002)	(0.002)	(0.002)
$r_M - r_f$	-0.047	0.016	-0.048
	(0.061)	(0.056)	(0.055)
SMB	-0.183*	-0.040	0.021
	(0.100)	(0.090)	(0.089)
HML	-0.085	-0.082	-0.132*
	(0.085)	(0.077)	(0.076)
2-12 Momentum	-0.097**	-0.081*	-0.107**
	(0.046)	(0.042)	(0.041)
ST Reversal	-0.074	-0.137**	-0.175***
	(0.067)	(0.060)	(0.060)
LT Reversal	0.153*	0.173**	0.262***
	(0.087)	(0.078)	(0.077)
CIV	0.519**	0.120	0.053
	(0.231)	(0.209)	(0.206)
PIV	-0.304	-0.025	-0.014
	(0.236)	(0.213)	(0.210)
IVS_{atm}	-0.356***	0.088	0.245**
	(0.118)	(0.106)	(0.105)
IVS_{otm}	0.679***	0.181	0.170
	(0.133)	(0.121)	(0.119)
Skew	0.600***	0.244**	0.135
	(0.125)	(0.113)	(0.111)
VOV	0.159*	0.139*	0.144*
	(0.084)	(0.076)	(0.075)
Δ CIV	0.055	0.288***	0.182*
	(0.119)	(0.107)	(0.106)
Δ PIV	0.135	0.170*	0.086
	(0.108)	(0.098)	(0.096)
Observations	227	227	227
R^2	0.657	0.367	0.344
Adjusted R^2	0.634	0.325	0.301
Residual Std. Error	0.023	0.020	0.020
F Statistic	29.003***	8.766***	7.949***
Note:			*p<0.1; **p<0.05; ***p<0.01

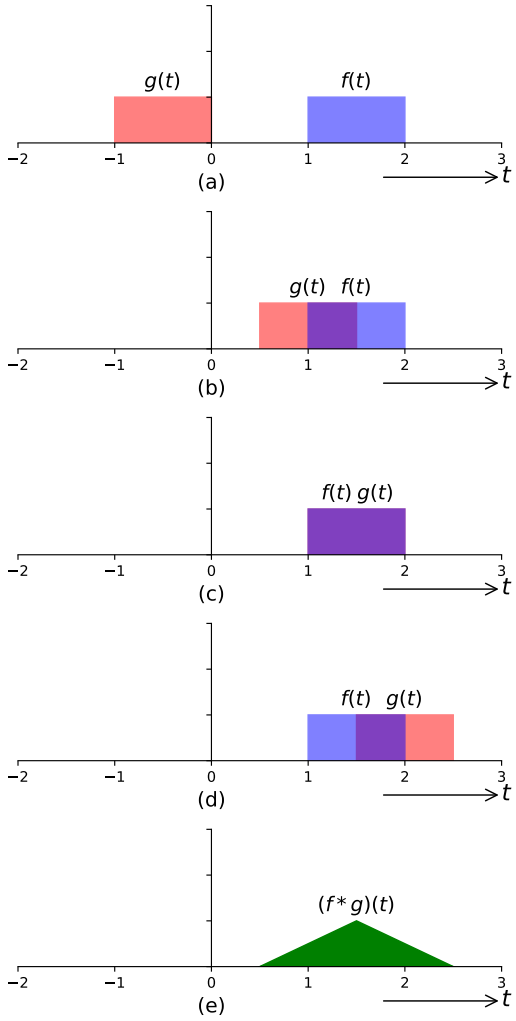


Figure 7: The figures above show the convolution operation $(f * g)(t)$ in continuous time. In this particular example, the kernel function, in red, $g(t)$, is equal to the input signal, in blue, $f(t)$, and they share the same area. In information theory, this is a known example where the convolution of two "rectangular" functions gives an output a "triangle." The more you slide the kernel function to the right, the kernel $g(t)$ will start overlapping with the signal $f(t)$. The overlapping area will start increasing (b), reaching the maximum at (c), and it will smoothly decay (d). The final figure (e) shows the convolution result.

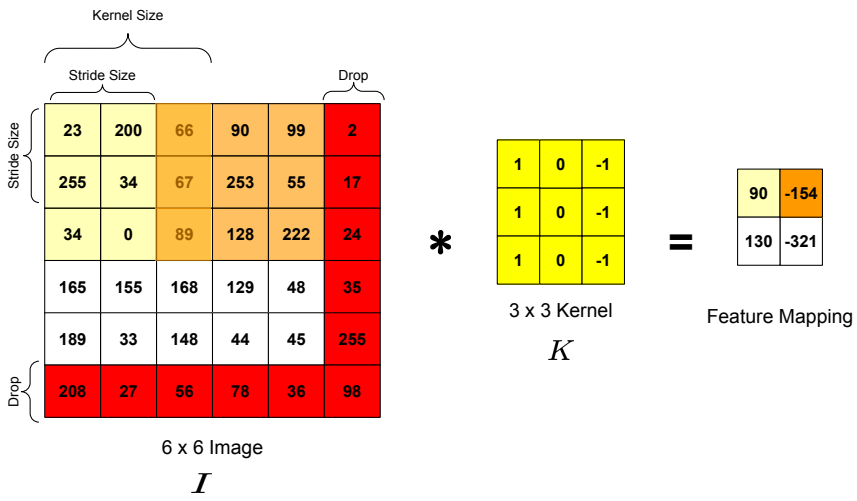


Figure 8: The figure above shows the convolution operation. A kernel filter with size 3x3 and stride equal to two is convolved to the 2-dimensional image matrix. The kernel filter moves clockwise and projects the output value to the Feature Mapping matrix. A padding is necessary to obtain the convolution in the figure. In this case, we show the "same" padding. Another popular option in literature is not to use padding at all. In case of no padding, the last column would have been dropped, and the feature mapping matrix would have been a 2x2 matrix.

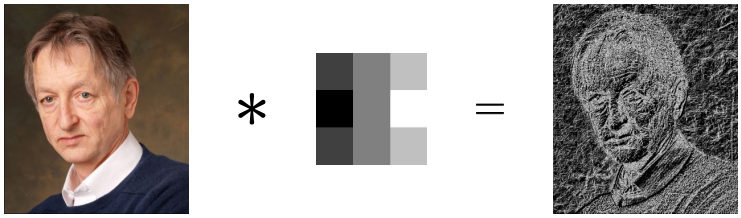


Figure 9: The figure above shows an example of an edge detection algorithm using the Sobel operator.

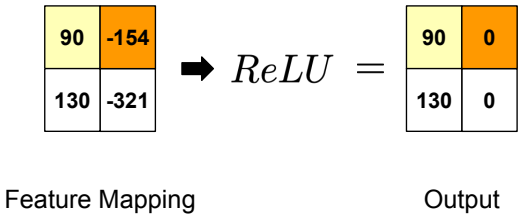


Figure 10: The figure above shows the output of a ReLU activation function applied to a 2×2 matrix.

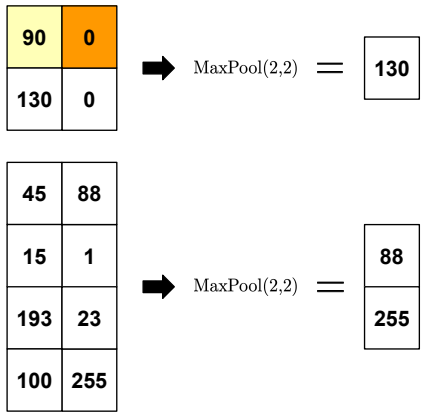


Figure 11: The figure above shows the max-pooling function. In this particular example, we show a max-pooling function with a kernel 2×2. That means the function extracts the maximum value in a pool as big as 2×2. Hence, the output of a 2×2 input has size 1×1, and similarly, the output of a 4×2 input has size 2×1.

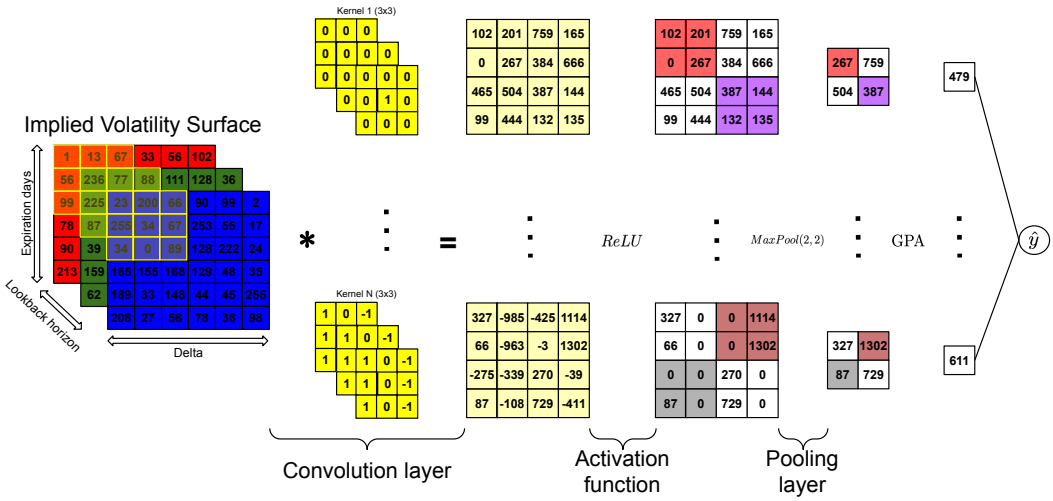
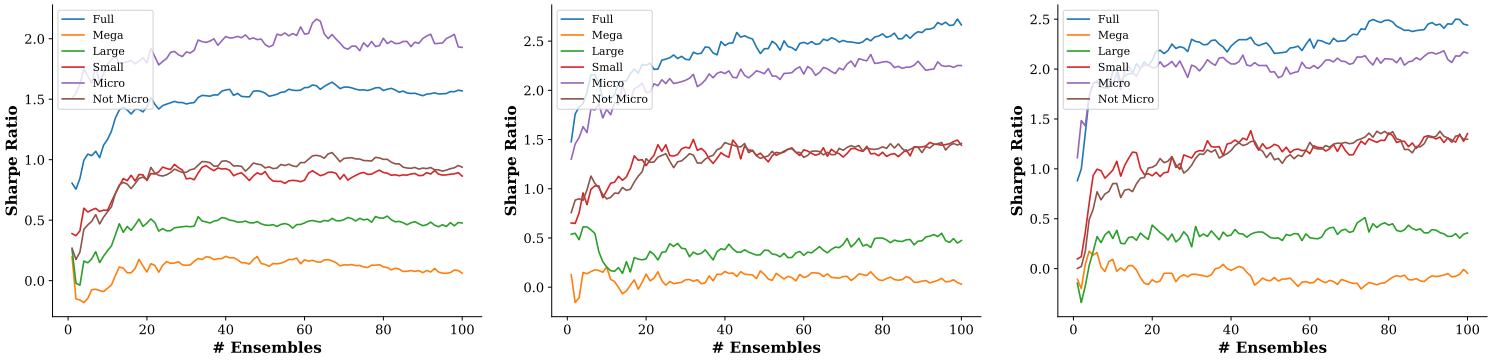


Figure 12: The figure above shows a building block of a CNN model. From left to right, we have the input image decomposed in an RGB tensor. To this tensor, a series of convolutions are applied using N different kernels, in this case, with size 3x3. Next, a ReLU activation function is applied to the previous step result. Finally, to reduce the image's complexity even more, a Pooling Layer is applied. The figure shows a classic MaxPool with kernel and stride size equal to two.



(a) (b) (c)

Figure 13: Sharpe ratios of ensemble-based returns (5) as a function of ensemble size. In particular, Figures 13a, 13b, and 13c show the performance of CNN1, CNN4, and CNN5, respectively.

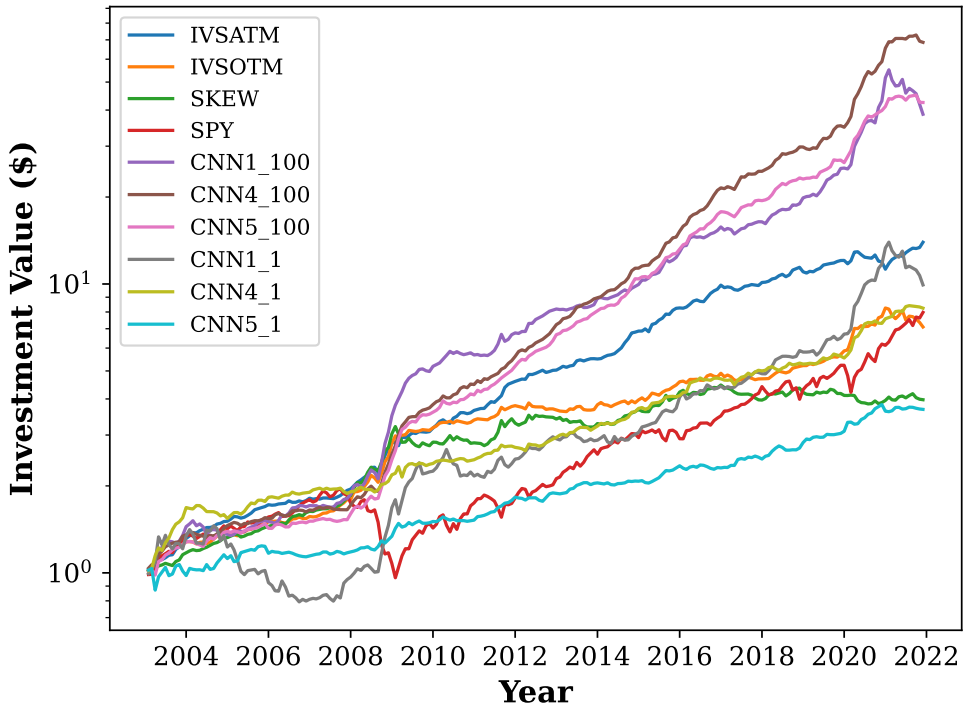


Figure 14: Cumulative Returns of the market neutral (long-short) strategy (5) for a full universe of stocks. CNNi_K refers to the ensemble of K CNNi models. IVSATM, IVSOTM, and SKEW are the top three option characteristics-based factors from (Neuhierl et al., 2022) in terms of Sharpe ratio, and SPY is the SP&500 ETF.

