Forward-looking $\mathcal{P}$

Maxim Ulrich* Simon Walther† Jonas Rothfuss‡ Fabio Ferreira§

14. August 2019

ABSTRACT

We present a forward-looking estimator for the time-varying physical return distribution with minimal prior assumptions about the shape of the distribution and no exogenous assumptions about the economy or preferences. Our estimator, which is based on a neural network, derives its forecasts from option-implied measures and predicts the conditional mean and volatility of returns such that profitable trading strategies can be derived. In contrast to backward-looking estimators and alternative forward-looking parametric and non-parametric approaches, its distribution forecasts cannot be rejected in statistical tests and it features lower prediction errors and higher conditional log likelihood values than the alternatives.

*Karlsruhe Institute of Technology (KIT)
†Corresponding Author, Karlsruhe Institute of Technology (KIT), Email address: simon.walther@kit.edu
‡ETH Zurich
§Karlsruhe Institute of Technology (KIT)

1 Introduction

$\mathcal{P}$, the conditional physical probability density function, is the key input to every application of Expected Utility Theory, which itself builds the foundation to any modern economic decision making problem.[1] For instance, any valuation problem is solved by finding the expected present value of future pay-offs, where the expectation is taken with regard to $\mathcal{P}$.[2] Analogous, the optimal solution to any consumption and investment problem depends crucially on $\mathcal{P}$.[3] The vast majority of the literature follows parametric or non-parametric backward-looking approaches to identify $\mathcal{P}$.[4] One serious limitation of backward-looking methodologies is their limited informativeness about forward-looking events in the presence of regime shifts or non-stationary data.[5] Another strand of the literature has relied on the seminal work of Ross (2015) and Hansen and Scheinkman (2009) to identify $\mathcal{P}$ with forward-looking option data and parametric assumptions on the representative investor's marginal utility.[6] Despite its theoretical appeal, recent empirical evidence for the S&P 500 concludes that the forward-looking $\mathcal{P}$ density of returns cannot be accurately recovered based on Ross (2015).[7] We suggest a fourth strategy for estimating $\mathcal{P}$, which similar to the important contribution of Linn et al. (2018) is a pure econometric approach that relies only on minimal mathematical continuity and differentiability assumptions and which hence can be considered free of parametric assumptions of the underlying economy.

Our econometric methodology follows the economic idea of Ross (2015) to back out $\mathcal{P}$ from inherently forward-looking option prices. Yet, instead of relying on simplifying assumptions on an investor's marginal utility, we do instead borrow from a well-developed machine learning literature which applies Mixture Density Networks (MDN) to approximate any conditional probability density function to arbitrary precision (Bishop, 1994). This allows us to extract $\mathcal{P}$ from option data with only minimal continuity and differentiability requirements and to remain agnostic about the underlying structure of the economy and preferences. Our method can also be applied to determine which factors are truly useful in identifying $\mathcal{P}$. Our analysis with S&P 500 return and option data also documents that backward-looking information such as past S&P 500 returns, returns of the Fama and French (1993) value and size factor, momentum return and their respective realized volatilities do not add noteworthy information about $\mathcal{P}$ that is not yet part of option-implied return

[1] In this work, our notion of $\mathcal{P}$ does not explicitly differentiate between objective and subjective probabilities. Early seminal work that discusses both concepts are Ramsey (1931), de Finetti (1937), von Neumann and Morgenstern (1947) and Savage (1954).

[2] For early work of this seminal concept see Gordon (1962), Lucas Jr (1978), Mehra and Prescott (1985), Hansen and Singleton (1982), Hansen and Singleton (1983). Classical textbook treatments are Cochrane (2005) and Duffie (2001).

[3] See Merton (1969), Merton (1975) for seminal work in optimal consumption and portfolio planning.

[4] Noteworthy applications of parametric models on historical returns are Rosenberg and Engle (2002), Barone-Adesi et al. (2008), Barone-Adesi and Dall'O (2010). Important non-parametric kernel density estimators on historical returns are Jackwerth (2000), Jackwerth (2004) and Aït-Sahalia and Lo (1998).

[5] Thorough explanations of this concern can be found in Bliss and Panigirtzoglou (2004) and Linn et al. (2018) and Cuesdeanu and Jackwerth (forthcoming).

[6] See Schneider and Trojani (2019), Jensen et al. (2019), Jackwerth and Menner (2018), Borovička et al. (2016), Carr and Yu (2012) and Walden (2017) for recent generalizations of Ross (2015).

[7] See Jackwerth and Menner (2018) and Dillschneider and Maurer (2018).

moments.

Our option-implied MDN approach works as follows: We assume that the conditional probability of next day's S&P 500 return is a weighted sum of Gaussian distributions, which introduces the aforementioned continuity and differentiability conditions on $\mathcal{P}$. As conditioning variables we use the risk-free rate and two sets of factors. One set of factors captures forward-looking option-implied risk-neutral moments of the S&P 500; namely (i) the $SVIX^2$ from Martin (2017), (ii) the risk-neutral skewness and (iii) risk-neutral kurtosis from Bakshi et al. (2003). The second set of factors captures a subset of classical backward-looking return factors such as the return of the value, size and momentum factor and their respective 10-trading day rolling window variances. Each mean and variance of the Gaussian mixture model as well as the weight that each Gaussian density obtains is allowed to be a function of the conditioning variables. It is key that we are agnostic about the type of function that these quantities follow. Instead, we approximate these functions by a feed-forward neural network with one hidden layer of neurons and let the neural network learn from the data, how to best approximate these functions and hence, $\mathcal{P}$. In order to keep this paper focused on core financial concepts, we refer the reader to our two technical reports that explain mathematical details for setting up well-specified MDNs (Rothfuss et al., 2019b) and for explaining step-by-step how to estimate well-specified MDNs (Rothfuss et al., 2019a).

We confront our methodology with end-of-day S&P 500 return and CBOE option data, spanning the period January 2004 to July 2017, to answer five questions. First, do backward-looking factor returns add information about $\mathcal{P}$ that is not yet spanned by forward-looking option-implied information? In order to answer this question we compare the in-sample and out-of-sample log likelihood values of three MDNs which only differ with regard to the set of conditioning variables. One MDN conditions on forward-looking option data and the risk-free rate only. A second one conditions on backward-looking factor returns and their historical volatility. A third MDN conditions on both types of information, forward-looking option data including the risk-free rate and backward looking factor returns and their historical volatilities. When comparing the respective log likelihood values we follow Welch and Goyal (2008) and compare all likelihood values to a baseline case, which we specify to be a non-parametric kernel density estimator (Jackwerth, 2000; Aït-Sahalia and Lo, 1998). Our analysis concludes that a MDN with only forward-looking option-implied information is sufficient to beat the non-parametric kernel density estimator by a margin of roughly 5%, both with in- and out-of-sample data. While adding backward-looking factor returns further increases the in-sample log-likelihood by a relative margin of 0.6%, the out-of-sample relative increase in the log-likelihood is only 0.04%. We hence conclude that the backward-looking factor returns and their historical volatilities do not add noteworthy information about $\mathcal{P}$ and can hence be easily skipped when building a forward-looking $\mathcal{P}$ estimate.

The second question of our paper is to test whether each of the four considered $\mathcal{P}$ estimates could be the data generating process for the realized time series of daily S&P 500 returns. We follow Jackwerth and Menner (2018) and apply a Berkowitz (2001) and Knüppel (2015) test. Based on out-of-sample data, we reject that the non-parametric kernel density estimator and the backward-

looking MDN are proper characterizations of $\mathcal{P}$. We fail to reject the hypothesis for the forward-looking MDN. These tests also show a text book like pattern for over-fitting when working with backward-looking $\mathcal{P}$ estimates. All of the considered backward-looking densities cannot be rejected using in-sample data, yet, they do not generalize well to previously unseen out-of-sample data and are strongly rejected here.

Based on the first two research questions we conclude that the forward-looking $\mathcal{P}$ estimate that combines a MDN with forward-looking option data and the risk-free rate provides an accurate statistical description of daily S&P 500 returns. As a third research question we want to understand whether our forward-looking $\mathcal{P}$ estimate is useful from a financial economic point of view. To assess that we implement two dynamic trading strategies that rely on $\mathcal{P}$ as a signal for trading and compare the resulting Sharpe ratio to a static trading strategy that does not rely on $\mathcal{P}$. One strategy goes long (short) the S&P 500 on days where the implied forward-looking expected return is positive (negative). We compare the Sharpe ratio of this strategy to a buy-and-hold strategy. The second trading strategy invests in delta-neutral straddles and closes the position after one day. We increase the short position on days where the conditional forward-looking variance expectation is falling and we reduce it for days where the variance forecast is rising. We compare the outcome of the straddle timing strategy to a static short straddle strategy. With regard to the first trading strategy, we document an in-sample (out-of-sample) Sharpe ratio of 0.69 (0.74), relative to a 0.39 (0.65) Sharpe ratio of the static buy-and-hold strategy. We document an in-sample (out-of-sample) Sharpe ratio for the dynamic straddle strategy of 4.9 (5.4), relative to a 2.7 (3.2) Sharpe ratio for the static short straddle strategy. Based on the in-sample and out-of-sample Sharpe ratio results, we conclude that the forward-looking $\mathcal{P}$ density carries economically meaningful information about the time-series properties of daily stock returns.

Our fourth research question aims to understand which of the forward-looking option-implied conditioning variables are especially important for the forward-looking $\mathcal{P}$ density. We perform an adjusted Patton and Timmermann (2010) test to assess the impact of a particular option-implied conditioning variable onto $\mathcal{P}$. Our findings highlight that all of the considered option-implied variables are informative about $\mathcal{P}$ at the 1% significance level. From all option-implied quantities, we find $SVIX^2$ to have the strongest effect on the ex-ante mean, variance, skewness and kurtosis of $\mathcal{P}$. Bakshi et al. (2003) option-implied skewness and kurtosis are only important for pinning down the forward-looking $\mathcal{P}$ skewness and kurtosis.

The adjusted Patton and Timmermann (2010) test does not reveal how the option-implied moments feed into $\mathcal{P}$. The fifth research question does therefore aim to learn from the feed-forward neural network. We follow Davison and Hinkley (1997) and construct confidence intervals for the predictive relationship that the neural network based MDN implies. With regard to VIX^2, we identify a close to linear positive predictive relationship to next day's expected return; well inline with the economic model of Martin (2017). The positive linear relationship is statistically significant for days when $SVIX^2$ is particularly large. Also of interest is the observation of a positive relationship between $SVIX^2$ and the expected $\mathcal{P}$ variance. The identified relationship is linear for days where the

annualized $SVIX$ is above 17%. As to the relationship between $SVIX^2$ and forward-looking $\mathcal{P}$ skewness we document that as option-implied variance drops from an annualized value of 17.3% to 12.2%, next day's $\mathcal{P}$ skewness reduces from roughly 0 to -0.62.

Our work is closest related to the young, yet fast growing and influential, literature on estimating $\mathcal{P}$ from option data. Ross (2015) develops an economic technique to recover the physical return density from its risk-neutral counter-part. His key economic restriction is that the representative investor's ratio of marginal utility between two states is transition independent and thus constant over time. The empirical success of the Ross (2015) theorem is mixed. Audrino et al. (2015) show that a trading strategy with signals extracted from Ross (2015) recovered $\mathcal{P}$ moments outperforms trading signals from risk-neutral moments. On the other hand, Jackwerth and Menner (2018) apply a series of statistical tests on the Ross (2015) implied $\mathcal{P}$ density. The authors reject the hypothesis that realized S&P 500 returns are drawn from the Ross (2015) implied $\mathcal{P}$ density. Jackwerth and Menner (2018) identify that a key challenge to the empirical success of Ross (2015) theorem is the difficulty to obtain the required transition state prices from option price data. Findings in Bakshi et al. (2018) on options on 30-year Treasury bond futures do also challenge the adequacy of the Ross (2015) required pricing kernel.

Jensen et al. (2019) generalize the work of Ross (2015). The authors replace the time-homogeneity restriction on the pricing kernel with a weaker time-separability constraint. The authors show that this extension improves the resulting accuracy of the implied $\mathcal{P}$ volatility forecast, yet it still does not pass a Berkowitz (2001) test. In contrast to these important contributions, our paper shows how to transform option-implied information to $\mathcal{P}$ without relying on the economically important, yet, empirically difficult to measure concept of a pricing kernel. Our approach is only data driven and uses a small-scale feed-forward neural network as part of a MDN to uncover $\mathcal{P}$ from a panel of option prices.