Table 4: **Mega-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, and CNN5 portfolios on the factor model that includes CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	CNN1	CNN4	CNN5
Intercept	-0.001	-0.001	-0.003
	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.065	0.107	0.062
	(0.073)	(0.073)	(0.071)
SMB	0.070	0.117	0.015
	(0.119)	(0.119)	(0.115)
HML	-0.061	-0.009	0.093
	(0.102)	(0.101)	(0.098)
2-12 Momentum	0.036	-0.029	-0.016
	(0.055)	(0.055)	(0.053)
ST Reversal	-0.238***	-0.154*	-0.202***
	(0.080)	(0.079)	(0.077)
LT Reversal	0.159	0.198*	0.124
	(0.103)	(0.103)	(0.100)
CIV	-0.443	-0.331	-0.444*
	(0.276)	(0.274)	(0.267)
PIV	0.206	0.242	0.376
	(0.282)	(0.280)	(0.272)
IVS_{atm}	-0.326**	0.031	0.195
	(0.141)	(0.140)	(0.136)
IVS_{otm}	0.232	-0.165	0.046
	(0.159)	(0.158)	(0.154)
Skew	0.666***	0.309**	0.147
	(0.149)	(0.148)	(0.144)
VOV	0.127	0.165*	0.209**
	(0.100)	(0.099)	(0.097)
Δ CIV	0.280**	0.246*	0.137
	(0.142)	(0.141)	(0.137)
Δ PIV	0.311**	0.106	-0.081
	(0.129)	(0.128)	(0.124)
Observations	227	227	227
R^2	0.523	0.100	0.098
Adjusted R^2	0.491	0.040	0.038
Residual Std. Error	0.027	0.027	0.026
F Statistic	16.591***	1.676*	1.636*
Note:			*p<0.1; **p<0.05; ***p<0.01

Table 5: **Large-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, and CNN5 portfolios on the factor model that includes CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	CNN1	CNN4	CNN5
Intercept	-0.000	-0.001	-0.001
	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.153**	0.085	0.016
	(0.064)	(0.058)	(0.061)
SMB	0.118	-0.080	0.013
	(0.103)	(0.095)	(0.100)
HML	-0.022	-0.118	-0.076
	(0.088)	(0.081)	(0.085)
2-12 Momentum	-0.025	-0.052	-0.066
	(0.048)	(0.044)	(0.046)
ST Reversal	-0.031	-0.088	-0.159**
	(0.069)	(0.063)	(0.067)
LT Reversal	0.029	0.195**	0.160*
	(0.090)	(0.082)	(0.086)
CIV	-0.099	-0.180	-0.379
	(0.239)	(0.219)	(0.231)
PIV	-0.077	0.176	0.367
	(0.244)	(0.223)	(0.236)
IVS_{atm}	-0.451***	0.226**	0.368***
	(0.122)	(0.111)	(0.117)
IVS_{otm}	0.684***	-0.094	-0.154
	(0.138)	(0.126)	(0.133)
Skew	0.577***	0.419***	0.375***
	(0.129)	(0.118)	(0.125)
VOV	-0.060	0.200**	0.207**
	(0.087)	(0.079)	(0.084)
Δ CIV	-0.050	0.114	0.126
	(0.123)	(0.112)	(0.118)
Δ PIV	0.166	-0.150	-0.199*
	(0.112)	(0.102)	(0.108)
Observations	227	227	227
R^2	0.534	0.199	0.194
Adjusted R^2	0.503	0.146	0.141
Residual Std. Error	0.023	0.021	0.023
F Statistic	17.321***	3.752***	3.641***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 6: **Small-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, and CNN5 portfolios on the factor model that includes CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	CNN1	CNN4	CNN5
Intercept	0.001	0.005**	0.005**
	(0.002)	(0.002)	(0.002)
$r_M - r_f$	-0.013	0.125	0.005
	(0.077)	(0.079)	(0.072)
SMB	-0.333***	-0.022	0.054
	(0.125)	(0.129)	(0.117)
HML	0.176*	0.065	0.006
	(0.107)	(0.110)	(0.099)
2-12 Momentum	0.022	0.007	-0.074
	(0.058)	(0.060)	(0.054)
ST Reversal	0.022	-0.050	-0.109
	(0.084)	(0.086)	(0.078)
LT Reversal	-0.022	-0.034	0.152
	(0.108)	(0.111)	(0.101)
CIV	0.347	0.374	0.207
	(0.290)	(0.298)	(0.270)
PIV	-0.068	-0.318	-0.181
	(0.295)	(0.304)	(0.275)
IVS_{atm}	-0.198	0.059	0.414***
	(0.147)	(0.152)	(0.137)
IVS_{otm}	0.511***	0.190	0.081
	(0.167)	(0.172)	(0.155)
Skew	0.540***	0.191	0.121
	(0.156)	(0.161)	(0.146)
VOV	-0.007	0.106	0.105
	(0.105)	(0.108)	(0.098)
Δ CIV	-0.039	0.011	-0.139
	(0.148)	(0.153)	(0.138)
Δ PIV	0.232*	0.124	-0.053
	(0.135)	(0.139)	(0.126)
Observations	227	227	227
R^2	0.457	0.132	0.190
Adjusted R^2	0.421	0.074	0.136
Residual Std. Error	0.028	0.029	0.026
F Statistic	12.723***	2.297***	3.547***
Note:			*p<0.1; **p<0.05; ***p<0.01

Table 7: **Micro-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, and CNN5 portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	CNN1	CNN4	CNN5
Intercept	0.034***	0.041***	0.038***
	(0.006)	(0.006)	(0.006)
$r_M - r_f$	-0.512***	-0.389*	-0.245
	(0.187)	(0.204)	(0.204)
SMB	-0.228	0.061	0.291
	(0.305)	(0.333)	(0.332)
HML	-0.417	-0.341	-0.456
	(0.260)	(0.283)	(0.283)
2-12 Momentum	-0.253*	-0.376**	-0.378**
	(0.141)	(0.154)	(0.154)
ST Reversal	-0.265	-0.639***	-0.670***
	(0.204)	(0.222)	(0.222)
LT Reversal	0.397	0.434	0.568**
	(0.264)	(0.288)	(0.288)
CIV	1.769**	0.543	0.459
	(0.706)	(0.770)	(0.769)
PIV	-1.351*	-0.478	-0.666
	(0.720)	(0.785)	(0.785)
IVS_{atm}	-0.304	0.553	0.440
	(0.359)	(0.391)	(0.391)
IVS_{otm}	0.858**	0.385	0.741*
	(0.406)	(0.443)	(0.443)
Skew	0.372	-0.031	-0.280
	(0.381)	(0.416)	(0.416)
VOV	0.279	1.043***	0.833***
	(0.255)	(0.279)	(0.278)
Δ CIV	0.301	0.691*	0.774*
	(0.362)	(0.394)	(0.394)
Δ PIV	0.212	0.370	0.293
	(0.329)	(0.359)	(0.359)
Observations	227	227	227
R^2	0.354	0.240	0.219
Adjusted R^2	0.312	0.190	0.167
Residual Std. Error	0.069	0.075	0.075
F Statistic	8.305***	4.778***	4.235***
Note:			*p<0.1; **p<0.05; ***p<0.01

Table 8: **Not-micro-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, and CNN5 portfolios on the factor model that includes CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	CNN1	CNN4	CNN5
Intercept	0.002	0.004**	0.003*
	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.061	0.047	0.023
	(0.057)	(0.051)	(0.048)
SMB	-0.067	0.017	0.032
	(0.093)	(0.083)	(0.079)
HML	0.018	0.029	-0.012
	(0.079)	(0.070)	(0.067)
2-12 Momentum	-0.042	-0.002	-0.026
	(0.043)	(0.038)	(0.036)
ST Reversal	-0.059	-0.001	-0.086
	(0.062)	(0.055)	(0.053)
LT Reversal	0.063	0.074	0.147**
	(0.080)	(0.071)	(0.068)
CIV	0.150	-0.076	-0.107
	(0.215)	(0.191)	(0.182)
PIV	-0.081	0.130	0.132
	(0.219)	(0.195)	(0.186)
IVS_{atm}	-0.303***	0.089	0.286***
	(0.109)	(0.097)	(0.093)
IVS_{otm}	0.490***	0.029	0.005
	(0.124)	(0.110)	(0.105)
Skew	0.582***	0.329***	0.256***
	(0.116)	(0.103)	(0.098)
VOV	0.018	0.095	0.111*
	(0.078)	(0.069)	(0.066)
Δ CIV	-0.065	0.087	-0.043
	(0.110)	(0.098)	(0.093)
Δ PIV	0.120	0.066	-0.081
	(0.100)	(0.089)	(0.085)
Observations	227	227	227
R^2	0.512	0.205	0.216
Adjusted R^2	0.479	0.153	0.164
Residual Std. Error	0.021	0.019	0.018
F Statistic	15.865***	3.908***	4.169***
Note:			*p<0.1; **p<0.05; ***p<0.01

Table 9: The number of parameters for a single, 40-ensemble, and 100-ensemble for our set of CNN models. Similarly, Single c, 40-ensemble c, and 100-ensemble c show the total model complexity as defined in (Kelly et al., 2021): $c \coloneqq P/T$, where $T = 973947$ is the total number of observations at the end of the sample (all models are trained using an expanding window).

Model	Single	40-Ensemble	100-Ensemble	Single c	40-Ensemble c	100-Ensemble c
CNN1	1921	76840	192100	0.002	0.079	0.197
CNN4	98241	3929640	9824100	0.101	4.035	10.087
CNN5	394561	15782440	39456100	0.405	16.205	40.512
NN1	23809	952360	2380900	0.024	0.978	2.445

Table 10: **All stocks, long-short portfolio.** Cross-sectional (Fama-MacBeth) regression of next-month security returns $r_{i,t+1}$ for the full stock universe on a set of predictive characteristics over the time period February 2003 to December 2021. $\hat{R}^{ens}_{t,CNN}$ is the prediction in (4). CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV and ΔPIV are option characteristics from (Neuhierl et al., 2022). $ret\_$ are momentum- and short-term-reversal-based characteristics; β are market beta characteristics; $ivol\_$ are idiosyncratic volatility characteristics. All these are taken from (Jensen et al., 2022b). T-statistics are reported in parentheses.

	CNN1	CNN4	CNN5
$\hat{R}^{ens}_{t,CNN}$	16.28***	8.30***	3.85**
	(4.38)	(4.42)	(2.44)
CIV	0.54**	0.52*	0.57**
	(2.02)	(1.91)	(2.14)
PIV	-0.70***	-0.64**	-0.68**
	(-2.58)	(-2.32)	(-2.46)
IVS_{atm}	-1.24***	-1.15***	-1.25***
	(-4.15)	(-3.80)	(-4.16)
IVS_{otm}	0.10	-0.13	-0.16
	(0.31)	(-0.40)	(-0.51)
Skew	-0.56*	-0.73**	-0.75**
	(-1.68)	(-2.19)	(-2.24)
VOV	-0.88**	-0.91**	-0.93**
	(-2.44)	(-2.50)	(-2.54)
ΔCIV	-0.22	-0.28	-0.33
	(-0.64)	(-0.85)	(-0.98)
ΔPIV	0.22	0.18	0.21
	(0.62)	(0.50)	(0.59)
ret_1_0	-0.48	-0.50	-0.54
	(-0.86)	(-0.89)	(-0.97)
ret_6_1	0.06	0.05	0.05
	(0.24)	(0.20)	(0.19)
ret_12_1	0.30**	0.28*	0.28*
	(2.01)	(1.90)	(1.91)
ret_18_1	-0.26**	-0.26**	-0.27**
	(-2.21)	(-2.16)	(-2.21)
rvol_21d	-8.68	-9.64	-9.34
	(-0.43)	(-0.48)	(-0.46)
rvol_252d	-125.12**	-116.52**	-113.90**
	(-2.15)	(-2.01)	(-1.96)
beta_21d	-0.08	-0.07	-0.08
	(-0.43)	(-0.42)	(-0.46)
beta_252d	0.70**	0.63*	0.63*
	(2.11)	(1.90)	(1.91)
ivol_capm_21d	6.02	6.74	6.67
	(0.31)	(0.35)	(0.35)
ivol_capm_252d	105.79*	98.22*	95.50*
	(1.89)	(1.76)	(1.71)
ami_126d	1.05	1.22	1.24
	(1.01)	(1.18)	(1.21)
Note:			*t>1.645; **t<1.960; ***t>2.576

Table 11: **Mega-cap segment, long-short portfolio.** Cross-sectional (Fama-MacBeth) regression of next-month security returns $r_{i,t+1}$ for the full stock universe on a set of predictive characteristics over the time period February 2003 to December 2021. $\hat{R}^{ens}_{t,CNN}$ is the prediction in (4). CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV and ΔPIV are option characteristics from (Neuhierl et al., 2022). $ret\_$ are momentum- and short-term-reversal-based characteristics; β are market beta characteristics; $ivol\_$ are idiosyncratic volatility characteristics. All these are taken from (Jensen et al., 2022b). T-statistics are reported in parentheses.