Our work is also related to the model-free $\mathcal{P}$ recoveries of Schneider and Trojani (2019) and Linn et al. (2018). Schneider and Trojani (2019) use economically motivated sign restrictions on tradable higher moment risk premiums to derive constraints on the physical conditional moments of returns. Their recovered $\mathcal{P}$ estimate is free of technical assumptions on the underlying economy and shown to predict S&P 500 returns. The model-free approach of Linn et al. (2018) is an innovative econometric approach that estimates $\mathcal{P}$ using the forward-looking option-implied density and the inverse of the Radon-Nikodym derivative. The authors' approach relies on a finite order cubic B-spline to approximate the inverse of the Radon-Nikodym derivative with a set of time-varying option-implied densities and return realizations that are sampled from the corresponding $\mathcal{P}$ density. Their model-free estimate for $\mathcal{P}$ shares the same information set as our option-implied forward-looking density and similar to Linn et al. (2018), our approach of recovering $\mathcal{P}$ is a pure econometric approach. Different to the previous two contributions however, we do not identify $\mathcal{P}$ based on the conceptually important, though rather indirect route via a pricing kernel. Instead, we directly approximate the unknown $\mathcal{P}$ distribution by a small-scale MDN with a feed-forward neural network that conditions on option-implied information.

Our work also adds to the growing literature that uses machine learning and neural network techniques in finance applications. Early work in this field includes Hutchinson et al. (1994) and Yao et al. (2000) who use neural networks to price options on the S&P 500 and Nikkei 225 futures. More recently, Ludwig (2015) documents several advantages when using a neural network to interpolate the option-implied volatility surface. Dunis et al. (2011) and Zhao et al. (2018) show how to use neural networks for portfolio formation and trading strategies. The influential study of Gu et al. (2019) compares a range of machine learning techniques for time series and cross-sectional return predictions. The authors conclude that well-performing machine learning techniques, such as neural networks, benefit from the ability to capture important data non-linearities. We add to this literature by showing that machine learning techniques are not only useful for return predictions, but they also help improve our understanding on deep financial economic questions, such as how to use the in real-time available rich cross-section of option data to learn about the full conditional return density under $\mathcal{P}$.

The concept of using MDNs and neural networks to approximate conditional probability density functions has been primarily developed in the computer science literature. The seminal work in that field is Bishop (1994). Recently, there has been a new interest in that literature to further improve on that technique. Rothfuss et al. (2019a) develop a noise regularization to machine learning tools like a MDN to prevent over-fitting in applications that have to rely on small training samples. In our implementation of the MDN we test for over-fitting using their regularization technique. Rothfuss et al. (2019b) use a controlled simulation study to assess how different non-parametric conditional density estimators perform when asked to learn a specific density function. The authors conclude that a MDN dominates the other considered non-parametric density estimators. On a technical level, we differ from these studies by letting the training algorithm choose the size of the neural network, which allows the data to determine the proper degree of complexity and which renders noise regularization unnecessary in our case. On an economic level, we differ from these studies by using this well developed tool from the computer science literature to address one of the most fundamental questions in financial economics, namely how to find a good forward-looking estimate for the conditional density of daily S&P 500 returns.

The rest of the paper is structured as follows. Section 2 provides an overview of the MDN and our estimation procedure. We present our data sets in section 3. Our main results are documented in section 4, followed by several robustness checks in section 5. Section 6 concludes.

2 Model-free Conditional Physical Density Estimation

In this section, we present our conditional density estimator that is able (but not restricted) to form physical density estimates based on forward-looking information alone. We consider this estimator to be model-free in the sense that we do not restrict in any way the shape of the distribution or the relationship between the distribution's shape and the inputs that it is conditioned on. Instead, the estimator learns these characteristics during the training phase from the data. The estimator,

called Mixture Density Network (MDN), combines a neural network with the class of mixture density models and has been developed by Bishop (1994).

Let $r_t = log\frac{S_t}{S_{t-1}}$ be the log return of an asset's price at time t and $X_{t-1} = (x_{1,t-1}, ..., x_{i,t-1}, ..., x_{N,t-1})^\intercal \in \mathcal{I} \subseteq \mathbb{R}^N$ be a set of N predictor variables, that we can observe in $t-1$. We start by expressing the conditional physical log return density $p(r_t|X_{t-1})$ as a mixture density model, more precisely, by a Gaussian mixture model

$$p(r_t|X_{t-1}) = \sum_{k=1}^{K} \alpha_k(X_{t-1})\, \mathcal{N}(r_t|\mu_k(X_{t-1}), \sigma_k^2(X_{t-1})). \tag{1}$$

Here, the conditional density is constructed as the weighted sum of K Gaussian densities with respective weights $\alpha_k(X_{t-1})$, means $\mu_k(X_{t-1})$ and variances $\sigma_k^2(X_{t-1})$. The weights, means and variances of the mixture model are an unknown function of the input variables X_{t-1}. In order to form a valid density, it must hold $\sum_{k=1}^{K} \alpha_k(X_{t-1}) = 1$. As Bishop (2006) emphasizes, this specification is flexible enough to approximate almost any distribution to arbitrary precision, provided the number of Gaussians K is large enough.

A MDN uses a feed-forward neural network to jointly estimate $\alpha_k(X_{t-1})$, $\mu_k(X_{t-1})$ and $\sigma_k^2(X_{t-1})$. For robustness, we restrict ourselves to neural networks with one hidden layer of neurons. To formalize that, we express the distribution's parameters as a function of the input variables that we condition on: $f : \mathcal{I} \to \mathbb{R}_+^K \times \mathbb{R}^K \times \mathbb{R}_+^K$, $f(X_{t-1}) = (\alpha_1(X_{t-1}), ..., \alpha_K(X_{t-1}), \mu_1(X_{t-1}), ..., \mu_K(X_{t-1}), \sigma_1^2(X_{t-1}), ..., \sigma_K^2(X_{t-1}))^\intercal$ and write the j-th element of f as

$$f_j(X_{t-1}) = \sum_{h=1}^{H} w_{2,j,h}\, \phi \quad \left(\sum_{i=1}^{N} w_{1,i,h}\, x_{i,t-1} \right) + w_{2,j,bias}. \tag{2}$$

Here, we assume that the input vector X_{t-1} already contains a constant element. In essence, the neural network, which is characterized by equation 2, can be seen as a weighted sum of homogeneous functionals of the input variables. The difference between the functionals only lies in the weights that are assigned to each input. Note that these first-level weights $w_{1,i,h}$ are shared among all elements of f, while the second-level weights $w_{2,j,h}$ and $w_{2,j,bias}$ are element-specific. The function $\phi(\cdot)$, called *activation function*, should be continuous, bounded and non-constant and we follow common practice in the literature in choosing the tangens hyperbolicus function, i.e. $\phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$. As Hornik (1991) showed, this neural network specification is flexible enough to approximate any continuous function $f_j(X_{t-1})$ to arbitrary precision, provided that the number of hidden neurons H is large enough. Put differently, the neural network's parameters $w_{1,i,h}$, $w_{2,j,h}$ and $w_{2,j,bias}$ define the function $f_j(X_{t-1})$, which is selected from the full space of continuous functions in the model training phase. Beside the number of hidden neurons H, which we let our training algorithm choose freely, we impose no restrictions on $f_j(X_{t-1})$ in any way. In combination with the flexible density definition in equation 1, the MDN can approximate any conditional distribution $p(r_t|X_{t-1})$ without prior restrictions about this distribution (Bishop, 1994).

2.1 Estimation

The parameters of our Gaussian mixture density from eq. 1 are subject to a number of natural constraints: The variances of the component densities $\sigma_k^2(X_{t-1})$ must be positive. Also, the weights $\alpha_k(X_{t-1})$ must be positive and add up to 1. Translating these restrictions into a set of constraints for the neural network's weights would be very challenging, if not impossible. Instead, we transform the neural network's weight and variance outputs to enforce fulfilling the constraints. More precisely, let $\alpha_k(X_{t-1})$ be the MDN output for a weight and $\sigma_k^2(X_{t-1})$ be the MDN output for the variance of a mixture component. The actual weight and variance of that component is then set to

$$\tilde{\sigma}_k^2(X_{t-1}) = e^{\tilde{\sigma}_k^2(X_{t-1})},$$
$$\tilde{\alpha}_k(X_{t-1}) = \frac{e^{\alpha_k(X_{t-1})}}{\sum_{i=1}^{K} e^{\alpha_i(X_{t-1})}}.$$

These transformations allow us to maintain an unconstrained output for the neural network while fulfilling the natural parameter constraints of a mixture density model.

We now collect a training data set $\{(r_t, X_{t-1})\}_{t\in[1,T]}$. As we are aiming for a full density estimator, we fit the neural network parameters $\theta = (w_{1,i,h}, w_{2,j,h}, w_{2,j,bias})$ by maximizing the conditional log-likelihood of the observed data points:

$$\theta^* = \arg\max_{\theta} \sum_{t=1}^{T} log\, \hat{p}(r_t|X_{t-1};\theta). \tag{3}$$

There exists a large body of literature about the training of neural networks. Bishop (2006) and Géron (2017) provide a good introduction and overview of different training techniques. We use stochastic gradient descent with the adaptive learning rate method of Kingma and Ba (2015) to train our Mixture Density Network. In a nutshell, this local optimization technique repeatedly iterates over the complete training data set and adjusts the parameters step-by-step to draw closer to the optimal solution. The starting values for the weights of the MDN are set randomly. Within each iteration, the method starts in the beginning of the training data set, feeds the respective input data point into the neural network and computes the model-implied likelihood of the associated observed return. The optimizer now slightly adjusts the weights along their gradients of the likelihood function to increase the likelihood of that return in subsequent estimations. Afterwards, it proceeds to the next observation and repeats updating the weights until the end of the data set is reached. This process is repeated L times.

A central issue when training neural networks is determining the optimal degree of complexity of the model. Models with too limited capacity may not be a able to sufficiently capture the structure of the data, inducing a restriction bias. On the other hand, if a model is too expressive, it is prone to over-fit the training data, resulting in poor generalization. There are three hyper-parameters embedded in our approach that allow to adjust the complexity of the Mixture Density Network: the number of mixture components K, the number of hidden neurons H and the number of training

iterations L. Increasing any of these hyper-parameters makes the model more complex.
In order to pin down the hyper-parameters and find the best degree of complexity for our use case, we perform a cross-validation-based hyper-parameter grid search. More precisely, we divide the training data set into 10 equally sized folds. For each hyper-parameter combination on the grid, we in turn leave out one of the folds and use the 9 other folds for training. We now train the MDN 5 times on this trimmed training data set with different starting values for the weight optimization. After each training, we record the average conditional log-likelihood for the returns in the fold that was left out. We arrive at a performance figure for a given hyper-parameter combination by taking the average log-likelihood across all folds and starting values. Finally, we choose the hyper-parameter combination with the highest performance, that is, the highest cross-validation conditional likelihood.

3 Data

Our forward-looking estimate of the physical return density heavily relies on risk-neutral information. We use the moments of the risk-neutral distribution, which can be inferred from option prices, as inputs for our estimation. To this end, we obtain end-of-day quotes for S&P 500 options from the Chicago Board Options Exchange (CBOE). The option data spans the period January 2004 until July 2017 and comes with matched underlying prices at the time of the option market's close. Bakshi et al. (2003) showed how the variance, skewness and kurtosis of the risk-neutral distribution can be backed out from option prices in a model-free way via aggregation over the strike range. Martin (2017) introduced the SVIX, a similarly constructed measure, that is closely related to the risk-neutral variance and that constitutes a lower bound for the expected return under reasonable assumptions. Due to its theoretical merits, we thus replace the risk-neutral variance estimate of Bakshi et al. (2003) with the SVIX in our physical density predictions.
A careful and precise estimation of the SVIX and the risk-neutral moments from the raw option data is crucial for our analysis. Ulrich and Walther (2018) compare several popular approaches for building the implied volatility surface and thus risk-neutral moment estimates and find large differences in the estimates that result from the choice of the calculation method. These differences are especially pronounced in the tail regions of the risk-neutral density, which are of high importance for the risk-neutral skewness and kurtosis measures that we use. We follow their advise and use a kernel regression over the strike range for interpolating the implied volatility surface at observed maturities. After having obtained the risk-neutral measures for observed maturities by aggregating appropriately over the strike range, we linearly interpolate these along the maturity dimension to a fix maturity of 30 calendar days. In that, we exclude options with maturities of less than 7 days as the term structure of risk-neutral moments becomes increasingly non-linear for very short-term options. At times, this filter removes all observations with maturities that are shorter than 30 calendar days, in which case we linearly extrapolate the moment estimates that relate to longer-term maturities.

Table 1: Data Summary Statistics

			Percentiles				
	Mean	Volatility	10%	25%	50%	75%	90%
S&P 500 return	0.03	1.17	-1.11	-0.39	0.07	0.53	1.13
Riskfree rate	0.005	0.007	0.00	0.00	0.001	0.008	0.019
$SVIX^2$	394.30	494.35	134.12	167.20	238.33	413.58	720.12
QSkew	-2.46	0.94	-3.46	-2.93	-2.41	-1.94	-1.51
QKurt	24.37	16.62	8.59	13.16	20.85	30.27	43.11
Value return	0.003	0.64	-0.57	-0.28	-0.00	0.26	0.58
Size return	0.002	0.57	-0.67	-0.34	0.01	0.34	0.67
Momentum return	-0.002	0.96	-0.94	-0.38	0.06	0.43	0.89
S&P 500 risk	1.88	6.01	0.22	0.40	0.84	1.77	3.71
Value Risk	0.41	0.92	0.04	0.07	0.14	0.29	0.77
Size risk	0.33	0.41	0.10	0.15	0.23	0.35	0.59
Momentum Risk	0.93	2.01	0.09	0.16	0.32	0.74	1.83
N	3303						

The table shows summary statistics for our full S&P 500 data set. All figures are in percentage values. We use log returns, the factor risk rows represent 10-trading-day rolling window variances of the respective log returns. The SVIX is by definition a measure for risk-neutral volatility. We use its squared version to make it consistent with our remaining risk measures, which are expressed in variance terms, and because $SVIX^2$ represents a lower bound for the expected equity risk premium, according to Martin (2017). QSkew (QKurt) denote the risk-neutral skewness and kurtosis measures. All quantities refer to the daily time-interval, beside the $SVIX^2$, which is annualized due to its definition. The data set spans the period 2004-01-05 to 2017-07-18.

In order to compare our results for purely forward-looking inputs with density predictions based on backward-looking data, we also obtain time series information. First, as we are interested in predicting the density of S&P 500 returns, we get daily S&P 500 closing prices from Bloomberg. Furthermore, we collect daily return time series for the size, value and momentum factor from Kenneth French's website and treat the S&P 500 as a measure for the market factor. To obtain a time-varying physical measure for factor risk, we compute the variance of factor returns over rolling 10-trading-day windows for each factor separately. We use Overnight Index Swap (OIS) rates as our measure for the risk-free rate, which we source from Bloomberg. We obtain all of these time series data for the same time frame as we have option data available.

We merge risk-neutral moment, return and physical risk time series that we gather and match them with subsequent S&P 500 returns. We drop days for which one of our inputs it not observed. Table 1 contains summary statistics of our data set. We split this full data set into a training and a validation subset. The training set consists of the first 80% of observations, while the last 20% of observations form the validation set. The estimation that we laid out in section 2.1, including the hyper-parameter selection, is only based on the training data set, such that the validation set is completely untouched before we evaluate a predictor's performance on it.

4 The Forward-Looking Return Density

4.1 Using purely forward-looking information

We start our analysis by inspecting the predictive information content of risk-neutral measures with respect to the complete daily return distribution. In particular, we check whether restricting the predictor variables to purely forward-looking measures comes along with a reduction in prediction performance. We measure performance of a density predictor via the average log-likelihood of the return observations $\{r_{t+1}\}_{t=1,\dots,T}$, given the density estimator $\hat{p}(\cdot)$ and the predictive variables $\{X_t\}_{t=1,\dots,T}$:

$$\frac{1}{T}\sum_{t=1}^{T}\log\hat{p}(r_{t+1}|X_t) \tag{4}$$

Intuitively, a conditional density estimator will reduce the likelihood of some potential returns and increase the likelihood of other returns based on some observations of X_t. If X_t is informative about the future return and if the density estimator is well specified, the ex post realizations of the returns will be more likely to show up in the regions with increased likelihood, thus leading to an increase in the average log-likelihood.

As the average log-likelihood is a relative performance measure between competing model specifications, we compare the Mixture Density Network estimator to a benchmark density estimator. In the spirit of Welch and Goyal (2008), who use the unconditional mean of the in-sample data set as benchmark for in- and out-of-sample mean predictions, we use the smoothed empirical in-sample return density as our benchmark. For smoothing, we apply a Gaussian kernel density estimator and select the bandwidth parameter via cross-validation, as suggested by Härdle (1991). We estimate three MDN densities, which differ in their input variables. The first MDN uses purely forward-looking information, i.e. the risk-free rate, the SVIX, risk-neutral skewness and kurtosis. The second MDN is based on purely backward-looking information, which is past S&P 500 and FF3 factor returns and volatilities. Finally, we consider a MDN specification that uses both types of information as input variables. In all cases, including the benchmark kernel density, we fit the estimator to the training data set. We then calculate the average log-likelihoods of the fitted density estimators for the training (in-sample) and the validation (out-of-sample) data set separately. As fitting a MDN employs a local optimization routine, the final conditional estimator may depend on the random starting values of the optimization. To address this concern, we fit each estimator 100 times with different starting values and calculate the average performance across all fits. We also calculate the standard deviation of the individual performance evaluations to get a sense for the stability of the estimator with respect to the starting values.

Panel a of table 2 contains the result of our performance evaluation. Throughout our estimations, the MDN estimator showed a higher average log-likelihood than the benchmark method, both in-sample and out-of-sample. In relative terms, the average log-likelihood of the forward-looking estimator is 4.7% higher than the benchmark in-sample and 4.6% higher out-of-sample. The similar performance increase in the in- and out-of-sample set is a first hint that the forward-looking

estimator is generalizing well. The standard deviation of the log-likelihood due to different starting values is low, indicating that the neural network fitting is not easily trapped in a local optimum, but instead converges to a robust estimator. Using purely backward-looking information not only comes along with a lower log-likelihood, but also only reaches a 0.9% performance increase against the unconditional benchmark in the out-of-sample set, after a 4.0% increase in the in-sample set. This decrease of the performance spread is a sign for over-fitting in the backward-looking estimator. Furthermore, adding backward-looking information to the forward-looking MDN does not appear to improve the estimator sustainably. The performance slightly increases in-sample, but does nearly not change out-of-sample, leading to a decrease in the performance spread against the bechmark between the in- and out-of-sample data set. It thus seems like the past return and risk information only introduces a slight tendency of over-fitting into the estimator. Our results show that a purely foreward-looking MDN estimator of the conditional return density improves on the unconditional return density and is most favorable with respect to performance and generalization. Figure 1 shows time series charts of the S&P 500 price development along with the expected physical moments of our forward-looking density estimator.

We continue by asking whether the MDN estimator is consistent with the data from a statistical point of view. To this purpose, we apply the tests of Berkowitz (2001) and Knüppel (2015). The null hypothesis of these tests is that the observations of a random variable are drawn from a given density estimator. Panel b of table 2 contains p-values for this null hypothesis for the in-sample and out-of-sample data set, separately. To obtain the p-values for the out-of-sample data set, we use the density estimator that was trained on the in-sample data set. In general, the findings of the two tests are strongly aligned. The tests reveal that the unconditional kernel density estimator appears to be inappropriate, as it is strongly rejected out-of-sample and even in-sample by the Berkowitz (2001) test. The forward-looking estimator shows high p-values and we cannot reject the null, that this estimator describes the true conditional density of returns. For the backward-looking estimator, in-sample p-values are high, but the estimator is very strongly rejected out-of-sample by both tests. Again, this is a clear sign of the over-fitting behaviour of the backward-looking estimator. The estimator that uses both, backward- and forward-looking information, is not rejected by either test. Beside the Berkowitz (2001) test of the in-sample data set, the p-values are considerably lower than for the forward-looking estimator, though.

At first sight, it appears puzzling that the benchmark estimator is rejected in-sample by the Berkowitz (2001) test. The p-value of the forward-looking estimator is low, too. This finding can be explained by the structure of the Berkowitz (2001) test, which not only checks whether the distribution matches, but also if there is autocorrelation in the conditional probabilities. If we disable the autocorrelation check, the p-value of the benchmark rises to 0.111, the forward-looking estimator's p-value even increases to 0.805 in the in-sample data set. Autocorrelation in the conditional probabilities can be induced by autocorrelation in the returns, if the estimator does not account for that. As neither the benchmark nor the forward-looking estimator conditions on past return data, they cannot correct for return autocorrelation. In contrast, the backward-looking

Table 2: Performance and validity of MDN conditional density estimates

(a) Average log-likelihood

Model	In-Sample	Out-of-Sample
Benchmark	3.1774	3.4318
Forward-looking	3.3327 (0.0056)	3.5968 (0.0094)
Backward-looking	3.3156 (0.0056)	3.4704 (0.0096)
All information	3.3524 (0.0082)	3.5984 (0.0141)
N	2642	661

(b) Density test p-values

	Berkowitz (2001)		Knüppel (2015)	
Model	In-Sample	Out-of-Sample	In-Sample	Out-of-Sample
Benchmark	0.000	0.000	0.466	0.001
Forward-looking	0.066	0.81	0.7033	0.974
Backward-looking	0.767	0.000	0.681	0.000
All information	0.537	0.549	0.605	0.386
N	2642	661	2642	661

Panel a of this table shows the average log likelihood of S&P 500 return observations for different conditional density estimators over 100 estimation runs with different starting values in the optimization. We call the unconditional kernel density estimator on the in-sample data set *Benchmark*. We estimate three conditional MDN density estimators, which differ in their input data. The *forward-looking* estimator is trained solely on risk-neutral information, the *backward-looking* estimator conditions on past returns and physical risk measures only. We also train an *all information* estimator, which uses both input variables sets. Standard errors of the neural network log-likelihood estimates due to random starting values of the neural network optimization are given in parentheses in panel a. Panel b shows p-values of the null hypothesis that the density estimator correctly specifies the return density.

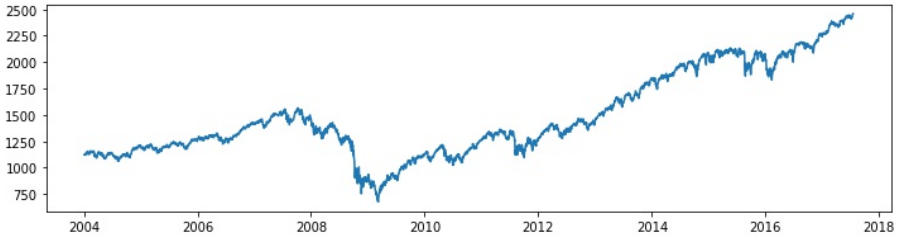
estimator, which makes use of past return data, shows a high p-value. The exploitation of return autocorrelation is dangerous though, as can be seen from the out-of-sample p-values. Here, the forward-looking estimator has a very high p-value, while the backward-looking estimator is strongly rejected. The reason for this observation is that the autocorrelation does not persist in the out-of-sample data set. Again, the result is due to the over-fitting behavior of the backward-looking estimator, which does not show up in the forward-looking estimator.