	CNN1	CNN4	CNN5
$\hat{R}_{t+1,CNN}$	-3.56	1.30	-6.29
	(-0.25)	(0.17)	(-0.98)
CIV	0.54	0.65	0.60
	(0.75)	(0.89)	(0.83)
PIV	0.11	0.32	0.25
	(0.14)	(0.44)	(0.34)
ΔCIV	0.45	0.45	0.43
	(0.45)	(0.43)	(0.41)
ΔPIV	-0.68	-0.75	-0.63
	(-0.65)	(-0.71)	(-0.60)
IVS_{atm}	-0.43	-0.33	-0.35
	(-0.43)	(-0.34)	(-0.36)
IVS_{otm}	-1.26	-1.14	-1.23
	(-1.54)	(-1.47)	(-1.58)
Skew	-0.53	-1.31	-1.29
	(-0.54)	(-1.31)	(-1.25)
VOV	-0.70	-0.64	-0.71
	(-1.04)	(-0.95)	(-1.06)
ret_1_0	-0.26	-0.32	-0.29
	(-0.34)	(-0.41)	(-0.38)
ret_6_1	0.16	0.17	0.16
	(0.42)	(0.44)	(0.42)
ret_12_1	0.35	0.35	0.35
	(1.11)	(1.09)	(1.11)
ret_18_1	0.05	0.03	0.03
	(0.23)	(0.16)	(0.14)
rvol_21d	-91.26**	-102.38**	-97.26**
	(-2.19)	(-2.45)	(-2.34)
rvol_252d	-31.30	-15.06	-22.78
	(-0.29)	(-0.14)	(-0.21)
beta_21d	0.58	0.68*	0.67*
	(1.57)	(1.81)	(1.79)
beta_252d	-0.48	-0.63	-0.56
	(-0.54)	(-0.71)	(-0.64)
ivol_capm_21d	80.89**	89.25**	84.49**
	(2.13)	(2.36)	(2.24)
ivol_capm_252d	17.16	1.28	8.71
	(0.18)	(0.01)	(0.09)
ami_126d	59.06	63.12	72.27
	(0.62)	(0.65)	(0.75)
Note:			*t>1.645; **t<1.960; ***t>2.576

Table 12: **Large-cap segment, long-short portfolio.** Cross-sectional (Fama-MacBeth) regression of next-month security returns $r_{i,t+1}$ for the full stock universe on a set of predictive characteristics over the time period February 2003 to December 2021. $\hat{R}^{ens}_{t,CNN}$ is the prediction in (4). CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV and ΔPIV are option characteristics from (Neuhierl et al., 2022). *ret_* are momentum- and short-term-reversal-based characteristics; β are market beta characteristics; *ivol_* are idiosyncratic volatility characteristics. All these are taken from (Jensen et al., 2022b). T-statistics are reported in parentheses.

	CNN1	CNN4	CNN5
$\hat{R}_{t+1,CNN}$	5.82	-1.05	-0.50
	(0.67)	(-0.28)	(-0.15)
CIV	0.43	0.33	0.37
	(1.02)	(0.79)	(0.88)
PIV	-0.28	-0.38	-0.33
	(-0.67)	(-0.91)	(-0.79)
ΔCIV	0.02	-0.06	-0.08
	(0.03)	(-0.11)	(-0.14)
ΔPIV	0.25	0.28	0.27
	(0.46)	(0.50)	(0.50)
IVS_{atm}	-0.70	-0.71	-0.69
	(-1.27)	(-1.25)	(-1.23)
IVS_{otm}	0.30	0.20	0.15
	(0.64)	(0.42)	(0.33)
Skew	-0.95**	-1.03**	-1.03**
	(-2.01)	(-2.08)	(-2.08)
VOV	-0.97**	-0.93*	-0.93*
	(-2.00)	(-1.92)	(-1.92)
ret_1_0	-0.69	-0.65	-0.63
	(-0.95)	(-0.91)	(-0.87)
ret_6_1	0.25	0.24	0.25
	(0.81)	(0.77)	(0.79)
ret_12_1	0.25	0.25	0.25
	(1.26)	(1.24)	(1.23)
ret_18_1	-0.24	-0.24	-0.24
	(-1.53)	(-1.50)	(-1.48)
rvol_21d	2.96	3.39	4.67
	(0.10)	(0.11)	(0.16)
rvol_252d	38.71	34.80	36.87
	(0.37)	(0.33)	(0.35)
beta_21d	-0.26	-0.26	-0.26
	(-1.06)	(-1.04)	(-1.08)
beta_252d	0.13	0.16	0.15
	(0.21)	(0.25)	(0.24)
ivol_capm_21d	2.28	1.63	0.29
	(0.08)	(0.06)	(0.01)
ivol_capm_252d	-59.77	-53.94	-56.82
	(-0.62)	(-0.57)	(-0.60)
ami_126d	4.24	0.60	0.05
	(0.25)	(0.04)	(0.00)
Note:			*t>1.645; **t<1.960; ***t>2.576

Table 13: **Small-cap segment, long-short portfolio.** Cross-sectional (Fama-MacBeth) regression of next-month security returns $r_{i,t+1}$ for the full stock universe on a set of predictive characteristics over the time period February 2003 to December 2021. $\hat{R}^{ens}_{t,CNN}$ is the prediction in (4). CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV and ΔPIV are option characteristics from (Neuhierl et al., 2022). $ret\_$ are momentum- and short-term-reversal-based characteristics; β are market beta characteristics; $ivol\_$ are idiosyncratic volatility characteristics. All these are taken from (Jensen et al., 2022b). T-statistics are reported in parentheses.

	CNN1	CNN4	CNN5
$\hat{R}_{t+1,CNN}$	19.85***	10.43***	6.88**
	(2.86)	(3.71)	(2.47)
CIV	0.85**	0.74*	0.77*
	(2.02)	(1.78)	(1.86)
PIV	-0.78**	-0.74*	-0.79**
	(-1.98)	(-1.92)	(-2.03)
ΔCIV	-0.45	-0.61	-0.65
	(-0.86)	(-1.18)	(-1.26)
ΔPIV	0.29	0.35	0.38
	(0.56)	(0.69)	(0.74)
IVS_{atm}	-1.62***	-1.48***	-1.57***
	(-3.70)	(-3.36)	(-3.53)
IVS_{otm}	-0.12	-0.38	-0.39
	(-0.29)	(-0.97)	(-0.97)
Skew	0.30	0.11	0.09
	(0.59)	(0.22)	(0.18)
VOV	-0.98**	-0.94**	-0.94**
	(-2.09)	(-2.10)	(-2.08)
ret_1_0	-0.61	-0.61	-0.65
	(-0.98)	(-0.98)	(-1.04)
ret_6_1	0.14	0.12	0.12
	(0.50)	(0.43)	(0.43)
ret_12_1	0.17	0.17	0.18
	(1.01)	(1.01)	(1.07)
ret_18_1	-0.23*	-0.22*	-0.23*
	(-1.75)	(-1.65)	(-1.75)
rvol_21d	-2.11	-2.74	-2.36
	(-0.07)	(-0.10)	(-0.08)
rvol_252d	-143.25	-133.55	-140.45
	(-1.22)	(-1.15)	(-1.20)
beta_21d	-0.27	-0.26	-0.26
	(-0.97)	(-0.93)	(-0.94)
beta_252d	0.63	0.55	0.58
	(1.22)	(1.06)	(1.13)
ivol_capm_21d	4.33	4.12	4.33
	(0.16)	(0.15)	(0.16)
ivol_capm_252d	121.52	113.08	119.74
	(1.10)	(1.03)	(1.08)
ami_126d	1.10	0.64	1.02
	(0.25)	(0.15)	(0.23)

Note: *t>1.645; **t<1.960; ***t>2.576

Table 14: **Micro-cap segment, long-short portfolio.** Cross-sectional (Fama-MacBeth) regression of next-month security returns $r_{i,t+1}$ for the full stock universe on a set of predictive characteristics over the time period February 2003 to December 2021. $\hat{R}^{ens}_{t,CNN}$ is the prediction in (4). CIV, PIV, IVS$_{atm}$, IVS$_{otm}$, Skew, VOV, ΔCIV and ΔPIV are option characteristics from (Neuhierl et al., 2022). $ret\_$ are momentum- and short-term-reversal-based characteristics; β are market beta characteristics; $ivol\_$ are idiosyncratic volatility characteristics. All these are taken from (Jensen et al., 2022b). T-statistics are reported in parentheses.

	CNN1	CNN4	CNN5
$\hat{R}_{t+1,CNN}$	16.11*	6.71	2.11
	(1.94)	(1.26)	(0.37)
CIV	-0.49	-0.48	-0.35
	(-0.83)	(-0.83)	(-0.56)
PIV	-0.79	-0.65	-0.70
	(-1.42)	(-1.14)	(-1.20)
ΔCIV	0.36	0.35	0.17
	(0.46)	(0.44)	(0.22)
ΔPIV	0.23	0.12	0.24
	(0.29)	(0.15)	(0.30)
IVS$_{atm}$	-0.29	-0.17	-0.35
	(-0.32)	(-0.19)	(-0.37)
IVS$_{otm}$	-0.58	-1.05	-1.02
	(-0.58)	(-1.13)	(-1.13)
Skew	-1.32	-1.63**	-1.68**
	(-1.58)	(-2.12)	(-2.24)
VOV	-3.94***	-4.43***	-4.50***
	(-2.77)	(-3.07)	(-3.07)
ret_1_0	-0.25	-0.26	-0.38
	(-0.32)	(-0.34)	(-0.49)
ret_6_1	-0.31	-0.30	-0.33
	(-0.71)	(-0.69)	(-0.76)
ret_12_1	0.33	0.24	0.25
	(0.94)	(0.70)	(0.71)
ret_18_1	-0.21	-0.16	-0.15
	(-0.76)	(-0.55)	(-0.51)
rvol_21d	-98.64	-91.46	-93.97
	(-1.60)	(-1.53)	(-1.59)
rvol_252d	197.02	241.10	261.30
	(0.60)	(0.74)	(0.80)
beta_21d	0.08	0.09	0.08
	(0.30)	(0.34)	(0.29)
beta_252d	-0.40	-0.63	-0.61
	(-0.40)	(-0.63)	(-0.62)
ivol_capm_21d	83.45	75.97	78.99
	(1.40)	(1.30)	(1.37)
ivol_capm_252d	-193.20	-236.00	-256.20
	(-0.60)	(-0.74)	(-0.80)
ami_126d	0.57	0.34	0.36
	(0.42)	(0.26)	(0.27)
Note:			*t>1.645; **t<1.960; ***t>2.576

Table 15: **Not-micro-cap segment, long-short portfolio.** Cross-sectional regression of next-month security returns $r_{i,t+1}$ for the full stock universe on a set of predictive characteristics over the time period February 2003 to December 2021. $\hat{R}^{ens}_{t,CNN}$ is the prediction in (4). CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV and ΔPIV are option characteristics from (Neuhierl et al., 2022). $ret\_$ are momentum- and short-term-reversal-based characteristics; β are market beta characteristics; $ivol\_$ are idiosyncratic volatility characteristics. All these are taken from (Jensen et al., 2022b). T-statistics are reported in parentheses.

	CNN1	CNN4	CNN5
$\hat{R}_{t+1,CNN}$	18.18***	7.50***	4.60**
	(3.38)	(3.15)	(2.17)
CIV	0.61*	0.53*	0.56*
	(1.93)	(1.69)	(1.77)
PIV	-0.50*	-0.49	-0.53*
	(-1.66)	(-1.59)	(-1.71)
ΔCIV	-0.23	-0.33	-0.35
	(-0.58)	(-0.83)	(-0.89)
ΔPIV	0.19	0.18	0.21
	(0.46)	(0.45)	(0.52)
IVS_{atm}	-1.12***	-1.02***	-1.08***
	(-3.58)	(-3.23)	(-3.43)
IVS_{otm}	0.15	-0.11	-0.13
	(0.44)	(-0.34)	(-0.39)
Skew	-0.36	-0.53	-0.55
	(-0.99)	(-1.48)	(-1.53)
VOV	-0.83**	-0.84**	-0.85**
	(-2.20)	(-2.24)	(-2.25)
ret_1_0	-0.66	-0.70	-0.70
	(-1.08)	(-1.14)	(-1.15)
ret_6_1	0.14	0.12	0.13
	(0.54)	(0.46)	(0.48)
ret_12_1	0.27*	0.27*	0.27*
	(1.73)	(1.71)	(1.74)
ret_18_1	-0.23*	-0.23*	-0.23*
	(-1.89)	(-1.85)	(-1.88)
rvol_21d	-4.91	-5.17	-4.94
	(-0.24)	(-0.25)	(-0.24)
rvol_252d	-101.99	-91.69	-93.77
	(-1.44)	(-1.30)	(-1.32)
beta_21d	-0.08	-0.09	-0.09
	(-0.40)	(-0.42)	(-0.41)
beta_252d	0.48	0.42	0.43
	(1.10)	(0.97)	(1.00)
ivol_capm_21d	7.84	7.85	7.73
	(0.40)	(0.40)	(0.39)
ivol_capm_252d	79.41	70.26	71.99
	(1.20)	(1.07)	(1.09)
ami_126d	2.58	2.79	3.06
	(0.70)	(0.76)	(0.83)