4.2 Trading on the conditional return distribution

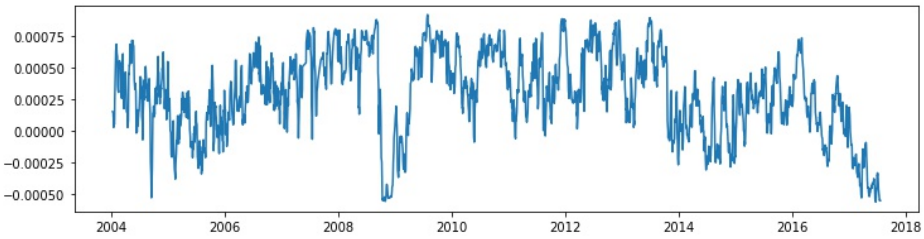
If the forward-looking MDN estimator is informative about the future return distribution, as our previous results suggest, it should be possible to derive profitable trading strategies based on the

Figure 1: S&P 500 price development and expected $\mathcal{P}$ moments

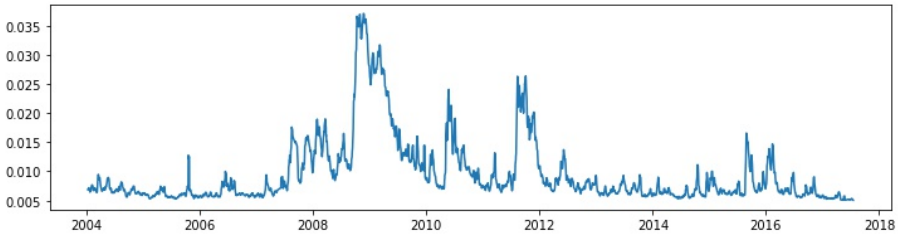
(a) S&P 500 Price



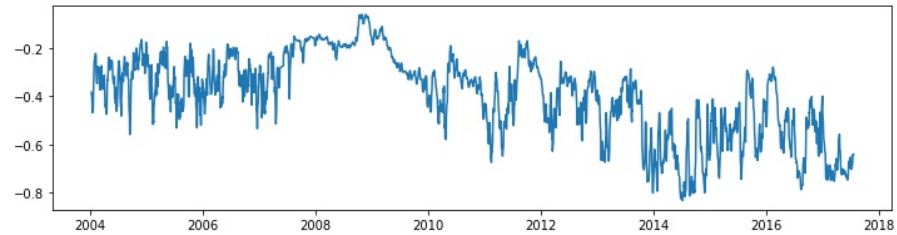
(b) Expected Mean



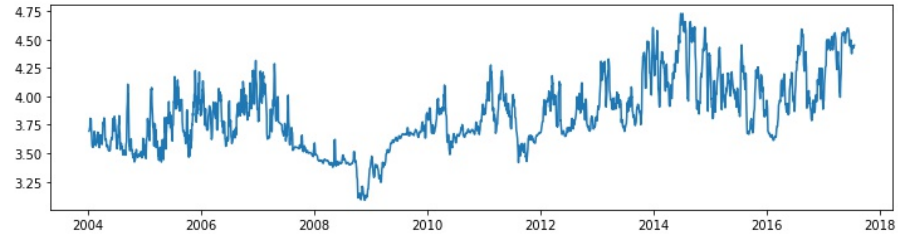
(c) Expected Standard Deviation



(d) Expected Skewness



(e) Expected Kurtosis



Panel a shows the price development of the S&P 500 between January 2004 and July 2017. Panels b through e show one week rolling window averages of the MDN-implied forward-looking mean, standard deviation, skewness and kurtosis expectation.

density forecast. In this section, we present two simple trading schemes that exploit information about the conditional physical mean and volatility, that the MDN implicitly predict. Essentially, both trading strategies use the moment prediction as a signal to increase or decrease a market position.

For our first strategy, we obtain the daily conditional one-day-ahead mean expectation $\hat{m}_{1,t+1}$ as

$$\hat{m}_{1,t+1} = E[r_{t+1}|X_t] = \int_{-\infty}^{\infty} u\,\hat{p}(u|X_t)\,du. \tag{5}$$

Here, $\hat{p}(\cdot)$ denotes the MDN estimator and X_t is a vector of all forward-looking variables that we observe at time t. Each day, we invest a fraction of wealth into the S&P 500 at its closing price and hold that position for one day. The fraction of wealth that we invest is defined as $w_t = \frac{\hat{m}_{1,t+1}}{\sigma(\hat{m}_{1,t+1})}$, where $\sigma(\hat{m}_{1,t+1})$ represents the standard deviation of the mean estimate. Note, that the weights are determined by the expected mean itself. Any other denominator beside the standard deviation could also be used and would only scale the returns of our trading strategy. The weight definition that we use can lead to negative position weights, in which case we initiate a short position in the S&P 500. If the weight exceeds 1, we assume that the position is leveraged accordingly. We compare the strategy returns to a simple buy-and-hold strategy's returns.

Table 3 gives an overview of this simple mean trading strategy. On average, the market timing strategy is invested with 80.7% of total wealth, with a standard deviation of 100%. Although the average fraction of wealth invested is lower than for the buy-and-hold strategy, the market timing strategy manages to realize an average yearly excess return of 16.4%, which is 9.2% higher than the average return of the buy-and-hold strategy. However, this higher average excess return comes along with an increased volatility of annualized 25.5%. Still, using the MDN mean forecast as a signal to time the market increases the Sharpe ratio of the buy-and-hold strategy of 0.388 by more than 78% to 0.694. The results for the full data set are confirmed by looking at the out-of-sample data set only. Although the market volatility and thus the potential for market timing has been significantly lower in the out-of-sample set than in the full data set, our market timing strategy still increases the Sharpe ratio of the buy-and-hold strategy from 0.647 to 0.743, an increase of nearly 15%. This increase results from a 1.5% higher annual excess return for the market timing strategy with a volatility increase of only 0.59%. In summary, timing the market based on the MDN's mean return forecast appears to be a beneficial and robust trading strategy.

We proceed with our second trading strategy, which makes use of the implicit volatility forecast of the MDN. Bakshi and Kapadia (2003) describe a negative variance risk premium, that can be earned by shorting options. We therefore base our volatility trading strategy on shorting at-the-money (ATM) straddles. A straddle consists of a Call and a Put option with the same strike and maturity. On expiration, the position earns a negative return, if the price of the underlying barely changed. On the other hand, if the price of the underlying moved strongly, the position will earn a positive return. On average, ATM straddles have significantly negative returns, which is consistent with a negative variance risk premium. However, it should be possible to time straddle returns, if

Table 3: Mean trading summary statistics

	Full data set		Out-of-sample	
Strategy	Buy-and-Hold	MDN	Buy-and-hold	MDN
Mean return	8.48 (5.11)	17.67 (7.02)	8.43 (7.89)	9.94 (8.25)
Mean excess return	7.18 (5.11)	16.38 (7.03)	8.28 (7.9)	9.78 (8.26)
Excess return volatility	18.52	25.45	12.79	13.38
Sharpe Ratio	0.388	0.694	0.647	0.743
N	3303	3303	661	661

The table shows annualized daily expected return, excess return, excess return volatility and Sharpe Ratio of the conditional mean trading strategy in the S&P 500, compared with the buy-and-hold strategy. The numbers in parentheses are standard errors of the average return estimates. The full data set covers January 2004 to July 2018. The out-of-sample data set covers the last 20% of the full data set and thus starts in November 2014.

a robust volatility forecast is available.

To this purpose, we calculate the conditional one-day-ahead second moment $\hat{m}_{2,t+1}$ on a daily basis as

$$\hat{m}_{2,t+1} = E[r_{t+1}^2|X_t] = \int_{-\infty}^{\infty} u^2 \, \hat{p}(u|X_t) \, du \tag{6}$$

and construct the volatility forecast $\hat{\sigma}_{t+1} = \sqrt{\hat{m}_{2,t+1} - \hat{m}_{1,t+1}^2}$. For each day, we select the strike that is closest to the S&P 500 price and the option series with the shortest maturity larger than 6 days. Thus, the combined position's delta exposure is approximately 0. At the selected strike and maturity, we initiate a straddle at the CBOE's reported end-of-day mid price. After one day, we close the position at their end-of-day mid price. Unconditionally, we would short the straddle position. However, if we predict the S&P 500's standard deviation to rise, we reduce the short position, or even go long the straddle. On the other hand, if we predict a lower standard deviation for the next day than for today, we increase the short position. More precisely, the position weight is defined as

$$\begin{aligned} \Delta\hat{m}_{2,t+1} &= \hat{m}_{2,t+1} - \hat{m}_{2,t} \\ w_t &= -1 + \frac{\Delta\hat{m}_{2,t+1}}{\sigma(\Delta\hat{m}_{2,t})}, \end{aligned}$$

where $\sigma(\Delta\hat{m}_{2,t})$ is the standard deviation of the daily change in the standard deviation forecast. If our volatility forecast is informative about actual future volatility, we expect this straddle timing to be beneficial for two reasons. First, a higher volatility expectation increases the probability for larger returns, i.e. high volatility realizations, in the underlying. Since a straddle essentially represents a bet on volatility and a large return realization increases the expected pay-off, such a high volatility realization directly increases the straddle's price. Second, as Bakshi and Kapadia (2003) showed, higher realized volatility comes along with an increase in the variance risk premium,

Table 4: Volatility trading summary statistics

	Full data set		Out-of-sample	
Strategy	Buy-and-Hold	MDN	Buy-and-hold	MDN
Mean return	523.86 (53.9)	1392.02 (79.03)	753.23 (144.12)	1464.58 (168.16)
Mean excess return	525.15 (53.9)	1393.29 (79.02)	753.61 (144.13)	1464.97 (168.17)
Excess return volatility	195.01	285.92	233.24	272.15
Sharpe Ratio	2.69	4.87	3.23	5.38
N	3299	3299	660	660

The table shows annualized daily expected return, excess return, excess return volatility and Sharpe Ratio of the conditional standard deviation trading strategy in S&P 500 straddles, compared with the always short strategy. The numbers in parentheses are standard errors of the average return estimates. The full data set covers January 2004 to July 2018. The out-of-sample data set covers the last 20% of the full data set and thus starts in November 2014.

which would translate into an increased straddle price. Together, straddle prices should rise if volatility increases.

Table 4 compares the returns of the fix always-short strategy and our straddle timing strategy. As is known in the literature, average returns to shorting options in general, and straddles in particular, are sizable. Constant shorting of straddles earns an average log excess return of 525% per year.[8] This high average return is put into perspective by the standard deviation of 195%, leading to a Sharpe ratio of 2.69. By using the MDN estimator's volatility forecast, the Sharpe ratio can be increased by more than 80% to 4.87. Out-of-sample, the volatility forecast increases the Sharpe ratio of 3.23 by 66% to 5.38.

Taken together, our results show that the MDN estimator can be used to time the market. This market timing becomes possible due to the predicting capabilities of our forward-looking estimator for the future mean and volatility of S&P 500 returns. In both cases, the results for the out-of-sample set strongly support the findings for the full data set. This confirms our previous findings that the MDN is not over-fitting the training data and generates a robust forward-looking density predictor.

4.3 Determinants of Index Return Densities

In the previous sections, we've established the Mixture Density Network as a flexible and robust predictor for the S&P 500 return distribution. The MDN bases its predictions on moments of the risk-neutral distribution. As a result, we can express the conditional return distribution, represented by its conditional moments, as a function of the risk-neutral moments. We further ask, whether all of these risk-neutral moments matter for the density prediction. To this end, we apply an adjusted form of the test of Patton and Timmermann (2010) to allow statistical inference about the impact

[8] We calculate returns with respect to the initial price of the straddle. Sometimes, returns are also calculated with respect to the underlying price, which technically makes average returns appear lower.

of a predictor variable on the conditional return distribution.

We start our analysis by estimating the MDN on the complete data set. Let x_i be the i-th predictor variable of the MDN, in our case the i-th risk-neutral moment, and $X' = (x_1, ..., x_{i-1}, x_{i+1}, ..., x_N)^\intercal$ be a vector of the remaining input variables. As is it not essential for this analysis, we dropped the time index for ease of notation. We obtain the 10% (90%) percentile of the observed value range of x_i, x_i^{low} (x_i^{high}), and discretize the interval $[x_i^{low}, x_i^{high}]$ into 100 data points with equal step size $x_i^{(n)}$, $n \in [1, 100]$. For each discretized data point, we calculate the MDN-implied conditional l-th centralized and normalized moment for $l \in [1, 2, 3, 4]$, i.e. mean, variance, skewness and kurtosis:

$$\hat{m}_{1,i}^{(n)} = E(r^1 | (x_i^{(n)}, X')), \tag{7}$$

$$\hat{m}_{2,i}^{(n)} = E((r - \hat{m}_{1,i}^{(n)})^2 | (x_i^{(n)}, X')), \tag{8}$$

$$\hat{m}_{3,i}^{(n)} = \frac{E((r - \hat{m}_{1,i}^{(n)})^3 | (x_i^{(n)}, X'))}{\left(\hat{m}_{2,i}^{(n)}\right)^{1.5}}, \tag{9}$$

$$\hat{m}_{4,i}^{(n)} = \frac{E((r - \hat{m}_{1,i}^{(n)})^4 | (x_i^{(n)}, X'))}{\left(\hat{m}_{2,i}^{(n)}\right)^{2}} \tag{10}$$

In that, we set the elements of the vector X' to their unconditional median estimates. Following Patton and Timmermann (2010), a test for the impact of x_i on $\hat{m}_l$ can now be built on this empirical representation of the functional relationship between the i-th predictor variable and the l-th conditional return moment. We identify the indices u and d that satisfy $\hat{m}_{l,i}^{(u)} = \max_n \hat{m}_{l,i}^{(n)}$ and $\hat{m}_{l,i}^{(d)} = \min_n \hat{m}_{l,i}^{(n)}$, that is the indices on the discretized range of $\{x_i\}$ for which we observed the lowest and the highest conditional moment estimate. If x_i does not predict the l-th conditional moment, this lowest and highest conditional moment estimate should be equal, which leads us to the null hypothesis

$$H_0: \quad \hat{\Delta}_{l,i} = \hat{m}_{l,i}^{(u)} - \hat{m}_{l,i}^{(d)} = 0. \tag{11}$$

We can quantify the likelihood of this null hypothesis via bootstrapping. We apply the stationary bootstrap of Politis and Romano (1994) and re-sample the training data $B = 1000$ times. For each re-sampled data set, we re-fit the MDN and obtain the bootstrapped moment samples along the discretized range of x_i, $\hat{m}_{l,i}^{(n,b)}$, $b \in \{1, ..., B\}$. For each bootstrap, we calculate the test statistic $\hat{\Delta}_{l,i}^{(b)} = \hat{m}_{l,i}^{(u,b)} - \hat{m}_{l,i}^{(d,b)}$ and eventually estimate the p-value for H_0 as

$$J_{l,i}^{(b)} = \hat{\Delta}_{l,i}^{(b)} - \hat{\Delta}_{l,i}$$

$$\hat{p}_{l,i}^{0} = \frac{1}{B} \sum_{b=1}^{B} 1_{\left[J_j^{(b)} > \hat{\Delta}_j\right]},$$

where $1_{\left[J_j^{(b)} > \hat{\Delta}_j\right]}$ is an indicator function that is 1 if the condition in brackets is fulfilled and 0 otherwise.

Table 5: Impact of predicting variable on the conditional return density

Variable	Mean	Variance	Skewness	Kurtosis
Riskfree rate	0.416	0.037**	0.286	0.288
$SVIX^2$	0.000***	0.000***	0.000***	0.004***
QSkew	0.231	0.117	0.000***	0.000***
QKurt	0.052*	0.165	0.007***	0.001***

The table shows p-values for the null hypothesis that the predicting variable in the first column is not informative about the moment of the column label. We use stationary bootstrapping with 1000 iterations and the test of Patton and Timmermann (2010) to obtain these estimates. If the null hypothesis is rejected at the 10% level, we mark the entry with a star, two and three stars mark rejection at the 5% and 1% level.

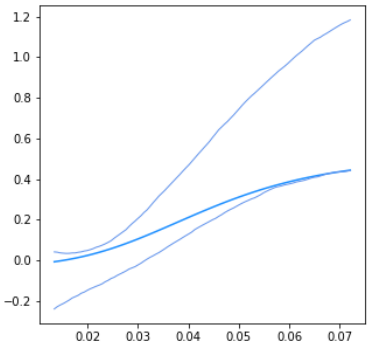
Table 5 shows our estimated p-values for H_0 for each input and the first four moments of the forward-looking return density. All of our forward-looking input variables are informative about the conditional return density at the 5% significance level. In particular, $SVIX^2$ helps predict the mean, standard deviation and higher moments of the return distribution. In all cases, the null hypothesis of no influence is very strongly rejected. Risk-neutral skewness and kurtosis are especially relevant for predicting their physical counter-parts. The evidence for an impact of these variables on the conditional mean and volatility is comparably weak and only significant at the 10% level. Finally, the riskfree rate only helps to pin down volatility expectations, which is significant at the 5% level. Surprisingly, we cannot reject the null that the riskfree rate is uninformative about the S&P 500 return. Our tests reveal that especially option-implied variables are robust predictors for the subsequent return distribution. However, these tests do not tell us about the shape of the predicting relationships, which is in principle unconstrained. We will now take a detailed look at these relationships, as they are estimated by the MDN.

In order to qualitatively and quantitatively inspect the impact of a forward-looking variables on the return density prediction, we make use of our empirical representations of the conditional moments along the input range, $\hat{m}_{l,i}^{(n)}$, and their respective bootstrap samples $\hat{m}_{l,i}^{(n,b)}$. We follow Davison and Hinkley (1997) in constructing 90% confidence intervals for the predicting relationships that the MDN identifies. For that, we first calculate the differences $\delta_{l,i}^{(n,b)} = \hat{m}_{l,i}^{(n,b)} - \hat{m}_{l,i}^{(n)}$. Let $\delta_{l,i}^{(n,0.05)}$ ($\delta_{l,i}^{n,(0.95)}$) be the empirical 5% (95%) percentile of $\delta_{l,i}^{(n,b)}$. The bootstrapped confidence interval for $\hat{m}_{l,i}^{(n)}$ is then $[\hat{m}_{l,i}^{(n)} - \delta_{l,i}^{(n,0.95)}, \hat{m}_{l,i}^{(n)} - \delta_{l,i}^{(n,0.05)}]$.

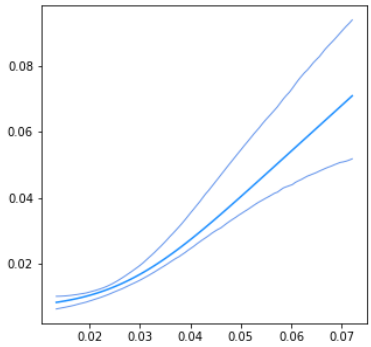
Figure 2 plots $\hat{m}_{l,i}^{(n)}$ for the input-moment relationships that we identified as significant at the 1% level in our previous tests. We start our inspection with the only forward-looking variable that predicts S&P 500 returns in our set-up, the $SVIX^2$. Panel a shows how the annualized conditional day-ahead return expectation changes, as the $SVIX^2$ changes. Under certain conditions, Martin (2017) shows that the $SVIX^2$ constitutes a lower bound for the equity risk premium. In line with this finding, our expected return rises with an increase in $SVIX^2$ and a linear relationship is well possible. For high values of $SVIX^2$ though, it appears like the lower bound is not tight, at least on the daily horizon. For example, an annualized $SVIX^2$ of 0.07 leads to an annualized day-ahead

Figure 2: Conditional S&P 500 moment by risk-neutral moment

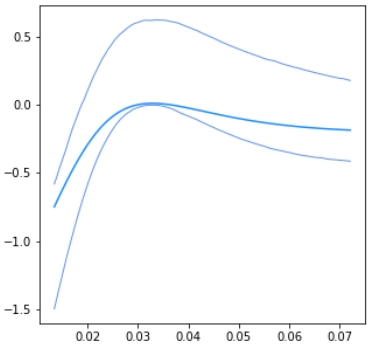
(a) Conditional mean by SVIX^2



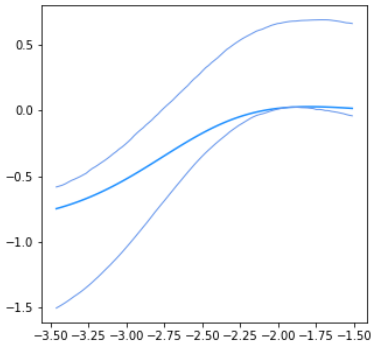
(b) Conditional variance by SVIX^2



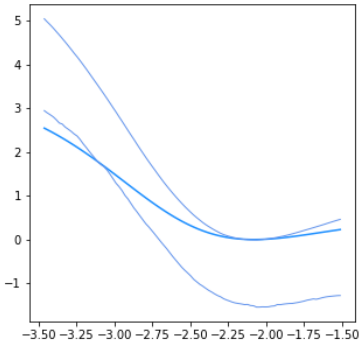
(c) Conditional skewness by SVIX^2



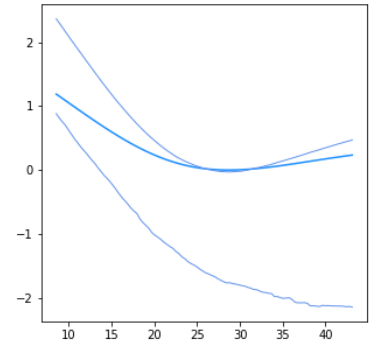
(d) Conditional skewness by QSkew



(e) Conditional excess kurtosis by QSkew



(f) Conditional excess kurtosis by QKurt



This figure shows selected predicting relationships between a forward-looking input variable and a conditional physical moment. Each panel's title indicates what it shows: A title of X *by* Y would mean that the MDN moment forecast of X is plotted on the vertical axis, and conditioned on Y, which is plotted on the horizontal axis. All other input variables beside Y are set to their unconditional median. The conditional mean and variance forecasts are annualized.

mean expectation well above 40%, if all other variables are at their unconditional medians. It is important to note that we use the $SVIX^2$ at the 30-day horizon to form expectations about the day-ahead return. If there exists a downward-sloping term structure in the SVIX, the lower bound relationship might still be tight. Calculating backwards, an annualized day-ahead return expectation of 40% would imply a value of the SVIX at the one-day horizon of 63.2%, if the lower bound is tight.

In essence, the $SVIX^2$ is a measure for risk-neutral variance expectations. Several studies (Chernov, 2007; Busch et al., 2011; Bekaert and Hoerova, 2014) document a strong relationship between risk-neutral variance expectations and subsequent variance realizations. Panel b of figure 2 shows the MDN-implied volatility forecast as a function of the $SVIX^2$. Note, that the MDN is not constrained to a linear link between risk-neutral and physical variance, but instead retrieves the relationship from the data. Nevertheless, the MDN also models an approximately linear relationship, but only if the SVIX is above 17% ($SVIX^2 > 0.03$). For lower values of the SVIX though, there appears to exist a non-linearity, which causes higher physical variance expectations compared to a general linear link. At the same time, the 90% confidence bands become very dense at this point, indicating that the estimator is relatively certain about the shape of the relationship for low SVIX values. The relationship allows the following interpretation: An increase of the SVIX from low to intermediate levels is primarily driven by an increase in the variance risk premium, as physical variance expectations are not rising strongly. Further increases in the SVIX are then driven by increases in physical variance expectations.

We now turn to the impact of option-implied variables on physical expectations of higher moments of the return density. Panels c and d of figure 2 show, how skewness expectations change with the $SVIX^2$ and risk-neutral skewness, panels e and f illustrate the response of conditional kurtosis to changes in risk-neutral skewness and kurtosis. The MDN estimates strongly non-linear responses of conditional higher moments to changes in the forward-looking inputs. For example, an increase in $SVIX^2$ from 0.015 (corresponds to a SVIX of 12.2%) to 0.03 (corresponds to a SVIX of 17.3%) leads to a corresponding increase in physical skewness from -0.62 to nearly 0. However, further increases in the $SVIX^2$ have nearly no significant impact on conditional skewness expectations. This finding is consistent with the notion that risk hides in the left tail in times of low market volatility. In times of higher risk-neutral volatility expectations, the return distribution is not significantly left skewed. A similar finding exists for the impact of risk-neutral skewness. Here, an increase in risk-neutral skewness also increases physical skewness expectations, but only up to a risk-neutral skewness of about -2. Similarly, for low values of risk-neutral skewness, conditional kurtosis is high and the return distribution becomes less heavy-tailed as it becomes less left skewed. Again, the effect disappears at a risk-neutral skewness of about -2 and expected return kurtosis stays nearly constant for larger risk-neutral skewness values. The MDN identifies a surprising relationship between risk-neutral and physical return kurtosis. As risk-neutral kurtosis increases, physical kurtosis drops until it reaches the normal distribution's kurtosis and stays roughly constant beyond this point.

In summary, the MDN identifies a range of non-linearities in the relationship of risk-neutral and

physical moments. The forward-looking return distribution appears to be closer to the normal distribution in crisis times and more strongly left skewed in calm times. At the same time, strong market fears about sudden crashes, as signaled by low risk-neutral skewness expectations, appear justified by a more heavy-tailed and left skewed physical return distribution. Our tests show that the MDN identifies relationships between risk-neutral and physical moments that are statistically robust and economically relevant.