Note: *t>1.645; **t<1.960; ***t>2.576

C.1 Long-only Portfolio Performance

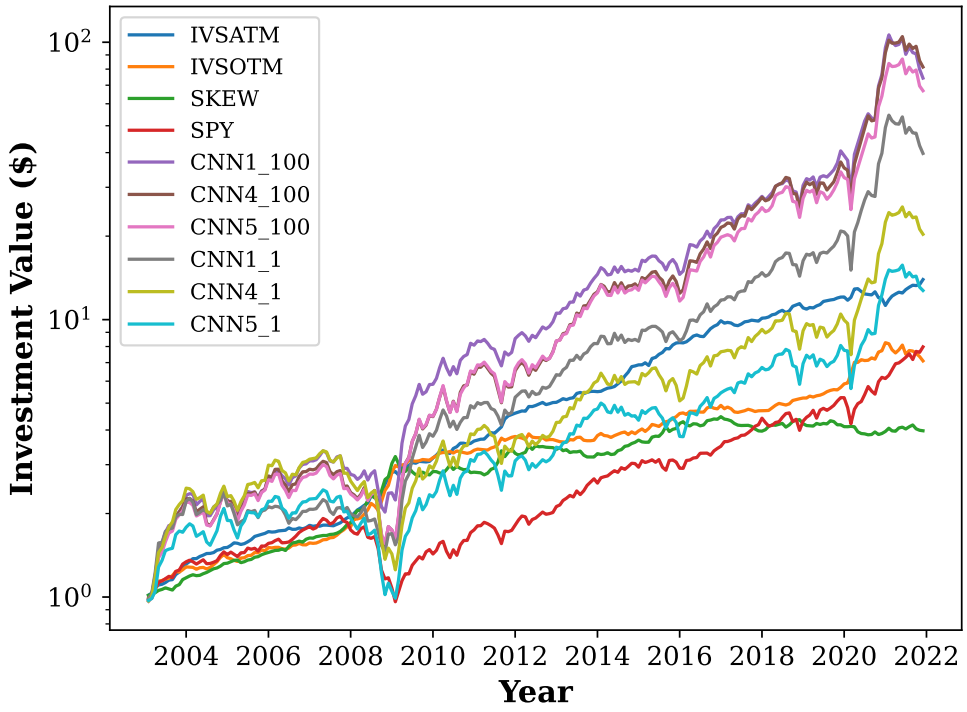


Figure 15: **Long-only strategy (7).** Cumulative Returns of (7) for the full universe of stocks. CNNi_K refers to the ensemble of K CNNi models. IVSATM, IVSOTM, and SKEW are the top three option characteristics-based factors from (Neuhierl et al., 2022), and SPY is the SP&500 ETF.

Table 16: **All stocks, long-only portfolio (7).** Monthly OLS regression of the CNN1, CNN4, and CNN5 long-only portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$
Intercept	0.004***	0.005***	0.005***
	(0.002)	(0.001)	(0.001)
$r_M - r_f$	0.835***	0.944***	0.930***
	(0.049)	(0.041)	(0.042)
SMB	0.248***	0.422***	0.473***
	(0.079)	(0.066)	(0.068)
HML	0.048	0.016	-0.014
	(0.067)	(0.056)	(0.058)
2-12 Momentum	-0.167***	-0.127***	-0.145***
	(0.037)	(0.031)	(0.032)
ST Reversal	0.048	0.015	0.008
	(0.053)	(0.044)	(0.046)
LT Reversal	0.099	0.130**	0.176***
	(0.068)	(0.057)	(0.059)
CIV	0.655***	0.568***	0.467***
	(0.183)	(0.153)	(0.158)
PIV	-0.105	-0.026	0.016
	(0.187)	(0.157)	(0.161)
IVS_{atm}	-0.299***	-0.157**	-0.050
	(0.093)	(0.078)	(0.080)
IVS_{otm}	0.401***	0.171*	0.142
	(0.105)	(0.088)	(0.091)
Skew	0.458***	0.279***	0.221**
	(0.099)	(0.083)	(0.085)
VOV	0.115*	0.125**	0.141**
	(0.066)	(0.056)	(0.057)
Δ CIV	0.035	0.112	0.029
	(0.094)	(0.079)	(0.081)
Δ PIV	-0.007	0.036	-0.022
	(0.085)	(0.072)	(0.074)
Observations	227	227	227
R^2	0.946	0.967	0.963
Adjusted R^2	0.943	0.965	0.961
Residual Std. Error	0.018	0.015	0.015
F Statistic	267.669***	444.960***	395.497***
Note:			*p<0.1; **p<0.05; ***p<0.01

C.2 The Impact of Costs

Table 17: Annualized Sharpe Ratio calculated for equal-weighted portfolios constructed using our 100-Ensemble models (CNNi and NN1) and Ridge Regression predictions. We consider both transaction and short-sale fees of 0, 20, and 40 bps for the micro-cap group, and fees of 0, 10, and 20 bps for all other groups.

		SR_{full}	SR_{mega}	SR_{large}	SR_{small}	SR_{micro}	SR_{not_micro}
Base Fee: 0bps							
	Model						
	CNN1	1.57	0.06	0.47	0.86	1.92	0.94
	CNN4	2.66	0.03	0.47	1.45	2.22	1.45
	CNN5	2.44	-0.05	0.36	1.34	2.14	1.29
	$\text{Ridge}_{z=0.1}$	1.70	0.43	0.59	1.09	1.93	0.98
	NN1	2.04	0.31	0.63	1.30	2.53	1.22
Base Fee: 10bps							
	Model						
	CNN1	1.20	-0.26	0.11	0.54	1.67	0.54
	CNN4	2.05	-0.42	-0.08	1.03	1.95	0.84
	CNN5	1.80	-0.51	-0.16	0.89	1.86	0.65
	$\text{Ridge}_{z=0.1}$	1.30	0.09	0.23	0.74	1.56	0.61
	NN1	1.62	-0.03	0.23	0.94	2.21	0.79
Base Fee: 20bps							
	Model						
	CNN1	0.83	-0.59	-0.26	0.21	1.41	0.13
	CNN4	1.43	-0.87	-0.62	0.61	1.67	0.23
	CNN5	1.16	-0.98	-0.69	0.43	1.58	0.00
	$\text{Ridge}_{z=0.1}$	0.90	-0.24	-0.13	0.39	1.19	0.23
	NN1	1.19	-0.38	-0.16	0.58	1.89	0.36

D Additional Results

D.1 Ridge Regression Results

Table 18: **All stocks, ridge long-short portfolio** Monthly OLS regression of the Ridge portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. z is the penalty term. Sharpe Ratio Full Sample: 1.70.

	$z = 10^{-5}$	$z = 10^{-3}$	$z = 10^{-1}$	$z = 10^{0}$	$z = 10^{1}$
Intercept	0.006***	0.006***	0.006***	0.006***	0.006***
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.066	0.066	0.065	0.059	0.059
	(0.051)	(0.051)	(0.051)	(0.051)	(0.050)
SMB	-0.028	-0.028	-0.028	-0.026	-0.042
	(0.083)	(0.083)	(0.083)	(0.082)	(0.081)
HML	-0.139*	-0.139**	-0.141**	-0.140**	-0.158**
	(0.071)	(0.071)	(0.070)	(0.070)	(0.069)
2-12 Momentum	0.109***	0.109***	0.108***	0.107***	0.117***
	(0.038)	(0.038)	(0.038)	(0.038)	(0.038)
ST Reversal	0.013	0.013	0.016	0.022	0.029
	(0.055)	(0.055)	(0.055)	(0.055)	(0.054)
LT Reversal	0.223***	0.223***	0.226***	0.223***	0.233***
	(0.072)	(0.072)	(0.072)	(0.071)	(0.070)
CIV	0.473**	0.473**	0.475**	0.500***	0.518***
	(0.192)	(0.192)	(0.191)	(0.190)	(0.188)
PIV	-0.169	-0.169	-0.171	-0.193	-0.190
	(0.195)	(0.195)	(0.195)	(0.194)	(0.192)
IVS_{atm}	-0.010	-0.010	-0.009	-0.014	-0.012
	(0.097)	(0.097)	(0.097)	(0.097)	(0.096)
IVS_{otm}	0.412***	0.412***	0.408***	0.405***	0.353***
	(0.110)	(0.110)	(0.110)	(0.110)	(0.108)
Skew	0.549***	0.549***	0.554***	0.555***	0.584***
	(0.104)	(0.104)	(0.103)	(0.103)	(0.102)
VOV	0.152**	0.152**	0.153**	0.155**	0.137**
	(0.069)	(0.069)	(0.069)	(0.069)	(0.068)
Δ CIV	-0.055	-0.055	-0.060	-0.066	-0.070
	(0.098)	(0.098)	(0.098)	(0.098)	(0.096)
Δ PIV	0.145	0.145	0.142	0.149*	0.158*
	(0.089)	(0.089)	(0.089)	(0.089)	(0.088)
Observations	227	227	227	227	227
R^2	0.734	0.734	0.735	0.738	0.743
Adjusted R^2	0.717	0.717	0.718	0.721	0.726
Residual Std. Error	0.019	0.019	0.019	0.019	0.018
F Statistic	41.861***	41.874***	42.093***	42.653***	43.801***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 19: **Mega-cap segment, ridge long-short portfolio** Monthly OLS regression of the Ridge portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. z is the penalty term. Sharpe Ratio Mega Sample: 0.42.

	$z = 10^{-5}$	$z = 10^{-3}$	$z = 10^{-1}$	$z = 10^{0}$	$z = 10^{1}$
Intercept	-0.001	-0.001	-0.001	-0.001	-0.001
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.173**	0.173**	0.173**	0.178**	0.180**
	(0.073)	(0.073)	(0.073)	(0.073)	(0.073)
SMB	0.055	0.055	0.057	0.060	0.048
	(0.118)	(0.118)	(0.118)	(0.119)	(0.119)
HML	-0.165	-0.165	-0.165	-0.153	-0.178*
	(0.101)	(0.101)	(0.101)	(0.101)	(0.101)
2-12 Momentum	0.190***	0.190***	0.190***	0.188***	0.199***
	(0.055)	(0.055)	(0.055)	(0.055)	(0.055)
ST Reversal	0.004	0.004	0.000	-0.000	-0.003
	(0.079)	(0.079)	(0.079)	(0.079)	(0.079)
LT Reversal	0.214**	0.214**	0.215**	0.214**	0.232**
	(0.102)	(0.102)	(0.102)	(0.103)	(0.103)
CIV	-0.050	-0.050	-0.057	-0.056	-0.025
	(0.273)	(0.273)	(0.273)	(0.275)	(0.275)
PIV	0.070	0.070	0.075	0.067	0.034
	(0.279)	(0.279)	(0.279)	(0.281)	(0.280)
IVS_{atm}	-0.262*	-0.262*	-0.261*	-0.272*	-0.311**
	(0.139)	(0.139)	(0.139)	(0.140)	(0.140)
IVS_{otm}	0.215	0.215	0.214	0.235	0.226
	(0.157)	(0.157)	(0.157)	(0.158)	(0.158)
Skew	0.597***	0.597***	0.593***	0.579***	0.571***
	(0.148)	(0.148)	(0.148)	(0.149)	(0.149)
VOV	0.551***	0.551***	0.550***	0.547***	0.540***
	(0.099)	(0.099)	(0.099)	(0.100)	(0.099)
Δ CIV	-0.077	-0.077	-0.075	-0.071	-0.014
	(0.140)	(0.140)	(0.140)	(0.141)	(0.141)
Δ PIV	0.174	0.174	0.173	0.181	0.229*
	(0.127)	(0.127)	(0.127)	(0.128)	(0.128)
Observations	227	227	227	227	227
R^2	0.444	0.444	0.443	0.439	0.447
Adjusted R^2	0.408	0.408	0.406	0.402	0.410
Residual Std. Error	0.027	0.027	0.027	0.027	0.027
F Statistic	12.106***	12.106***	12.051***	11.847***	12.240***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 20: **Large-cap segment, ridge long-short portfolio** Monthly OLS regression of the Ridge portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. z is the penalty term. Sharpe Ratio Large Sample: 0.60.

	$z = 10^{-5}$	$z = 10^{-3}$	$z = 10^{-1}$	$z = 10^{0}$	$z = 10^{1}$
Intercept	-0.003	-0.003	-0.003	-0.003	-0.002
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.214***	0.214***	0.214***	0.204***	0.188***
	(0.062)	(0.062)	(0.062)	(0.062)	(0.061)
SMB	0.321***	0.321***	0.322***	0.304***	0.327***
	(0.101)	(0.101)	(0.101)	(0.101)	(0.099)
HML	-0.239***	-0.239***	-0.237***	-0.243***	-0.283***
	(0.086)	(0.086)	(0.086)	(0.086)	(0.085)
2-12 Momentum	0.185***	0.185***	0.184***	0.181***	0.174***
	(0.047)	(0.047)	(0.047)	(0.047)	(0.046)
ST Reversal	0.183***	0.183***	0.183***	0.174**	0.191***
	(0.068)	(0.068)	(0.068)	(0.067)	(0.066)
LT Reversal	0.194**	0.194**	0.193**	0.197**	0.247***
	(0.088)	(0.088)	(0.088)	(0.087)	(0.086)
CIV	-0.028	-0.028	-0.035	-0.035	-0.030
	(0.234)	(0.234)	(0.234)	(0.233)	(0.230)
PIV	0.046	0.046	0.052	0.064	0.046
	(0.239)	(0.239)	(0.239)	(0.238)	(0.234)
IVS_{atm}	-0.072	-0.072	-0.074	-0.052	-0.062
	(0.119)	(0.119)	(0.119)	(0.118)	(0.117)
IVS_{otm}	0.281**	0.281**	0.279**	0.261*	0.265**
	(0.135)	(0.135)	(0.135)	(0.134)	(0.132)
Skew	0.587***	0.587***	0.590***	0.595***	0.568***
	(0.127)	(0.127)	(0.126)	(0.126)	(0.124)
VOV	0.262***	0.262***	0.260***	0.275***	0.277***
	(0.085)	(0.085)	(0.085)	(0.084)	(0.083)
Δ CIV	-0.156	-0.156	-0.154	-0.170	-0.194
	(0.120)	(0.120)	(0.120)	(0.119)	(0.118)
Δ PIV	0.258**	0.258**	0.258**	0.234**	0.256**
	(0.109)	(0.109)	(0.109)	(0.109)	(0.107)
Observations	227	227	227	227	227
R^2	0.557	0.557	0.557	0.556	0.570
Adjusted R^2	0.528	0.528	0.527	0.527	0.542
Residual Std. Error	0.023	0.023	0.023	0.023	0.022
F Statistic	19.063***	19.062***	19.005***	18.971***	20.088***
Note:					*p<0.1; **p<0.05; ***p<0.01

Table 21: **Small-cap segment, ridge long-short portfolio** Monthly OLS regression of the Ridge portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. z is the penalty term. Sharpe Ratio Small Sample: 1.09.

	$z = 10^{-5}$	$z = 10^{-3}$	$z = 10^{-1}$	$z = 10^{0}$	$z = 10^{1}$
Intercept	0.000	0.000	0.000	0.000	0.000
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.038	0.038	0.035	0.032	0.045
	(0.067)	(0.067)	(0.067)	(0.067)	(0.067)
SMB	-0.022	-0.022	-0.026	-0.031	-0.036
	(0.109)	(0.109)	(0.109)	(0.109)	(0.109)
HML	0.026	0.026	0.020	0.020	0.032
	(0.093)	(0.093)	(0.093)	(0.093)	(0.092)
2-12 Momentum	0.174***	0.173***	0.173***	0.173***	0.162***
	(0.051)	(0.051)	(0.051)	(0.050)	(0.050)
ST Reversal	-0.015	-0.015	-0.015	-0.016	-0.001
	(0.073)	(0.073)	(0.073)	(0.073)	(0.073)
LT Reversal	0.111	0.111	0.116	0.121	0.099
	(0.095)	(0.095)	(0.095)	(0.094)	(0.094)
CIV	0.121	0.121	0.110	0.107	0.050
	(0.253)	(0.253)	(0.253)	(0.252)	(0.251)
PIV	0.153	0.153	0.166	0.172	0.228
	(0.258)	(0.258)	(0.258)	(0.257)	(0.256)
IVS_{atm}	0.194	0.194	0.199	0.201	0.173
	(0.129)	(0.129)	(0.129)	(0.128)	(0.128)
IVS_{otm}	0.386***	0.385***	0.380***	0.365**	0.376**
	(0.146)	(0.146)	(0.146)	(0.145)	(0.145)
Skew	0.370***	0.370***	0.375***	0.374***	0.408***
	(0.137)	(0.137)	(0.137)	(0.136)	(0.136)
VOV	0.041	0.041	0.044	0.044	0.040
	(0.091)	(0.091)	(0.092)	(0.091)	(0.091)
Δ CIV	-0.027	-0.026	-0.029	-0.021	0.015
	(0.130)	(0.130)	(0.130)	(0.129)	(0.129)
Δ PIV	0.125	0.126	0.123	0.141	0.169
	(0.118)	(0.118)	(0.118)	(0.118)	(0.117)
Observations	227	227	227	227	227
R^2	0.526	0.526	0.526	0.526	0.532
Adjusted R^2	0.494	0.494	0.495	0.495	0.501
Residual Std. Error	0.025	0.025	0.025	0.025	0.024
F Statistic	16.786***	16.785***	16.805***	16.793***	17.197***
Note:					*p<0.1; **p<0.05; ***p<0.01

Table 22: **Micro-cap segment, ridge long-short portfolio** Monthly OLS regression of the Ridge portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. z is the penalty term. Sharpe Ratio Micro Sample: 1.90.

	$z = 10^{-5}$	$z = 10^{-3}$	$z = 10^{-1}$	$z = 10^{0}$	$z = 10^{1}$
Intercept	0.015***	0.015***	0.015***	0.015***	0.018***
	(0.004)	(0.004)	(0.004)	(0.004)	(0.005)
$r_M - r_f$	0.242*	0.242*	0.246*	0.237*	-0.011
	(0.139)	(0.139)	(0.138)	(0.139)	(0.147)
SMB	-0.030	-0.030	-0.031	-0.030	-0.193
	(0.226)	(0.226)	(0.225)	(0.226)	(0.240)
HML	-0.021	-0.021	-0.028	-0.013	-0.001
	(0.192)	(0.192)	(0.192)	(0.192)	(0.205)
2-12 Momentum	-0.149	-0.149	-0.147	-0.139	-0.125
	(0.104)	(0.104)	(0.104)	(0.104)	(0.111)
ST Reversal	0.172	0.172	0.169	0.197	0.107
	(0.151)	(0.151)	(0.151)	(0.151)	(0.161)
LT Reversal	-0.116	-0.116	-0.112	-0.141	0.039
	(0.195)	(0.195)	(0.195)	(0.196)	(0.208)
CIV	1.241**	1.241**	1.252**	1.270**	1.729***
	(0.522)	(0.522)	(0.521)	(0.523)	(0.556)
PIV	-0.972*	-0.972*	-0.979*	-0.991*	-1.390**
	(0.533)	(0.533)	(0.532)	(0.533)	(0.567)
IVS_{atm}	0.284	0.284	0.278	0.268	0.415
	(0.266)	(0.266)	(0.265)	(0.266)	(0.283)
IVS_{otm}	0.320	0.320	0.320	0.303	0.285
	(0.301)	(0.301)	(0.300)	(0.301)	(0.320)
Skew	0.947***	0.947***	0.949***	0.958***	0.815***
	(0.282)	(0.282)	(0.282)	(0.282)	(0.300)
VOV	0.230	0.230	0.222	0.214	0.456**
	(0.189)	(0.189)	(0.189)	(0.189)	(0.201)
Δ CIV	-0.535**	-0.535**	-0.523*	-0.532**	-0.595**
	(0.268)	(0.268)	(0.267)	(0.268)	(0.285)
Δ PIV	-0.422*	-0.422*	-0.418*	-0.417*	-0.295
	(0.244)	(0.244)	(0.243)	(0.244)	(0.259)
Observations	227	227	227	227	227
R^2	0.345	0.345	0.348	0.345	0.333
Adjusted R^2	0.302	0.302	0.304	0.302	0.289
Residual Std. Error	0.051	0.051	0.051	0.051	0.054
F Statistic	7.982***	7.982***	8.067***	7.974***	7.556***
Note:					*p<0.1; **p<0.05; ***p<0.01

Table 23: **Not-micro-cap segment, ridge long-short portfolio** Monthly OLS regression of the Ridge portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. z is the penalty term. Sharpe Ratio Non-Micro Sample: 0.99

	$z = 10^{-5}$	$z = 10^{-3}$	$z = 10^{-1}$	$z = 10^{0}$	$z = 10^{1}$
Intercept	-0.001	-0.001	-0.001	-0.001	-0.001
	(0.002)	(0.002)	(0.002)	(0.002)	(0.001)
$r_M - r_f$	0.118**	0.118**	0.118**	0.120**	0.121**
	(0.049)	(0.049)	(0.049)	(0.049)	(0.048)
SMB	0.151*	0.151*	0.152*	0.152*	0.160**
	(0.080)	(0.080)	(0.080)	(0.080)	(0.078)
HML	-0.137**	-0.137**	-0.139**	-0.140**	-0.150**
	(0.068)	(0.068)	(0.068)	(0.068)	(0.067)
2-12 Momentum	0.146***	0.146***	0.146***	0.145***	0.144***
	(0.037)	(0.037)	(0.037)	(0.037)	(0.036)
ST Reversal	0.053	0.053	0.054	0.055	0.051
	(0.054)	(0.054)	(0.054)	(0.053)	(0.052)
LT Reversal	0.173**	0.173**	0.172**	0.169**	0.179***
	(0.069)	(0.069)	(0.069)	(0.069)	(0.068)
CIV	-0.002	-0.002	-0.001	-0.021	-0.018
	(0.185)	(0.185)	(0.185)	(0.185)	(0.181)
PIV	0.203	0.203	0.203	0.221	0.228
	(0.189)	(0.189)	(0.189)	(0.189)	(0.184)
IVS_{atm}	0.061	0.061	0.060	0.061	0.057
	(0.094)	(0.094)	(0.094)	(0.094)	(0.092)
IVS_{otm}	0.301***	0.301***	0.299***	0.303***	0.285***
	(0.107)	(0.107)	(0.107)	(0.106)	(0.104)
Skew	0.519***	0.519***	0.522***	0.517***	0.533***
	(0.100)	(0.100)	(0.100)	(0.100)	(0.098)
VOV	0.148**	0.148**	0.150**	0.149**	0.142**
	(0.067)	(0.067)	(0.067)	(0.067)	(0.065)
Δ CIV	-0.055	-0.055	-0.057	-0.051	-0.063
	(0.095)	(0.095)	(0.095)	(0.095)	(0.093)
Δ PIV	0.209**	0.209**	0.211**	0.216**	0.212**
	(0.086)	(0.086)	(0.086)	(0.086)	(0.084)
Observations	227	227	227	227	227
R^2	0.681	0.681	0.683	0.683	0.698
Adjusted R^2	0.660	0.660	0.662	0.662	0.678
Residual Std. Error	0.018	0.018	0.018	0.018	0.018
F Statistic	32.393***	32.393***	32.555***	32.660***	34.945***
Note:					*p<0.1; **p<0.05; ***p<0.01

D.2 Comparison: Simple NN against CNN

Table 24: **All stocks, long-short.** Monthly OLS regression of the CNNs portfolios on the factor model that includes the NN, CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. *As we can see, the alphas survive for any CNNi even if the NN factor portfolio is included on the right-hand side.*

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$
Intercept	0.004**	0.009***	0.009***
	(0.001)	(0.002)	(0.002)
$r_M - r_f$	-0.134***	-0.045	-0.096*
	(0.047)	(0.048)	(0.050)
SMB	-0.170**	-0.030	0.028
	(0.076)	(0.078)	(0.081)
HML	-0.035	-0.047	-0.104
	(0.065)	(0.066)	(0.069)
2-12 Momentum	-0.142***	-0.112***	-0.132***
	(0.035)	(0.036)	(0.038)
ST Reversal	-0.065	-0.131**	-0.170***
	(0.051)	(0.052)	(0.054)
LT Reversal	0.053	0.102	0.205***
	(0.066)	(0.068)	(0.071)
CIV	0.242	-0.076	-0.104
	(0.177)	(0.182)	(0.190)
PIV	-0.153	0.082	0.071
	(0.180)	(0.184)	(0.192)
IVS_{atm}	-0.394***	0.061	0.224**
	(0.089)	(0.092)	(0.096)
IVS_{otm}	0.427***	0.002	0.028
	(0.103)	(0.106)	(0.110)
Skew	0.222**	-0.024	-0.079
	(0.100)	(0.102)	(0.107)
VOV	0.089	0.089	0.104
	(0.064)	(0.066)	(0.068)
Δ CIV	0.049	0.283***	0.179*
	(0.090)	(0.092)	(0.096)
Δ PIV	0.039	0.102	0.032
	(0.082)	(0.084)	(0.088)
NN	0.619***	0.438***	0.350***
	(0.049)	(0.051)	(0.053)
Observations	227	227	227
R^2	0.803	0.532	0.456
Adjusted R^2	0.789	0.499	0.418
Residual Std. Error	0.017	0.018	0.018
F Statistic	57.399***	15.985***	11.814***
Note:			*p<0.1; **p<0.05; ***p<0.01

Table 25: **All stocks, long-short.** Monthly OLS regression of the NN portfolio on the factor model that includes the CNN, CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. *As we can see, the NN's alpha does not survive when the CNN1 factor portfolio is included in the right-hand side.*

	$NN1_{ew}$
Intercept	0.001
	(0.002)
$r_M - r_f$	0.172***
	(0.049)
SMB	0.104
	(0.081)
HML	-0.022
	(0.068)
2-12 Momentum	0.139***
	(0.037)
ST Reversal	0.037
	(0.054)
LT Reversal	0.057
	(0.070)
CIV	0.090
	(0.187)
PIV	-0.034
	(0.190)
IVS_{atm}	0.306***
	(0.096)
IVS_{otm}	-0.060
	(0.113)
Skew	0.198*
	(0.105)
VOV	0.004
	(0.068)
Δ CIV	-0.027
	(0.095)
Δ PIV	0.063
	(0.087)
CNN1	0.688***
	(0.055)
Observations	227
R^2	0.755
Adjusted R^2	0.738
Residual Std. Error	0.018
F Statistic	43.409***
Note:	*p<0.1; **p<0.05; ***p<0.01