5 Robustness

5.1 Conditional Density Estimation Method

In this study, we present a forward-looking P density estimator that is based on a neural network. However, the forward-looking feature is introduced by basing our forecasts purely on risk-neutral information and independent of the use of a Mixture Density Network as conditional density estimator. We therefore inspect in this section, whether more traditional density estimators can also be used to form valid P densities, and whether the MDN does indeed perform better than these alternatives. To this purpose, we perform an out-of-sample horse race between the MDN and popular parametric, non-parametric and machine learning techniques. We compare these methods with respect to their average conditional log likelihood, as defined in eq. 4, as well as their root mean squared prediction error for the mean and volatility:

$$\text{Mean RMSE} = \sqrt{\frac{1}{T}\sum_{t=t^*+1}^{T}(r_t - E(r_t|X_{t-1}))^2}, \tag{12}$$

$$\sigma \text{ RMSE} = \sqrt{\frac{1}{T}\sum_{t=t^*+1}^{T}(|r_t - E(r_t|X_{t-1})| - \sigma(r_t|X_{t-1}))^2}, \tag{13}$$

where $E(r_t|X_{t-1})$ denotes the conditional mean expectation and $\sigma(r_t|X_{t-1})$ the conditional volatility expectation of a density estimator. All methods are trained on a training data set $\{(r_t,\ X_{t-1})\}_{t\in[1,t^*]}$, such that we obtain out-of-sample performance figures based on the validation data set $\{(r_t,\ X_{t-1})\}_{t\in[t^*+1,T]}$. As before, the validation data set consists of the last 20% of our total available data. Finally, we test whether a conditional density estimator can be rejected statistically by applying the distribution tests of Berkowitz (2001) and Knüppel (2015).

We consider six alternative density estimators in our performance evaluations. First, as described in section 4.1, we use a Gaussian kernel density estimator as unconditional benchmark. We further estimate a parametric model, where we assume that returns are conditionally normally distributed and conditional mean and variance are linear in the forward-looking predictor variables. We inspect three different non-parametric estimator. The Conditional Kernel Density Estimator (CKDE) is closely related to our unconditional benchmark. It first forms an estimate of the joint distribution of r_t and X_{t-1} and then obtains a conditional density estimate by building the marginal distribution at

Table 6: Out-of-sample performance of conditional density estimators

Model	Avg. $\log \mathcal{L}$	Mean RMSE	σ RMSE	Berkowitz (2001)	Knüppel (2015)
Benchmark	3.4318	0.8049	0.708	0.000	0.001
Linear	2.7125	0.8789	1.9254	0.000	0.000
CKDE	3.2231	0.8051	0.9914	0.000	0.000
NKDE	2.9564	1.2522	0.8617	0.000	0.000
LSCDE	3.0777	0.8041	1.2172	0.000	0.000
MDN	3.5968	0.8045	0.5284	0.81	0.951
KMN	3.5854	0.8042	0.5582	0.767	0.778

This table compares the out-of-sample performance for a range of forward-looking conditional density estimators for the S&P 500. *Linear* represents a model that makes the mean and standard deviation of a normal distribution linearly dependent on the forward-looking variables. The remaining estimators are: Conditional Kernel Density Estimation (CKDE), Neighborhood Kernel Density Estimation (NKDE), Least-Squares Conditional Density Estimation (LSCDE), Mixture Density Network (MDN) and Kernel Mixture Network (KMN). The third and the forth column report the root mean squared errors for the mean and standard deviation prediction. For the methods that involve an optimization (LSCDE, MDN, KMN), we report the error of the average estimator of 100 estimation runs with different starting values.

a given X_{t-1}. Related to this approach, the Neighborhood Conditional Density Estimator (NKDE) differs by only considering observations whose input measure values are close to X_{t-1} when forming the density estimate. The least-squares conditional density estimator (LSCDE) turns the kernel density estimation into a regression task by reducing the number of kernels, fixing their positions and only determining their weight for a given estimate. Finally, we consider another neural network based approach, the Kernel Mixture Network (KMN). A detailed description of all methods and how we perform estimation can be found in appendix A.

Table 6 shows the results of our horse race. The two neural network approaches show the highest average log likelihood and lowest volatility prediction errors. As is well known, returns are nearly not predictable at the daily horizon, but still the 0.8045 (0.8042) RMSE of the MDN (KMN) mean prediction corresponds to an out-of-sample R^2 of 0.1% (0.17%). We pick the MDN over the KMN in our main analysis due to its slightly higher average conditional log likelihood and lower volatility forecasting error. The linear model shows the lowest out-of-sample performance of all analyzed methods. Its unfavorable performance compared to the unconditional benchmark is a sign for over-fitting in this set-up. At the same time, the non-parametric methods also underperform the unconditional benchmark. One reason might lie in the size of our training set. Nonparametric methods typically require large amounts of data to build an expressive estimator. Our training data set contains 2642 data points, which might not be enough for these approaches. Finally, no density estimator beside the neural network estimators passes the Berkowitz (2001) and Knüppel (2015) tests. Put differently, we can reject the null hypothesis that returns are sampled from the respective conditional densities with almost certainty. Only the MDN and the KMN build conditional densities for which we cannot reject this null hypothesis.

Table 7: Euro Stoxx 50 out-of-sample performance of conditional density estimators

Model	Avg. $\log \mathcal{L}$	Mean RMSE	σ RMSE	Berkowitz (2001)	Knüppel (2015)
Benchmark	2.9957	1.2615	1.0686	0.000	0.067
Linear	2.7309	1.2657	1.6516	0.000	0.000
CKDE	2.8856	1.2611	1.2321	0.000	0.000
NKDE	2.6432	1.7439	1.2161	0.000	0.000
LSCDE	2.7906	1.2609	1.4234	0.000	0.000
MDN	3.0924	1.2627	0.8297	0.434	0.966
KMN	3.0883	1.2608	0.8635	0.888	0.901

This table compares the out-of-sample performance for a range of forward-looking conditional density estimators for the Euro Stoxx 50. *Linear* represents a model that makes the mean and standard deviation of a normal distribution linearly dependent on the forward-looking variables. The remaining estimators are: Conditional Kernel Density Estimation (CKDE), Neighborhood Kernel Density Estimation (NKDE), Least-Squares Conditional Density Estimation (LSCDE), Mixture Density Network (MDN) and Kernel Mixture Network (KMN). The third and the forth column report the root mean squared errors for the mean and standard deviation prediction. For the methods that involve an optimization (LSCDE, MDN, KMN), we report the error of the average estimator of 100 estimation runs with different starting values.

It seems like the neural network approaches in general, and the MDN in particular, represent a bridge between the parametric and non-parametric world. They are comparable to nonparametric approaches in their flexibility, but their parametric structure enables us to form expressive density estimators based on relatively small training data sets. Our results suggest that neural network approaches are superior to standard alternatives and in our tests, they were the only approaches that produced a valid forward-looking density estimator.

5.2 International Evidence

It is thinkable that unknown characteristics in the relationship of the S&P 500 and its associated options work in favor of the MDN. For this reason, we repeat the analysis of section 5.1 for the Euro Stoxx 50. The Euro Stoxx 50 represents the leading equity index for the euro area and highly liquid options on the index are traded at the Eurex.

Table 7 contains the results of the performance evaluation for the Euro Stoxx 50. Throughout all methods and performance figures, it appears like the density of Euro Stoxx 50 returns is harder to predict than the density of the S&P 500. However, in our validation data set, daily S&P 500 returns had an average volatility of 0.8%, while the volatility of daily Euro Stoxx 50 returns was at 1.26%. Relative to the benchmark method, the MDN performs very similar as for the S&P 500: The average conditional log likelihood is 3.2% higher than the benchmark, compared to 4.8% for the S&P 500. The Mean RMSE grows by 0.1%, compared to a decrease of 0.05% in the S&P 500. Finally, the RMSE of the volatility forecast is 22.4% below the benchmark, compared to a decrease of 25.4% in the S&P 500. In relative terms, the ordering of the different methods with respect to their performance is very similar to the S&P 500 application. The results of the Berkowitz (2001)

Table 8: Out-of-sample MDN performance with and without Noise Regularization

Noise Regularization	Yes	No
Avg. $\log \mathcal{L}$	3.5936 (0.0094)	3.5968 (0.0094)
Mean RMSE	0.8045 (0.0012)	0.8044 (0.0012)
Mean MAE	0.5566 (0.0017)	0.5566 (0.0017)
σ RMSE	0.5327 (0.0069)	0.5284 (0.0075)
σ MAE	0.4271 (0.0094)	0.4209 (0.0102)

The table shows the effect of noise regularization on out-of-sample performance figures of the MDN estimator.

and Knüppel (2015) tests also draw the same picture as for the S&P 500: The densities of all methods beside the neural network based approached are rejected.

Overall, the results of the Euro Stoxx 50 exercise confirm our previous findings. Note, that our analysis does not state that the mechanics between forward-looking variables and returns are the same for the S&P 500 and the Euro Stoxx 50. Instead, it shows that the MDN is flexible enough to capture general and stable relationships between these variables in both markets.

5.3 Over-Fitting

A central concern when working with neural network approaches is over-fitting. Neural networks are highly flexible with respect to the approximated functional relationship in the training data. It can therefore happen that the trained model traces input-output relationships that existed in the training data only due to randomness. The generalization capabilities of such a model would be severely reduced.

Over-fitting in neural network is mainly driven by the size of the network and expresses itself in a good in-sample, but bad out-of-sample performance. Our estimation methodology is designed to address the over-fitting issue.[9] Still, ensuring that over-fitting does not appear in our forward-looking density estimator is of first order importance. Rothfuss et al. (2019b) propose noise regularization to prevent over-fitting in Mixture Density Networks. In particular, they propose to add small random noise terms to the input and return data during the training phase. Intuitively, the noise slightly blurs the training data, thus making it impossible for the neural network to identify the small random pseudo relationships in the training data that lead to over-fitting. At the same time, fundamental relationships between the inputs and the returns are unaffected as they still hold in expectation. More formally, Rothfuss et al. (2019b) show that noise regularization introduces a curvature penalty term into the objective function of the estimator and thus introduces a tendency to smooth the conditional density estimate.

We re-estimate our forward-looking density estimator with noise regularization to check whether

[9] We determine the network size during the hyper-parameter search. In that, different parts of the training data set are treated as validation set, that is not used for model fitting. Over-fitting increases the error in these validation sets. The algorithm chooses the network size such that the error in the validation sets is minimized, thus counteracting over-fitting.

this additional shield against over-fitting improves the out-of-sample performance. We treat the size of the noise that is added to the training data as a hyper-parameter, which is determined in the hyper-parameter search phase of our training algorithm. Table 8 compares the out-of-sample performance of the MDN estimator with and without noise regularization. None of the performance figures changes notably. In both estimations, the variation in the performance figures due to different starting values is low, thus indicating stable convergence of the estimation. Together with the observation from table 2 that the performance increase of the MDN against the benchmark is nearly the same in the in-sample and the out-of-sample data set, we conclude that over-fitting is highly unlikely for our conditional density estimator.

6 Conclusion

In this study, we presented a forecasting method for the full distribution of returns that is based on Mixture Density Networks. Our estimator is forward-looking as it is purely based on option-implied, risk-neutral expectations about the future, which where measured in a model-free way. The MDN places very little constraints on the statistical return distribution and is agnostic about the stochastic discount factor, that links the risk-neutral and physical return distribution. The approach can therefore be considered as model-free.

We showed that the forward-looking estimator generalizes predicting relationships better than a backward-looking estimator, which is based on past return information. It also outperforms a number of alternative parametric and nonparametric forward-looking return distribution estimators. While the out-of-sample conditional densities of the backward-looking estimator and the forward-looking alternatives are rejected in our statistical tests, this is not the case for the conditional forward-looking return distribution. It is therefore well possible that the MDN accurately recovers the true physical return distribution. The MDN uncovers significant nonlinear relationships between risk-neutral and physical moments. Our results indicate that an exploitation of these nonlinearities in combination with the adaptive smoothness and continuity constraints of our estimation approach is key in forecasting the return distribution.

The inclusion of other types of forward-looking information like analyst forecast (Ulrich et al., 2019) or text measures (Engle et al., 2019) may improve the performance of the estimator even further. Furthermore, while we restrict our analysis to the inspection of equity index returns, the method can easily be translated to other asset classes with associated options.

References

Yacine Aït-Sahalia and Andrew W Lo. Nonparametric Estimation of State-Price Densities Implicit in Financial Asset Prices. *The Journal of Finance*, 53(2):499–547, 1998.

Luca Ambrogioni, Umut Güçlü, Marcel A. J. van Gerven, and Eric Maris. The Kernel Mixture Network: A Nonparametric Method for Conditional Density Estimation of Continuous Random Variables. 2017. URL `http://arxiv.org/abs/1705.07111`.

Francesco Audrino, Robert Huitema, and Markus Ludwig. An Empirical Analysis of the Ross Recovery Theorem. *Working Paper*, 2015. URL `https://ssrn.com/abstract=2433170`.

Gurdip Bakshi and Nikunj Kapadia. Delta-Hedged Gains and the Negative Market Volatility Risk Premium. *The Review of Financial Studies*, 16(2):527–566, 2003.