E Additional Analysis for Different Size Groups of Stocks

Table 26: **All-stocks, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **10 bps** and a short-sale monthly cost of **10 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.010***	0.013***	0.011***	0.010***	0.007***
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	-0.047	0.016	-0.047	0.099*	0.065
	(0.061)	(0.055)	(0.054)	(0.058)	(0.050)
SMB	-0.179*	-0.035	0.024	-0.024	-0.025
	(0.100)	(0.090)	(0.089)	(0.095)	(0.082)
HML	-0.085	-0.080	-0.130*	-0.105	-0.141**
	(0.085)	(0.077)	(0.075)	(0.081)	(0.070)
2-12 Momentum	-0.098**	-0.081*	-0.107***	0.064	0.107***
	(0.046)	(0.042)	(0.041)	(0.044)	(0.038)
ST Reversal	-0.074	-0.135**	-0.173***	-0.039	0.017
	(0.067)	(0.060)	(0.059)	(0.063)	(0.055)
LT Reversal	0.156*	0.173**	0.261***	0.239***	0.227***
	(0.086)	(0.078)	(0.077)	(0.082)	(0.071)
CIV	0.525**	0.120	0.048	0.463**	0.480**
	(0.231)	(0.209)	(0.205)	(0.220)	(0.191)
PIV	-0.314	-0.029	-0.011	-0.266	-0.179
	(0.236)	(0.213)	(0.209)	(0.224)	(0.195)
IVS_{atm}	-0.357***	0.089	0.248**	0.018	-0.010
	(0.118)	(0.106)	(0.105)	(0.112)	(0.097)
IVS_{otm}	0.686***	0.190	0.176	0.432***	0.415***
	(0.133)	(0.120)	(0.118)	(0.127)	(0.110)
Skew	0.594***	0.237**	0.129	0.542***	0.548***
	(0.125)	(0.113)	(0.111)	(0.119)	(0.103)
VOV	0.159*	0.139*	0.144*	0.115	0.153**
	(0.084)	(0.076)	(0.074)	(0.080)	(0.069)
Δ CIV	0.050	0.279**	0.176*	0.067	-0.069
	(0.119)	(0.108)	(0.106)	(0.113)	(0.098)
Δ PIV	0.130	0.162*	0.081	0.160	0.135
	(0.108)	(0.098)	(0.096)	(0.103)	(0.089)
Observations	227	227	227	227	227
R^2	0.657	0.366	0.345	0.620	0.735
Adjusted R^2	0.634	0.324	0.302	0.595	0.718
Residual Std. Error	0.022	0.020	0.020	0.021	0.019
F Statistic	28.965***	8.752***	7.972***	24.756***	42.095***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 27: **Mega-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **10 bps** and a short-sale monthly cost of **10 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.000	-0.002	-0.004*	0.001	0.001
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.065	0.108	0.064	0.149*	0.172**
	(0.073)	(0.073)	(0.071)	(0.076)	(0.072)
SMB	0.069	0.119	0.017	0.022	0.056
	(0.119)	(0.118)	(0.115)	(0.124)	(0.118)
HML	-0.063	-0.007	0.095	-0.016	-0.166*
	(0.101)	(0.101)	(0.098)	(0.106)	(0.100)
2-12 Momentum	0.034	-0.028	-0.015	0.211***	0.190***
	(0.055)	(0.055)	(0.053)	(0.058)	(0.054)
ST Reversal	-0.237***	-0.153*	-0.202***	-0.129	-0.000
	(0.080)	(0.079)	(0.077)	(0.083)	(0.079)
LT Reversal	0.161	0.194*	0.121	0.187*	0.218**
	(0.103)	(0.102)	(0.099)	(0.108)	(0.102)
CIV	-0.444	-0.330	-0.448*	-0.240	-0.050
	(0.276)	(0.274)	(0.267)	(0.289)	(0.273)
PIV	0.206	0.240	0.377	0.206	0.067
	(0.282)	(0.279)	(0.272)	(0.294)	(0.279)
IVS_{atm}	-0.327**	0.026	0.192	-0.214	-0.265*
	(0.141)	(0.140)	(0.136)	(0.147)	(0.139)
IVS_{otm}	0.234	-0.160	0.051	0.041	0.217
	(0.159)	(0.158)	(0.154)	(0.167)	(0.158)
Skew	0.664***	0.305**	0.144	0.768***	0.589***
	(0.149)	(0.148)	(0.144)	(0.156)	(0.148)
VOV	0.125	0.162	0.206**	0.390***	0.548***
	(0.100)	(0.099)	(0.097)	(0.105)	(0.099)
Δ CIV	0.280*	0.245*	0.138	0.133	-0.076
	(0.143)	(0.141)	(0.138)	(0.149)	(0.141)
Δ PIV	0.308**	0.104	-0.082	0.233*	0.170
	(0.129)	(0.128)	(0.125)	(0.135)	(0.128)
Observations	227	227	227	227	227
R^2	0.522	0.099	0.097	0.385	0.442
Adjusted R^2	0.491	0.039	0.037	0.345	0.405
Residual Std. Error	0.027	0.027	0.026	0.028	0.027
F Statistic	16.561***	1.661*	1.628*	9.485***	11.992***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 28: **Large-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **10 bps** and a short-sale monthly cost of **10 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	-0.001	-0.002	-0.002	-0.002	-0.002
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.152**	0.086	0.017	0.202***	0.215***
	(0.063)	(0.058)	(0.061)	(0.064)	(0.062)
SMB	0.117	-0.084	0.008	0.208**	0.324***
	(0.103)	(0.094)	(0.099)	(0.105)	(0.101)
HML	-0.027	-0.119	-0.072	-0.150*	-0.239***
	(0.088)	(0.080)	(0.084)	(0.089)	(0.086)
2-12 Momentum	-0.025	-0.048	-0.062	0.148***	0.182***
	(0.048)	(0.043)	(0.046)	(0.049)	(0.047)
ST Reversal	-0.029	-0.088	-0.157**	0.123*	0.184***
	(0.069)	(0.063)	(0.066)	(0.070)	(0.067)
LT Reversal	0.032	0.197**	0.157*	0.191**	0.192**
	(0.089)	(0.081)	(0.086)	(0.091)	(0.087)
CIV	-0.098	-0.179	-0.375	0.038	-0.028
	(0.239)	(0.218)	(0.230)	(0.243)	(0.234)
PIV	-0.080	0.179	0.368	-0.105	0.041
	(0.244)	(0.222)	(0.235)	(0.248)	(0.238)
IVS_{atm}	-0.449***	0.220**	0.362***	-0.007	-0.081
	(0.122)	(0.111)	(0.117)	(0.124)	(0.119)
IVS_{otm}	0.682***	-0.100	-0.157	0.220	0.284**
	(0.138)	(0.126)	(0.133)	(0.140)	(0.135)
Skew	0.577***	0.428***	0.381***	0.636***	0.585***
	(0.129)	(0.118)	(0.124)	(0.131)	(0.126)
VOV	-0.061	0.195**	0.204**	0.119	0.259***
	(0.087)	(0.079)	(0.083)	(0.088)	(0.085)
Δ CIV	-0.055	0.113	0.123	-0.018	-0.164
	(0.124)	(0.113)	(0.119)	(0.126)	(0.121)
Δ PIV	0.157	-0.149	-0.200*	0.247**	0.253**
	(0.112)	(0.102)	(0.108)	(0.114)	(0.109)
Observations	227	227	227	227	227
R^2	0.533	0.199	0.192	0.452	0.556
Adjusted R^2	0.502	0.146	0.139	0.416	0.527
Residual Std. Error	0.023	0.021	0.022	0.024	0.023
F Statistic	17.267***	3.754***	3.600***	12.503***	18.951***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 29: **Small-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **10 bps** and a short-sale monthly cost of **10 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.002	0.005**	0.003	0.002	0.002
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	-0.003	0.130*	0.013	0.154**	0.038
	(0.077)	(0.079)	(0.071)	(0.072)	(0.068)
SMB	-0.334***	-0.017	0.072	-0.087	-0.034
	(0.125)	(0.128)	(0.116)	(0.118)	(0.110)
HML	0.183*	0.065	-0.008	0.112	0.010
	(0.106)	(0.109)	(0.099)	(0.100)	(0.094)
2-12 Momentum	0.020	0.010	-0.080	0.096*	0.174***
	(0.058)	(0.059)	(0.054)	(0.054)	(0.051)
ST Reversal	0.006	-0.048	-0.106	-0.027	-0.010
	(0.083)	(0.086)	(0.078)	(0.079)	(0.074)
LT Reversal	-0.014	-0.019	0.162	0.015	0.135
	(0.108)	(0.111)	(0.101)	(0.102)	(0.095)
CIV	0.371	0.325	0.155	-0.012	0.146
	(0.289)	(0.297)	(0.270)	(0.273)	(0.255)
PIV	-0.100	-0.274	-0.145	0.181	0.130
	(0.295)	(0.303)	(0.275)	(0.279)	(0.260)
IVS_{atm}	-0.216	0.074	0.422***	0.503***	0.182
	(0.147)	(0.152)	(0.138)	(0.139)	(0.130)
IVS_{otm}	0.518***	0.194	0.086	0.257	0.353**
	(0.167)	(0.171)	(0.156)	(0.158)	(0.147)
Skew	0.537***	0.187	0.128	0.499***	0.393***
	(0.156)	(0.161)	(0.146)	(0.148)	(0.138)
VOV	0.006	0.105	0.103	0.005	0.045
	(0.105)	(0.108)	(0.098)	(0.099)	(0.092)
Δ CIV	-0.042	-0.001	-0.153	-0.057	-0.062
	(0.149)	(0.153)	(0.139)	(0.141)	(0.132)
Δ PIV	0.228*	0.116	-0.067	0.049	0.119
	(0.135)	(0.139)	(0.126)	(0.128)	(0.119)
Observations	227	227	227	227	227
R^2	0.454	0.131	0.185	0.416	0.514
Adjusted R^2	0.418	0.073	0.131	0.377	0.482
Residual Std. Error	0.028	0.029	0.026	0.027	0.025
F Statistic	12.580***	2.280***	3.443***	10.783***	16.034***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 30: **Micro-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **20 bps** and a short-sale monthly cost of **20 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.036***	0.046***	0.042***	0.036***	0.013***
	(0.005)	(0.006)	(0.006)	(0.005)	(0.004)
$r_M - r_f$	-0.517***	-0.401*	-0.254	-0.160	0.245*
	(0.185)	(0.203)	(0.203)	(0.160)	(0.136)
SMB	-0.246	0.049	0.257	-0.325	-0.017
	(0.301)	(0.331)	(0.331)	(0.261)	(0.221)
HML	-0.427*	-0.310	-0.410	-0.079	-0.007
	(0.256)	(0.282)	(0.282)	(0.222)	(0.188)
2-12 Momentum	-0.259*	-0.366**	-0.359**	-0.057	-0.137
	(0.139)	(0.153)	(0.153)	(0.121)	(0.102)
ST Reversal	-0.242	-0.625***	-0.664***	-0.027	0.132
	(0.201)	(0.221)	(0.221)	(0.174)	(0.148)
LT Reversal	0.412	0.402	0.544*	0.063	-0.105
	(0.260)	(0.286)	(0.286)	(0.225)	(0.191)
CIV	1.858***	0.732	0.524	1.615***	1.269**
	(0.697)	(0.769)	(0.767)	(0.605)	(0.513)
PIV	-1.457**	-0.663	-0.713	-1.189*	-0.984*
	(0.711)	(0.783)	(0.782)	(0.617)	(0.523)
IVS_{atm}	-0.321	0.525	0.460	-0.020	0.227
	(0.356)	(0.392)	(0.391)	(0.308)	(0.261)
IVS_{otm}	0.883**	0.415	0.748*	0.563	0.302
	(0.402)	(0.443)	(0.443)	(0.349)	(0.296)
Skew	0.341	-0.072	-0.300	0.467	0.945***
	(0.377)	(0.415)	(0.414)	(0.327)	(0.277)
VOV	0.289	1.020***	0.819***	0.503**	0.152
	(0.253)	(0.278)	(0.278)	(0.219)	(0.186)
Δ CIV	0.267	0.691*	0.749*	-0.138	-0.481*
	(0.360)	(0.397)	(0.396)	(0.312)	(0.265)
Δ PIV	0.155	0.367	0.253	0.068	-0.367
	(0.326)	(0.359)	(0.359)	(0.283)	(0.240)
Observations	227	227	227	227	227
R^2	0.356	0.239	0.215	0.305	0.347
Adjusted R^2	0.314	0.188	0.163	0.259	0.304
Residual Std. Error	0.068	0.075	0.075	0.059	0.050
F Statistic	8.378***	4.748***	4.139***	6.635***	8.057***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 31: **Not-micro-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **10 bps** and a short-sale monthly cost of **10 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.002	0.004**	0.002	0.001	0.001
	(0.002)	(0.002)	(0.001)	(0.002)	(0.001)
$r_M - r_f$	0.066	0.049	0.028	0.151***	0.123**
	(0.057)	(0.051)	(0.048)	(0.053)	(0.049)
SMB	-0.069	0.022	0.036	0.071	0.151*
	(0.093)	(0.082)	(0.078)	(0.087)	(0.080)
HML	0.018	0.031	-0.010	-0.026	-0.144**
	(0.079)	(0.070)	(0.066)	(0.074)	(0.068)
2-12 Momentum	-0.040	-0.000	-0.025	0.123***	0.145***
	(0.043)	(0.038)	(0.036)	(0.040)	(0.037)
ST Reversal	-0.064	-0.002	-0.085	0.015	0.056
	(0.062)	(0.055)	(0.052)	(0.058)	(0.053)
LT Reversal	0.066	0.073	0.149**	0.139*	0.176**
	(0.080)	(0.071)	(0.068)	(0.075)	(0.069)
CIV	0.160	-0.099	-0.132	-0.020	0.014
	(0.215)	(0.191)	(0.181)	(0.201)	(0.186)
PIV	-0.090	0.150	0.151	0.105	0.185
	(0.219)	(0.195)	(0.185)	(0.205)	(0.189)
IVS_{atm}	-0.313***	0.096	0.284***	0.211**	0.056
	(0.110)	(0.097)	(0.092)	(0.102)	(0.095)
IVS_{otm}	0.485***	0.029	0.009	0.291**	0.299***
	(0.124)	(0.110)	(0.104)	(0.116)	(0.107)
Skew	0.587***	0.329***	0.260***	0.486***	0.519***
	(0.116)	(0.103)	(0.098)	(0.108)	(0.100)
VOV	0.014	0.097	0.110*	0.045	0.146**
	(0.078)	(0.069)	(0.066)	(0.073)	(0.067)
Δ CIV	-0.059	0.082	-0.041	0.031	-0.070
	(0.111)	(0.099)	(0.094)	(0.104)	(0.096)
Δ PIV	0.126	0.058	-0.082	0.165*	0.209**
	(0.100)	(0.089)	(0.085)	(0.094)	(0.087)
Observations	227	227	227	227	227
R^2	0.511	0.202	0.216	0.528	0.679
Adjusted R^2	0.479	0.150	0.164	0.497	0.658
Residual Std. Error	0.021	0.019	0.018	0.020	0.018
F Statistic	15.812***	3.838***	4.170***	16.950***	32.048***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 32: **All-stocks, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **20 bps** and a short-sale monthly cost of **20 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.012***	0.013***	0.011***	0.012***	0.009***
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	-0.046	0.016	-0.047	0.098*	0.065
	(0.061)	(0.055)	(0.054)	(0.058)	(0.050)
SMB	-0.174*	-0.031	0.027	-0.020	-0.021
	(0.099)	(0.090)	(0.088)	(0.094)	(0.082)
HML	-0.086	-0.079	-0.127*	-0.105	-0.141**
	(0.084)	(0.076)	(0.075)	(0.080)	(0.070)
2-12 Momentum	-0.098**	-0.082**	-0.107***	0.063	0.106***
	(0.046)	(0.041)	(0.041)	(0.044)	(0.038)
ST Reversal	-0.074	-0.133**	-0.171***	-0.037	0.019
	(0.066)	(0.060)	(0.059)	(0.063)	(0.055)
LT Reversal	0.158*	0.173**	0.260***	0.241***	0.228***
	(0.086)	(0.077)	(0.076)	(0.081)	(0.071)
CIV	0.531**	0.120	0.043	0.471**	0.485**
	(0.231)	(0.208)	(0.205)	(0.219)	(0.190)
PIV	-0.324	-0.033	-0.008	-0.278	-0.187
	(0.235)	(0.212)	(0.208)	(0.223)	(0.194)
IVS_{atm}	-0.359***	0.090	0.250**	0.018	-0.011
	(0.118)	(0.106)	(0.105)	(0.112)	(0.097)
IVS_{otm}	0.693***	0.199*	0.182	0.441***	0.421***
	(0.133)	(0.120)	(0.118)	(0.127)	(0.110)
Skew	0.588***	0.230**	0.125	0.533***	0.542***
	(0.125)	(0.112)	(0.110)	(0.118)	(0.103)
VOV	0.159*	0.140*	0.144*	0.116	0.152**
	(0.084)	(0.075)	(0.074)	(0.079)	(0.069)
Δ CIV	0.045	0.271**	0.171	0.054	-0.078
	(0.120)	(0.108)	(0.106)	(0.114)	(0.099)
Δ PIV	0.124	0.154	0.076	0.147	0.127
	(0.108)	(0.097)	(0.096)	(0.103)	(0.089)
Observations	227	227	227	227	227
R^2	0.656	0.366	0.346	0.621	0.735
Adjusted R^2	0.634	0.324	0.303	0.596	0.718
Residual Std. Error	0.022	0.020	0.020	0.021	0.018
F Statistic	28.926***	8.747***	8.003***	24.782***	42.107***
Note:					*p<0.1; **p<0.05; ***p<0.01

Table 33: **Mega-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **20 bps** and a short-sale monthly cost of **20 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.001	-0.003	-0.005*	0.003	0.002
	(0.003)	(0.003)	(0.003)	(0.003)	(0.003)
$r_M - r_f$	0.065	0.110	0.066	0.149*	0.172**
	(0.073)	(0.072)	(0.070)	(0.076)	(0.072)
SMB	0.070	0.120	0.018	0.023	0.056
	(0.119)	(0.118)	(0.115)	(0.124)	(0.118)
HML	-0.064	-0.004	0.096	-0.017	-0.166*
	(0.101)	(0.100)	(0.097)	(0.106)	(0.100)
2-12 Momentum	0.034	-0.028	-0.015	0.210***	0.189***
	(0.055)	(0.054)	(0.053)	(0.057)	(0.054)
ST Reversal	-0.237***	-0.152*	-0.202***	-0.129	-0.001
	(0.079)	(0.079)	(0.077)	(0.083)	(0.078)
LT Reversal	0.163	0.191*	0.120	0.189*	0.219**
	(0.103)	(0.102)	(0.099)	(0.107)	(0.102)
CIV	-0.443	-0.330	-0.450*	-0.232	-0.043
	(0.276)	(0.274)	(0.267)	(0.289)	(0.273)
PIV	0.203	0.238	0.378	0.196	0.058
	(0.282)	(0.279)	(0.272)	(0.294)	(0.278)
IVS_{atm}	-0.327**	0.022	0.189	-0.219	-0.270*
	(0.141)	(0.140)	(0.136)	(0.148)	(0.140)
IVS_{otm}	0.238	-0.155	0.054	0.046	0.222
	(0.160)	(0.158)	(0.154)	(0.167)	(0.158)
Skew	0.660***	0.300**	0.142	0.762***	0.584***
	(0.149)	(0.148)	(0.144)	(0.156)	(0.148)
VOV	0.122	0.160	0.203**	0.387***	0.546***
	(0.100)	(0.099)	(0.097)	(0.105)	(0.099)
Δ CIV	0.280*	0.243*	0.139	0.132	-0.079
	(0.144)	(0.142)	(0.139)	(0.150)	(0.142)
Δ PIV	0.305**	0.101	-0.082	0.229*	0.167
	(0.129)	(0.128)	(0.125)	(0.135)	(0.128)
Observations	227	227	227	227	227
R^2	0.522	0.098	0.097	0.384	0.441
Adjusted R^2	0.490	0.039	0.037	0.343	0.404
Residual Std. Error	0.027	0.027	0.026	0.028	0.026
F Statistic	16.527***	1.648*	1.620*	9.425***	11.930***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 34: **Large-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **20 bps** and a short-sale monthly cost of **20 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	-0.001	-0.003	-0.003	-0.000	-0.000
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.151**	0.086	0.017	0.201***	0.214***
	(0.063)	(0.057)	(0.061)	(0.064)	(0.062)
SMB	0.118	-0.082	0.008	0.209**	0.323***
	(0.103)	(0.094)	(0.099)	(0.105)	(0.101)
HML	-0.028	-0.119	-0.072	-0.151*	-0.240***
	(0.087)	(0.080)	(0.084)	(0.089)	(0.085)
2-12 Momentum	-0.025	-0.048	-0.061	0.148***	0.182***
	(0.048)	(0.043)	(0.046)	(0.048)	(0.046)
ST Reversal	-0.029	-0.087	-0.155**	0.123*	0.184***
	(0.069)	(0.063)	(0.066)	(0.070)	(0.067)
LT Reversal	0.033	0.196**	0.157*	0.193**	0.194**
	(0.089)	(0.081)	(0.085)	(0.090)	(0.087)
CIV	-0.093	-0.174	-0.371	0.046	-0.023
	(0.239)	(0.218)	(0.230)	(0.243)	(0.234)
PIV	-0.086	0.173	0.363	-0.115	0.035
	(0.244)	(0.222)	(0.234)	(0.248)	(0.238)
IVS_{atm}	-0.450***	0.216*	0.359***	-0.009	-0.082
	(0.122)	(0.111)	(0.118)	(0.124)	(0.120)
IVS_{otm}	0.685***	-0.099	-0.157	0.223	0.286**
	(0.138)	(0.126)	(0.133)	(0.141)	(0.135)
Skew	0.573***	0.427***	0.380***	0.631***	0.580***
	(0.129)	(0.118)	(0.124)	(0.131)	(0.126)
VOV	-0.062	0.194**	0.202**	0.118	0.259***
	(0.087)	(0.079)	(0.083)	(0.088)	(0.085)
Δ CIV	-0.059	0.111	0.121	-0.024	-0.170
	(0.124)	(0.113)	(0.119)	(0.126)	(0.121)
Δ PIV	0.153	-0.151	-0.202*	0.240**	0.248**
	(0.112)	(0.102)	(0.108)	(0.114)	(0.109)
Observations	227	227	227	227	227
R^2	0.532	0.197	0.190	0.451	0.555
Adjusted R^2	0.501	0.144	0.137	0.414	0.525
Residual Std. Error	0.023	0.021	0.022	0.024	0.023
F Statistic	17.195***	3.716***	3.562***	12.419***	18.862***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 35: **Small-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **20 bps** and a short-sale monthly cost of **20 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.004	0.004	0.002	0.004	0.004*
	(0.003)	(0.003)	(0.003)	(0.003)	(0.002)
$r_M - r_f$	-0.002	0.130*	0.013	0.153**	0.037
	(0.076)	(0.078)	(0.071)	(0.072)	(0.067)
SMB	-0.331***	-0.014	0.072	-0.085	-0.033
	(0.124)	(0.128)	(0.116)	(0.118)	(0.110)
HML	0.180*	0.066	-0.008	0.109	0.008
	(0.106)	(0.109)	(0.099)	(0.100)	(0.093)
2-12 Momentum	0.019	0.010	-0.079	0.096*	0.174***
	(0.057)	(0.059)	(0.054)	(0.054)	(0.051)
ST Reversal	0.005	-0.048	-0.105	-0.025	-0.009
	(0.083)	(0.085)	(0.077)	(0.079)	(0.073)
LT Reversal	-0.012	-0.019	0.161	0.018	0.136
	(0.107)	(0.110)	(0.100)	(0.102)	(0.095)
CIV	0.380	0.325	0.155	-0.001	0.153
	(0.289)	(0.297)	(0.270)	(0.274)	(0.255)
PIV	-0.111	-0.276	-0.145	0.167	0.121
	(0.294)	(0.302)	(0.275)	(0.279)	(0.260)
IVS_{atm}	-0.221	0.073	0.422***	0.499***	0.179
	(0.148)	(0.152)	(0.138)	(0.140)	(0.131)
IVS_{otm}	0.520***	0.197	0.085	0.260	0.355**
	(0.167)	(0.171)	(0.156)	(0.158)	(0.148)
Skew	0.534***	0.182	0.126	0.492***	0.388***
	(0.156)	(0.160)	(0.146)	(0.148)	(0.138)
VOV	0.006	0.105	0.102	0.003	0.043
	(0.105)	(0.108)	(0.098)	(0.099)	(0.093)
Δ CIV	-0.041	-0.004	-0.159	-0.063	-0.067
	(0.150)	(0.154)	(0.140)	(0.142)	(0.133)
Δ PIV	0.228*	0.114	-0.070	0.043	0.114
	(0.135)	(0.139)	(0.126)	(0.128)	(0.120)
Observations	227	227	227	227	227
R^2	0.453	0.130	0.184	0.413	0.512
Adjusted R^2	0.417	0.072	0.130	0.374	0.480
Residual Std. Error	0.028	0.029	0.026	0.027	0.025
F Statistic	12.540***	2.257***	3.410***	10.656***	15.890***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 36: **Micro-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **40 bps** and a short-sale monthly cost of **40 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.037***	0.052***	0.047***	0.036***	0.010**
	(0.007)	(0.007)	(0.007)	(0.006)	(0.005)
$r_M - r_f$	-0.514***	-0.399**	-0.253	-0.160	0.244*
	(0.184)	(0.202)	(0.202)	(0.159)	(0.135)
SMB	-0.241	0.053	0.259	-0.321	-0.017
	(0.299)	(0.330)	(0.329)	(0.260)	(0.220)
HML	-0.427*	-0.308	-0.407	-0.081	-0.008
	(0.254)	(0.280)	(0.280)	(0.221)	(0.187)
2-12 Momentum	-0.256*	-0.362**	-0.355**	-0.055	-0.135
	(0.138)	(0.152)	(0.152)	(0.120)	(0.102)
ST Reversal	-0.237	-0.617***	-0.656***	-0.023	0.132
	(0.200)	(0.220)	(0.220)	(0.173)	(0.147)
LT Reversal	0.414	0.403	0.543*	0.067	-0.102
	(0.259)	(0.285)	(0.284)	(0.224)	(0.190)
CIV	1.873***	0.733	0.527	1.629***	1.273**
	(0.696)	(0.766)	(0.765)	(0.603)	(0.511)
PIV	-1.480**	-0.671	-0.721	-1.209*	-0.990*
	(0.709)	(0.781)	(0.779)	(0.615)	(0.521)
IVS_{atm}	-0.321	0.533	0.467	-0.021	0.222
	(0.356)	(0.392)	(0.391)	(0.309)	(0.261)
IVS_{otm}	0.897**	0.435	0.765*	0.575	0.305
	(0.402)	(0.442)	(0.442)	(0.348)	(0.295)
Skew	0.318	-0.096	-0.322	0.448	0.936***
	(0.376)	(0.414)	(0.413)	(0.326)	(0.276)
VOV	0.285	1.017***	0.815***	0.499**	0.148
	(0.252)	(0.278)	(0.277)	(0.219)	(0.185)
Δ CIV	0.239	0.665*	0.721*	-0.165	-0.489*
	(0.362)	(0.398)	(0.397)	(0.314)	(0.266)
Δ PIV	0.129	0.339	0.226	0.043	-0.374
	(0.326)	(0.359)	(0.358)	(0.282)	(0.239)
Observations	227	227	227	227	227
R^2	0.354	0.237	0.213	0.302	0.346
Adjusted R^2	0.311	0.187	0.161	0.256	0.303
Residual Std. Error	0.067	0.074	0.074	0.058	0.050
F Statistic	8.295***	4.708***	4.101***	6.566***	8.020***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 37: **Not-micro-cap segment, long-short portfolio.** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses. We apply a linear fee of **20 bps** and a short-sale monthly cost of **20 bps** to the returns of CNN1, CNN4, CNN5, NN1, Ridge Regression and all option-based portfolios from (Neuhierl et al., 2022).