Gurdip Bakshi, Nikunj Kapadia, and Dilip Madan. Stock Return Characteristics, Skew Laws, and the Differential Pricing of Individual Equity Options. *The Review of Financial Studies*, 16(1): 101–143, 2003.

Gurdip Bakshi, Fousseni Chabi-Yo, and Xiaohui Gao. A Recovery that We Can Trust? Deducing and Testing the Restrictions of the Recovery Theorem. *The Review of Financial Studies*, 31(2): 532–555, 2018.

Giovanni Barone-Adesi and Hakim Dall'O. Is the price kernel monotone? *Working Paper*, 2010. URL `https://ssrn.com/abstract=1539363`.

Giovanni Barone-Adesi, Robert F. Engle, and Loriano Mancini. A GARCH Option Pricing Model with Filtered Historical Simulation. *The Review of Financial Studies*, 21(3):1223–1258, 2008.

Geert Bekaert and Marie Hoerova. The vix, the variance premium and stock market volatility. *Journal of Econometrics*, 183(2):181–192, 2014.

Jeremy Berkowitz. Testing Density Forecasts, With Applications to Risk Management. *Journal of Business & Economic Statistics*, 19(4):465–474, 2001.

Christopher M Bishop. Mixture Density Networks. Technical Report. Aston University, Birmingham., 1994.

Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, NY, 2006.

Robert R. Bliss and Nikolaos Panigirtzoglou. Option-implied risk aversion estimates. *The Journal of Finance*, 59(1):407–446, 2004.

Tim Bollerslev. Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, 1986.

Jaroslav Borovička, Lars Peter Hansen, and José A. Scheinkman. Misspecified Recovery. *The Journal of Finance*, 71(6):2493–2544, 2016.

Thomas Busch, Bent Jesper Christensen, and Morten Ørregaard Nielsen. The role of implied volatility in forecasting future realized volatility and jumps in foreign exchange, stock, and bond markets. *Journal of Econometrics*, 160(1):48–57, 2011.

Peter Carr and Jiming Yu. Risk, Return, and Ross Recovery. *The Journal of Derivatives*, 20(1): 38–59, 2012.

Mikhail Chernov. On the Role of Risk Premia in Volatility Forecasting. *Journal of Business & Economic Statistics*, 25(4):411–426, 2007.

John H. Cochrane. *Asset Pricing: Revised Edition*. Princeton University Press, Princeton and Oxford, 2005.

Horatio Cuesdeanu and Jens Jackwerth. The Pricing Kernel Puzzle in Forward Looking Data. *Review of Derivatives Research*, forthcoming.

Anthony C. Davison and David V. Hinkley. *Bootstrap Methods and their Application*. Cambridge Series on Statistical and Probabilistic Mathematics ; [1]. Cambridge University Press, Cambridge [u.a.], 1997.

Bruno de Finetti. La prévision: ses lois logiques, ses sources subjectives. In *Annales de l'institut Henri Poincaré*, volume 7, pages 1–68, 1937.

Yannick Dillschneider and Raimond Maurer. Functional Ross Recovery: Theoretical Results and Empirical Tests. *Working Paper*, 2018. URL `https://ssrn.com/abstract=2991984`.

Darrell Duffie. *Dynamic Asset Pricing Theory. Third Edition*. Princeton University Press, Princeton and Oxford, 2001.

Christian L. Dunis, Jason Laws, and Georgios Sermpinis. Higher order and recurrent neural architectures for trading the EUR/USD exchange rate. *Quantitative Finance*, 11(4):615–629, 2011.

Robert Engle, Stefano Giglio, Heebum Lee, Bryan Kelly, and Johannes Stroebel. Hedging Climate Change News. *Working Paper*, 2019. URL `http://pages.stern.nyu.edu/~jstroebe/PDF/EGKLS_ClimateRisk.pdf`.

Robert F Engle and Andrew J Patton. What Good is a Volatility Model? In *Forecasting Volatility in the Financial Markets*, pages 47–63. Elsevier, 2007.

Eugene F. Fama and Kenneth R. French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33(1):3–56, 1993.

Myron J Gordon. *The Investment, Financing, and Valuation of the Corporation*. RD Irwin Homewood, IL, 1962.

Shihao Gu, Bryan Kelly, and Dacheng Xui. Empirical Asset Pricing via Machine Learning. *Working Paper*, 2019. URL https://ssrn.com/abstract=3159577.

Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools and Techniques to Build Intelligent Systems*. O'Reilly Media, Sebastopol, CA, 2017.

Lars Peter Hansen and José A. Scheinkman. Long-Term Risk: An Operator Approach. *Econometrica*, 77(1):177–234, 2009.

Lars Peter Hansen and Kenneth J. Singleton. Generalized Instrumental Variables Estimation of Nonlinear Rational Expectations Models. *Econometrica*, 50(5):1269–1286, 1982.

Lars Peter Hansen and Kenneth J. Singleton. Stochastic Consumption, Risk Aversion, and the Temporal Behavior of Asset Returns. *Journal of Political Economy*, 91(2):249–265, 1983.

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4 (2):251–257, 1 1991. ISSN 0893-6080. doi: 10.1016/0893-6080(91)90009-T. URL https://www.sciencedirect.com/science/article/pii/089360809190009T?via%3Dihub.

James M. Hutchinson, Andrew Lo, and Tomaso Poggio. A Nonparametric Approach to Pricing and Hedging Derivative Securities Via Learning Networks. *The Journal of Finance*, 49(3):851–889, 1994.

Wolfgang Härdle. *Applied Nonparametric Regression*. Econometric Society monographs ; 19. Cambridge Univ. Pr., Cambridge [u.a.], 1. paperback ed. edition, 1991.

Jens Carsten Jackwerth. Recovering Risk Aversion from Option Prices and Realized Returns. *The Review of Financial Studies*, 13(2):433–451, 2000.

Jens Carsten Jackwerth. Option-implied risk-neutral distributions and risk aversion. 2004.

Jens Carsten Jackwerth and Marco Menner. Does the Ross Recovery Theorem work Empirically? *Working Paper*, 2018. URL https://ssrn.com/abstract=2960733.

Christian Skov Jensen, David Lando, and Lasse Heje Pedersen. Generalized Recovery. *Journal of Financial Economics*, 133(1):154–174, 2019.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 12 2015. URL http://arxiv.org/abs/1412.6980.

Malte Knüppel. Evaluating the Calibration of Multi-Step-Ahead Density Forecasts Using Raw Moments. *Journal of Business & Economic Statistics*, 33(2):270–281, 2015.

Qi Li and Jeffrey S. Racine. *Nonparametric econometrics : theory and practice*. Princeton University Press, 2007.

Matthew Linn, Sophie Shive, and Tyler Shumway. Pricing Kernel Monotonicity and Conditional Information. *The Review of Financial Studies*, 31(2):493—-531, 2018.

Robert E Lucas Jr. Asset prices in an exchange economy. *Econometrica*, 46:1429–1445, 1978.

Markus Ludwig. Robust Estimation of Shape-Constrained State Price Density Surfaces. *The Journal of Derivatives*, 22(3):56–72, 2015.

Ian Martin. What is the expected return on the market? *The Quarterly Journal of Economics*, 132(1):367–433, 2017.

Rajnish Mehra and Edward C. Prescott. The equity premium: A puzzle. *Journal of Monetary Economics*, 15(2):145–161, 1985.

Robert C. Merton. Lifetime Portfolio Selection Under Uncertainty: The Continuous-time Case. *The Review of Economics and Statistics*, 51(3):247–257, 1969.

Robert C. Merton. Optimum Consumption and Portfolio Rules in a Continuous-Time Model. In *Stochastic Optimization Models in Finance*, pages 621–661. 1975.

Emanuel Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 9 1962. ISSN 0003-4851. doi: 10.1214/aoms/1177704472. URL `http://projecteuclid.org/euclid.aoms/1177704472`.

Andrew J. Patton and Allan Timmermann. Monotonicity in asset returns: New tests with applications to the term structure, the CAPM, and portfolio sorts. *Journal of Financial Economics*, 98:605–625, 2010.

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

Frank P. Ramsey. *The Foundations of Mathematics and Other Logical Essays. With a Pref. By G.E. Moore.* K. Paul, Trench, Truber and Co., London, 1931.

Joshua V. Rosenberg and Robert F. Engle. Empirical pricing kernels. *Journal of Financial Economics*, 64(3):341–372, 2002.

Murray Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27(3):832–837, 9 1956. ISSN 0003-4851. doi: 10.1214/aoms/1177728190. URL `http://projecteuclid.org/euclid.aoms/1177728190`.

Steve Ross. The Recovery Theorem. *The Journal of Finance*, 70(2):615–648, 2015.

Jonas Rothfuss, Fabio Ferreira, Simon Boehm, Simon Walther, Maxim Ulrich, Tamim Asfour, and Andreas Krause. Noise Regularization for Conditional Density Estimation. *Working Paper*, 2019a. URL `https://arxiv.org/abs/1907.08982`.

Jonas Rothfuss, Fabio Ferreira, Simon Walther, and Maxim Ulrich. Conditional Density Estimation with Neural Networks: Best Practices and Benchmarks. *Working Paper*, 2019b. URL `https://arxiv.org/abs/1903.00954`.

Leonard J. Savage. *The Foundations of Statistics*. John Wiley, New York, 1954.

Paul Schneider and Fabio Trojani. (Almost) Model-Free Recovery. *The Journal of Finance*, 74(1): 323–370, 2019.

Jandhyala L Sharma, Mbodja Mougoue, and Ravindra Kamath. Heteroscedasticity in stock market indicator return data: volume versus GARCH effects. *Applied Financial Economics*, 6(4):337–342, 1996.

Robert H. Shumway and David S. Stoffer. *Time Series Analysis and Its Applications: With R Examples*. Springer Texts in Statistics. Springer, Cham, Switzerland, fourth edition edition, 2017.

Masashi Sugiyama and Ichiro Takeuchi. Conditional density estimation via Least-Squares Density Ratio Estimation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9, pages 781–788, 2010. URL `http://machinelearning.wustl.edu/mlpapers/paper_files/AISTATS2010_SugiyamaTSKH010.pdf`.

Maxim Ulrich and Simon Walther. Option-Implied Information: What's the Vol Surface Got to Do With It? *Working Paper*, 2018. URL `https://ssrn.com/abstract=3184767`.

Maxim Ulrich, Stephan Florig, and Christian Wuchte. A Model-Free Term Structure of U.S. Dividend Premiums. *Working Paper*, 2019. URL `https://ssrn.com/abstract=3217096`.

John von Neumann and Oskar Morgenstern. Theory of games and economic behavior, 2nd rev. 1947.

Johan Walden. Recovery with unbounded diffusion processes. *Review of Finance*, 21(4):1403–1444, 2017.

Ivo Welch and Amit Goyal. A Comprehensive Look at The Empirical Performance of Equity Premium Prediction. *The Review of Financial Studies*, 21(4):1455–1508, 2008.

Jingtao Yao, Yili Li, and Chew Lim Tan. Option price forecasting using neural networks. *Omega*, 28(4):455–466, 2000.

Yang Zhao, Charalampos Stasinakis, Georgios Sermpinis, and Yukun Shi. Neural network copula portfolio optimization for exchange traded funds. *Quantitative Finance*, 18(5):761–775, 2018.

A Alternative Physical Density Estimators

In this appendix, we give present the alternative density estimators that we use in our robustness tests. A large range of conditional density estimators are known to the literature. On a high level, one can distinguish between parametric (Shumway and Stoffer, 2017) and non-parametric methods (Li and Racine, 2007). Recently, a third class of modeling approaches emerged which is based on machine-learning techniques (Bishop, 2006; Ambrogioni et al., 2017). Our favored MDN approach also belongs to this class. In essence, these methods are parametric, but they are often so flexible that they can approximate a very large class of alternative parametric models without requiring exogenous information about which exact model to approximate. In a simulation study, Rothfuss et al. (2019b) show the superiority of these methods compared to non-parametric density estimators for distributions that are relevant to financial applications.

In the following, we will give an overview of the methods that we employ in our analysis. To fix notation, let r_t stand for the log return of an asset's price at time t. Let $X_{t-1} = (X_{i,t-1})_{i\in\{1,\dots,N\}}$ stand for a set of predictor variables that can be observed in $t-1$. We are now interested in the conditional density $p(r_t|X_{t-1})$ of r_t.