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.003	0.003*	0.000	0.003*	0.003*
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.067	0.049	0.028	0.151***	0.122**
	(0.057)	(0.050)	(0.048)	(0.053)	(0.049)
SMB	-0.068	0.024	0.037	0.072	0.152*
	(0.092)	(0.082)	(0.078)	(0.086)	(0.080)
HML	0.016	0.031	-0.010	-0.028	-0.145**
	(0.079)	(0.070)	(0.066)	(0.073)	(0.068)
2-12 Momentum	-0.040	0.000	-0.024	0.123***	0.145***
	(0.043)	(0.038)	(0.036)	(0.040)	(0.037)
ST Reversal	-0.065	-0.002	-0.084	0.015	0.057
	(0.062)	(0.055)	(0.052)	(0.058)	(0.053)
LT Reversal	0.068	0.072	0.149**	0.142*	0.177**
	(0.080)	(0.071)	(0.067)	(0.075)	(0.069)
CIV	0.166	-0.095	-0.132	-0.010	0.024
	(0.215)	(0.191)	(0.181)	(0.201)	(0.186)
PIV	-0.098	0.145	0.151	0.092	0.173
	(0.219)	(0.194)	(0.184)	(0.205)	(0.189)
IVS_{atm}	-0.315***	0.093	0.283***	0.209**	0.052
	(0.110)	(0.098)	(0.093)	(0.103)	(0.095)
IVS_{otm}	0.487***	0.031	0.009	0.295**	0.302***
	(0.124)	(0.110)	(0.104)	(0.116)	(0.107)
Skew	0.585***	0.326***	0.259***	0.480***	0.513***
	(0.116)	(0.103)	(0.098)	(0.108)	(0.100)
VOV	0.013	0.096	0.109*	0.044	0.145**
	(0.078)	(0.069)	(0.066)	(0.073)	(0.067)
Δ CIV	-0.058	0.079	-0.043	0.025	-0.075
	(0.112)	(0.099)	(0.094)	(0.104)	(0.097)
Δ PIV	0.125	0.056	-0.084	0.158*	0.204**
	(0.101)	(0.089)	(0.085)	(0.094)	(0.087)
Observations	227	227	227	227	227
R^2	0.510	0.200	0.214	0.526	0.677
Adjusted R^2	0.478	0.147	0.162	0.494	0.656
Residual Std. Error	0.021	0.018	0.018	0.019	0.018
F Statistic	15.762***	3.786***	4.131***	16.788***	31.776***

Note: *p<0.1; **p<0.05; ***p<0.01

Table 38: **Capped-value-weighted long-short portfolio (weights are set in proportion to stock market capitalisation, with market caps winsorized at 80% NYSE percentile as described in (Jensen et al., Forthcoming)).** Monthly OLS regression of the CNN1, CNN4, CNN5, NN1, and Ridge Regression **long-short** portfolios on the factor model that includes the CIV, PIV, IVS_{atm}, IVS_{otm}, Skew, VOV, ΔCIV, and ΔPIV factor **long-short** portfolios from (Neuhierl et al., 2022), along with the standard Fama-French factors. The intercept coefficient is reported in monthly return terms, with corresponding standard errors in parentheses.

	$CNN1_{ew}$	$CNN4_{ew}$	$CNN5_{ew}$	$NN1_{ew}$	$z = 0.1$
Intercept	0.000	0.003*	0.002	0.000	0.001
	(0.002)	(0.002)	(0.002)	(0.002)	(0.002)
$r_M - r_f$	0.079	0.081	0.053	0.112*	0.044
	(0.073)	(0.059)	(0.059)	(0.063)	(0.053)
SMB	-0.110	0.128	0.104	0.061	0.126
	(0.109)	(0.088)	(0.087)	(0.093)	(0.078)
HML	-0.048	-0.012	0.039	-0.003	-0.207***
	(0.101)	(0.082)	(0.081)	(0.087)	(0.073)
2-12 Momentum	-0.117**	-0.119***	-0.076*	0.092*	0.024
	(0.056)	(0.045)	(0.044)	(0.047)	(0.040)
ST Reversal	-0.040	0.044	-0.043	0.016	0.025
	(0.080)	(0.065)	(0.064)	(0.069)	(0.058)
LT Reversal	0.012	0.012	0.045	0.091	0.109
	(0.107)	(0.087)	(0.085)	(0.091)	(0.077)
CIV	0.144	0.039	-0.045	0.109	0.109
	(0.160)	(0.130)	(0.128)	(0.136)	(0.115)
PIV	-0.005	-0.022	-0.001	0.030	0.138
	(0.165)	(0.134)	(0.132)	(0.141)	(0.119)
IVS_{atm}	-0.141	0.165*	0.190*	0.412***	0.200**
	(0.122)	(0.099)	(0.098)	(0.104)	(0.088)
IVS_{otm}	0.771***	0.066	0.091	0.371***	0.410***
	(0.140)	(0.114)	(0.112)	(0.120)	(0.101)
Skew	0.818***	0.364***	0.234**	0.621***	0.596***
	(0.122)	(0.099)	(0.097)	(0.104)	(0.088)
VOV	-0.049	-0.036	0.101	0.010	0.160**
	(0.090)	(0.073)	(0.072)	(0.077)	(0.065)
Δ CIV	-0.096	0.221*	0.137	-0.107	0.027
	(0.147)	(0.120)	(0.118)	(0.126)	(0.106)
Δ PIV	0.019	0.149	0.055	0.056	0.168*
	(0.134)	(0.109)	(0.108)	(0.115)	(0.097)
Observations	227	227	227	227	227
R^2	0.597	0.280	0.174	0.589	0.728
Adjusted R^2	0.571	0.233	0.119	0.562	0.710
Residual Std. Error	0.027	0.022	0.022	0.023	0.020
F Statistic	22.476***	5.897***	3.183***	21.684***	40.452***

Note: *p<0.1; **p<0.05; ***p<0.01

F Proofs

Proof of Lemma 1. For any x_0, define

$$g(y) \;=\; f((I - UU')x_0 + Uy)\,, \tag{23}$$

and note that g is also real analytic. Then, defining $\tilde{x} = (I - UU')x_0 + UU'x$

$$g(U'x) - f(x) \;=\; f(\tilde{x}) \;-\; f(x) \;=\; \int_0^1 \nabla f(t(\tilde{x} - x) + x)'(\tilde{x} - x)dt. \tag{24}$$

We have

$$(I - UU')M_*(I - UU') \;=\; (I - UU')UDU'M_*(I - UU') \;=\; 0 \tag{25}$$

since $(I - UU')U = 0$. Thus, for any vector z, the function $\nabla f(x)'(I - UU')z$ is zero for Lebesgue almost every x because $E[(\nabla f(x)'(I - UU')z)^2] = z'M_*z = 0$ and, hence, the real analytic function

$$G(x_1, x_2) \;=\; \nabla f(x_1)'(I - UU')x_2 \tag{26}$$

is zero for Lebesgue almost every (x_1, x_2). Therefore, for any fixed t, the real analytic function

$$\hat{G}(t, x_0, x) \;=\; \nabla f(t(\tilde{x}-x)+x)'(\tilde{x}-x) \;=\; \nabla f(t(I-UU')(x-x_0)+x)'(I-UU')(x-x_0) \tag{27}$$

is zero for Lebesgue almost every (x_0, x). Hence,

$$g(U'x) - f(x) \;=\; 0 \tag{28}$$

for Lebesgue almost every (x_0, x). Thus, for Lebesgue almost every x_0, we have that $g(U'x) - f(x)$. To prove the last statement, note that

$$E[\nabla f(X)\nabla f(X)'] \;=\; E[M'\nabla g(MX)\nabla g(MX)'M] \;=\; M'E[\nabla g(MX)\nabla g(MX)]M. \tag{29}$$

Since the components of $\nabla g(MX)$ are linearly independent, we have that the term

$$E[\nabla g(MX)\nabla g(MX)'] \in \mathbb{R}^{r\times r}$$

is strictly positive definite, and hence, $\text{rank}(M'E[\nabla g(MX)\nabla g(MX)]M) = r$. Indeed,

$$a'E[\nabla g(MX)\nabla g(MX)']a = E[(\nabla g(MX)'a)^2] > 0 \tag{30}$$

because $\nabla g(MX)'a$ is not identically zero (linear independence) and hence is almost surely non-zero (by real analyticity). Then, for any y in the image of M (which has dimension r), we have that $y'E[\nabla g(MX)\nabla g(MX)]y > 0$, this concludes the proof. □

Proof of Lemma 2. Let $U_p \in \mathbb{R}^{d\times p}$ be the matrix with first p eigenvectors and $U_{-p} \in \mathbb{R}^{d\times(d-p)}$ the matrix with the last $d-p$ eigenvectors. Then, defining $y = U_p'x \in \mathbb{R}^p$ and $z = U_{-p}'x \in \mathbb{R}^{d-p}$, we get $x = U_p y + U_{-p}z$. Then, the conditional multivariate Gaussian $z|y \sim N(\mu(y), \hat{\Sigma}_z)$, where $\hat{\Sigma}_z \in \mathbb{R}^{d\times d}$ satisfies

$$\hat{\Sigma}_z \;=\; \Sigma_z - \Sigma_{zy}\Sigma_y^{-1}\Sigma_{yz} \;\leq\; \lambda_1(\Sigma)I, \tag{31}$$

where we have defined the blocks

$$\Sigma_z \;=\; U_{-p}'\Sigma U_{-p} \in \mathbb{R}^{(d-p)\times(d-p)},\; \Sigma_{zy} = U_{-p}'\Sigma U_p \in \mathbb{R}^{(d-p)\times d},\; \Sigma_y = U_p'\Sigma U_p \in \mathbb{R}^{p\times p}. \tag{32}$$

Let $f_p(y) \;=\; E[f(x)|U_p'x = y]$. Let us fix y and define $F(z) = f(U_{-p}z + U_p y) - f_p(y) : \mathbb{R}^{d-p} \to \mathbb{R}$ (with a fixed y). Then, its gradient $\nabla_z F(z) \in \mathbb{R}^{d-p}$ satisfies (by the chain rule)

$$\nabla_z F(z) \;=\; \nabla_z(f(U_{-p}z + U_p y) - f_p(y) \;=\; U_{-p}'\nabla_x f(x) \tag{33}$$

for $x = U_{-p}z + U_p y$. Then, by the (Brascamp and Lieb, 1976) inequality,

$$\begin{aligned}
&\mathrm{Var}[F(z)|y] \;=\; E[F(z)^2|y] \;\leq\; E[\nabla_z F(z)'\hat{\Sigma}_z \nabla_z F(z)|y] \\
&\leq\; \Lambda_1(\Sigma) E[\|\nabla_z F(z)\|^2|y] \;=\; \Lambda_1(\Sigma) E[\|\nabla_z F(z)\|^2|y] \\
&=\; \lambda_1(\Sigma) E[\|U'_{-p}\nabla_x f(x)\|^2 | U'_p x = y] \\
&=\; \lambda_1(\Sigma) E[(U'_{-p}\nabla_x f(x))' U'_{-p}\nabla_x f(x) | U'_p x = y] \\
&=\; \lambda_1(\Sigma) E[\nabla_x f(x)' U_{-p} U'_{-p}\nabla_x f(x) | U'_p x = y] \\
&=\; \lambda_1(\Sigma)\,\mathrm{tr}\, E[U_{-p}U'_{-p}\nabla_x f(x)\nabla_x f(x)' | U'_p x = y] \\
&=\; \lambda_1(\Sigma)\,\mathrm{tr}(U_{-p}U'_{-p} E[\nabla_x f(x)\nabla_x f(x)' | U'_p x = y])
\end{aligned} \tag{34}$$

and, hence,

$$\begin{aligned}
&E[(f(x) - f_p(U'_p x))^2] \;=\; E[F(z)^2] \;=\; E[E[F(z)^2|y]] \\
&\leq\; \lambda_1(\Sigma) E[\lambda_1(\Sigma)\,\mathrm{tr}(U_{-p}U'_{-p} E[\nabla_x f(x)\nabla_x f(x)' | U'_p x = y])] \\
&=\; \lambda_1(\Sigma)\,\mathrm{tr}(U_{-p}U'_{-p} E[\nabla_x f(x)\nabla_x f(x)']) \\
&=\; \lambda_1(\Sigma)\,\mathrm{tr}(U_{-p}U'_{-p}\bar{M}_*) \;=\; \lambda_1(\Sigma)\Lambda_{-p}(M_*).
\end{aligned} \tag{35}$$

□