A.1 Parametric Density

A first approach to specifying $p(r_t|X_{t-1})$ is to pose a parametric structure on the evolution of the asset price and thus return. The well-known class of ARMA-GARCH models (Bollerslev, 1986) is a natural candidate for this task, however, it is by construction backward-looking as it conditions on past return data. Engle and Patton (2007) and Sharma et al. (1996) investigate such time series models that are enriched with exogenous predictor variables. We follow their intuition, but shut down the backward-looking channel by dropping past return information from the model. More precisely, we assume that log-returns are conditionally normally distributed, with mean and variance that are linear in the forward-looking input variables:

$$\begin{aligned} r_t &= \alpha + \beta\, X_{t-1} + \epsilon_t, \qquad \epsilon_t \sim N(0, \sigma_t^2), \\ \sigma_t^2 &= a + b\, X_{t-1} + \nu_t. \end{aligned}$$

Here, X_t is the vector of forward-looking variables and log-returns are distributed according to $r_t \sim N(\alpha + \beta\, X_{t-1}, a + b\, X_{t-1})$. We estimate the parameters in a two-pass estimation. In a first step, we regress the observed returns in the training set on X_{t-1} via OLS, thus obtaining initial estimates for α and β. We square the residuals and regress these squared residuals on X_{t-1} again, which provides us with OLS estimates for a and b. In the second estimation step, we use the previous parameter estimates as starting values for a joint maximum likelihood estimation.

A.2 Non-Parametric Density

Three kernel-based methods represent the non-parametric approach in our analysis. First, we use the kernel density estimator to estimate the unconditional distributions of $(r_t, X_{t-1})^\intercal$ jointly and X_{t-1} alone and form the ratio between these two estimates to obtain the conditional distribution of r_t. Putting more weight on the training data points that are closest to the current $(r_t, X_{t-1})^\intercal$ observation leads to neighborhood kernel density estimation. Finally, least-squared conditional density estimation relaxes the weight constraints that are imposed upon the estimator by the previous methods.

Conditional Kernel Density Estimator

Given a set of training data points $\{x_k\}_{k\in[1,M]}$ the unconditional kernel density estimator for an potential subsequent observation x reads as follows (Rosenblatt, 1956; Parzen, 1962):

$$\hat{p}(x) = \frac{1}{Mh}\sum_{k=1}^{M} K\left(\frac{x - x_k}{h}\right) \tag{14}$$

Kernel density estimation (KDE) can be understood as placing a simple density function $K(\cdot)$ into each data point x_k and forming an equally weighted mixture of the M densities. The difference between x and the training data point x_k is scaled by the bandwidth parameter h. In the case of multivariate variable $x \in \mathbb{R}^J, J > 1$, the density can be estimated as the product of marginal kernel density estimates:

$$\hat{p}(x) = \prod_{j=1}^{J} \hat{p}(x^{(j)}) = \prod_{j=1}^{J} \frac{1}{Mh^{(j)}} \sum_{k=1}^{M} K\left(\frac{x^{(j)} - x_k^{(j)}}{h^{(j)}}\right) \tag{15}$$

In that, $x^{(j)}$ denotes the j-th element of the column vector x and $h^{(j)}$ is the bandwidth corresponding to the j-th dimension. One popular choice of $K(\cdot)$ is the Gaussian kernel

$$K(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}. \tag{16}$$

Other common choices of $K(\cdot)$ are the Epanechnikov and exponential kernels. Provided a continuous kernel function, the estimated PDF in (15) is continuous. Beyond the appropriate choice of $K(\cdot)$, a central challenge is the selection of the bandwidth parameter h which controls the smoothing of the estimated PDF. We determine h by minimizing the Integrated Mean Squared Error (IMSE) using a cross-validation approach, as recommended by Li and Racine (2007).

The non-parametric KDE approach can be extended to the conditional case (Conditional KDE; CKDE). Unconditional KDE can be used to estimate both the joint density $\hat{p}(r_t, X_{t-1})$ and the marginal density $\hat{p}(X_{t-1})$. Then, the conditional density estimate follows as the density ratio

$$\hat{p}(r_t|X_{t-1}) = \frac{\hat{p}(r_t, X_{t-1})}{\hat{p}(X_{t-1})} \tag{17}$$

where both the enumerator and denominator are the sums of Kernel functions as in equation (15).

Neighborhood Conditional Kernel Density Estimation

Similar to kernel density estimation, neighborhood kernel density estimation (NKDE) employs standard kernel density estimation in a local ϵ-neighborhood around a query point x (Sugiyama and Takeuchi, 2010). The method uses kernels in the training data points as well, however, rather than using all past observations, NKDE only considers a local subset of the training samples $\{x_k\}_{k \in \mathcal{K}_{x,\epsilon}}$, where $\mathcal{K}_{x,\epsilon}$ contains all sample indices that fulfill $||x_k - x||_2 \leq \epsilon$.
In order to obtain a conditional density estimate, we again build unconditional density estimates for $\hat{p}(r_t, X_{t-1})$ and $\hat{p}(X_{t-1})$ and apply equation (17). Again, we use a Gaussian kernel function and select both, the bandwidth hyper-parameter h and the neighborhood hyper-parameter ϵ via cross-validation.

Least-Squares Conditional Density Estimation

The Least-Squares Conditional Density Estimation (LSCDE) approach of Sugiyama and Takeuchi (2010) estimates the conditional density as a linear combination of kernel functions $K(\cdot)$

$$\hat{p}_\alpha(r_t|X_{t-1}) \propto \alpha^T K((r_t, X_{t-1})^\intercal) \tag{18}$$

Here, $K((r_t, X_{t-1})^\intercal) = (K_1((r_t, X_{t-1})^\intercal), ..., K_M((r_t, X_{t-1})^\intercal))^T$ are kernel functions. The main difference between LSCDE and the previous kernel methods is the direct estimation of the weights α via regression. Furthermore, the kernel functions are not necessarily bound to the past observations. In principle, any number of kernel functions, that is located anywhere in the domain of $(r_t, X_{t-1})^\intercal$ is possible. Practically, Sugiyama and Takeuchi (2010) advise picking randomly a number of past observations at which kernels are located. This number is typically much smaller than the amount of observations, thus making the estimation of α more robust. The parameters $\alpha \in \mathbb{R}^M$ are then obtained by minimizing the integrated squared error

$$J(\alpha) = \int \int \left((\hat{p}_\alpha(r_t|X_{t-1}) - p(r_t|X_{t-1}))^2 p(X_{t-1}) dX_{t-1} dr_t. \right. \tag{19}$$

Sugiyama and Takeuchi (2010) derive the closed-form solution for α for the case of Gaussian kernels. After having obtained $\alpha^* = \arg\min_\alpha J(\alpha)$, the conditional density of r_t can be computed as follows:

$$\hat{p}_\alpha(r_t|X_{t-1}) = \frac{(\alpha^*)^T K((r_t, X_{t-1})^\intercal)}{\int \left((\alpha^*)^T K((r_t, X_{t-1})^\intercal) dy \right.} \tag{20}$$

The denominator in equation (20) is traceable and can be computed analytically. Hence, neither numerical optimization nor numerical integration is needed for obtaining conditional density estimates with LSCDE. However, three hyper-parameters need to be determined: the bandwidth parameter of the Gaussian kernels, the number of kernel functions to use and a regularization parameter that can be used in the estimation of α^*. As before, we estimate these hyper-parameters

via cross-validation.

A.3 Kernel Mixture Network

Beside the Mixture Density Network, we consider one further neural network density estimator, the Kernel Mixture Network. While MDNs resemble a purely parametric conditional density model, the Kernel Mixture Network (KMN), combines both non-parametric and parametric elements (Ambrogioni et al., 2017). Similar to MDNs, a mixture density model of $\hat{p}(r_t)$ is combined with a neural network which takes the conditional variable X_{t-1} as an input. However, the neural network only controls the weights of the mixture components while the component centers and scales are fixed w.r.t. to X_{t-1}. Figuratively, one can imagine the neural network as choosing between a very large amount of pre-existing kernel functions to build up the final combined density function. As common for non-parametric density estimation, the kernels are placed in each of the training samples or a subset of the samples. For each of the kernel centers, one or multiple bandwidth parameters σ_m are chosen. As for MDNs, we employ Gaussians as mixture components, wherein the scale parameter directly coincides with the standard deviation.
Let M be the number of kernel centers μ_k and S the number of different kernel scales σ_s. The KMN conditional density estimate reads as follows:

$$\hat{p}(r_t|X_{t-1}) = \sum_{k=1}^{M} \sum_{s=1}^{S} \Big(w_{k,s}(X_{t-1};\theta) \mathcal{N}(r_t|\mu_k, \sigma_s^2) \tag{21}$$

In order to form a valid density, the weights $w_{k,s}$ must resemble a multinomial distribution. Hence, the output non-linearity of the neural network is chosen as a softmax function. Ambrogioni et al. (2017) propose to choose the kernel centers μ_k by sub-sampling the training data by recursively removing each point $r_i, i < t$ that is closer than a constant δ to any of its predecessor points. This can be seen as a naive form of clustering which depends on the ordering of the dataset. Instead, we use a well-established clustering method such as K-means for selecting the kernel centers. The scales of the Gaussian kernels can either be fixed or jointly trained with the neural network weights. In practice, considering the scales as trainable parameters consistently improves the performance. Overall, the KMN model is more restrictive than MDN as the locations and scales of the mixture components are fixed during inference and cannot be controlled by the neural network.

B Additional Results

Table 9: Trading based on the backward-looking density estimator

(a) Mean trading summary statistics

	Full data set		Out-of-sample	
Strategy	Buy-and-Hold	Backward-looking	Buy-and-hold	Backward-looking
Mean return	8.48 (5.11)	34.03 (11.02)	8.43 (7.89)	11.91 (14.65)
Mean excess return	7.18 (5.11)	33.16 (11.03)	8.28 (7.9)	11.59 (14.67)
Excess return volatility	18.52	39.94	12.79	23.77
Sharpe Ratio	0.388	0.852	0.647	0.501
N	3303	3303	661	661

(b) Volatility trading summary statistics

	Full data set		Out-of-sample	
Strategy	Buy-and-Hold	Backward-looking	Buy-and-hold	Backward-looking
Mean return	523.86 (53.9)	886.51 (87.17)	753.23 (144.12)	1160.43 (304.51)
Mean excess return	525.15 (53.9)	887.78 (87.17)	753.61 (144.13)	1160.82 (304.51)
Excess return volatility	195.01	315.39	233.24	492.8
Sharpe Ratio	2.69	2.81	3.23	2.36
N	3299	3299	660	660

The table shows annualized daily expected return, excess return, excess return volatility and Sharpe Ratio of the conditional mean and volatility trading strategy in the S&P 500, compared with the buy-and-hold strategy. The forecasts that are used for the strategies come from the backward-looking MDN estimator. The numbers in parentheses are standard errors of the average return estimates. The full data set covers January 2004 to July 2018. The out-of-sample data set covers the last 20% of the full data set and thus starts in November 2014.

Table 10: Trading based on the all information density estimator

(a) Mean trading summary statistics

	Full data set		Out-of-sample	
Strategy	Buy-and-Hold	All information	Buy-and-hold	All information
Mean return	8.48 (5.11)	30.7 (7.1)	8.43 (7.89)	11.65 (12.04)
Mean excess return	7.18 (5.11)	29.88 (7.11)	8.28 (7.9)	11.49 (12.07)
Excess return volatility	18.52	25.75	12.79	19.56
Sharpe Ratio	0.388	1.193	0.647	0.596
N	3303	3303	661	661

(b) Volatility trading summary statistics

	Full data set		Out-of-sample	
Strategy	Buy-and-Hold	All information	Buy-and-hold	All information
Mean return	523.86 (53.9)	778.85 (87.02)	753.23 (144.12)	1366.67 (282.83)
Mean excess return	525.15 (53.9)	780.13 (87.01)	753.61 (144.13)	1367.06 (282.83)
Excess return volatility	195.01	314.83	233.24	457.72
Sharpe Ratio	2.69	2.48	3.23	2.99
N	3299	3299	660	660

The table shows annualized daily expected return, excess return, excess return volatility and Sharpe Ratio of the conditional mean and volatility trading strategy in the S&P 500, compared with the buy-and-hold strategy. The forecasts that are used for the strategies come from the MDN estimator with both, backward- and forward-looking information. The numbers in parentheses are standard errors of the average return estimates. The full data set covers January 2004 to July 2018. The out-of-sample data set covers the last 20% of the full data set and thus starts in November 2014.