REPRINTED FROM

Risk.net

Risk.net October 2024

Quantum cognition machine learning: financial forecasting

Quantum cognition machine learning: financial forecasting

Traditional machine learning methods suffer from the curse of dimensionality. Here, Ryan Samson, Jeffrey Berger, Luca Candelori, Vahagn Kirakosyan, Kharen Musaelian and Dario Villani introduce a novel machine learning approach based on the ideas of quantum cognition, which they call quantum cognition machine learning (QCML). The complexity of QCML scales linearly with the number of inputs, rather than exponentially. The authors demonstrate an application of QCML to forecasting stock returns

The last decade has seen massive improvements in machine learning techniques that hold promise for breakthrough advances in productivity and technology. At the same time, there is a growing understanding that something fundamental is missing in the current framework of generative artificial intelligence and statistical machine learning in general (Dawid & LeCun 2022).

In this paper, we show how a machine learning approach based on the ideas of quantum cognition (see Busemeyer & Bruza (2012) and Musaelian *et al* (2024) and the references therein) is capable of capturing features of a system differently from existing machine learning approaches. In addition, this approach can accommodate a large number of features and concept creation not bound by sharp categories.

Shortcomings of classical machine learning

Whether implicitly or explicitly, machine learning has always been about learning a joint probability distribution of preprocessed features, whose relevance and significance is ascertained by a human understanding of the domain. The fundamental problem with this classical probabilistic approach is that its complexity grows exponentially with the number of features. For example, a probability distribution over N binary features corresponds to a vector of size $2^N - 1$. This also leads to an exponential requirement for the amount of data needed to learn this distribution statistically (Musaelian *et al* 2024).

In finance, given the nonstationarity of financial time series, it is easy to see that even for a small set of features there is insufficient data to model complex joint distributions. For example, imagine you have available for training 10 years of daily data on 3,000 firms, which would amount to approximately 7.8 million observations. If you wish to model the joint distribution of 10 features bucketed into quintiles, you have 5^{10} (approximately 9.8 million) bins, and statistics is clearly not possible.

You could alleviate the dimensionality problem by using coarser bins: 2^{10} bins would allow for statistics, but this model is likely to be far less detailed than desired. Alternatively, in some cases you could expand the amount of training data by extending the sample much further back in time, but this would incorporate data from when markets were substantially different, which again leads to undesirable (less relevant) results. A surprising answer to these problems lies in using quantum, rather than classical, probabilities.

Quantum cognition machine learning

The idea of quantum cognition emerged from the works of Aerts & Aerts (1995), Khrennikov (2006) and Busemeyer & Bruza (2012) (see Pothos & Busemeyer (2022) for a recent survey). In these works, it is posited that a state of mind is formally given by a quantum state – that is, a vector in a Hilbert space – and all questions that can be answered by someone in that state of mind are represented as operators in that Hilbert space.

We demonstrate here how representing data as a vector in a Hilbert space with observables represented by operators (matrices) can lead to a logarithmic reduction in the complexity of representation. This dramatic economy of representation may explain why evolution would select quantum cognition over classical statistical learning. As living creatures, we do not encounter the world as well-structured data relevant to the task at hand. Instead, we are confronted by a barrage of unstructured inputs that need to be made sense of, while focusing on what is important and drawing conclusions by abstracting away what is irrelevant.

Building on these insights, we demonstrate a new practical form of machine learning, which we call quantum cognition machine learning (QCML). Our formulation naturally lends itself to implementation on quantum computing hardware, but it is also easily implementable on classical hardware at lower Hilbert space dimensions. In our formulation, we define an error Hamiltonian as a sum of a loss function for each observable:

$$H(\boldsymbol{x}_t, \mathcal{L}, \{\hat{A}_k\}) = \sum_k \mathcal{L}(\hat{A}_k, \boldsymbol{x}_{t,k}) \tag{1}$$

where $\boldsymbol{x}_t$ is a data vector with K elements and is one of T total data vectors, $\mathcal{L}$ is an arbitrary nonnegative loss function with a Hermitian output, and $\{\hat{A}_k\}$ is a set of Hermitian observable operators that must be learned. We have flexibility in how we choose $\mathcal{L}$ as long as the result is nonnegative and Hermitian; we can choose simple forms inspired by Gaussian loss:

$$H(\boldsymbol{x}_t, \mathcal{L}, \{\hat{A}_k\}) = \sum_k (\hat{A}_k - \boldsymbol{x}_{t,k} \cdot I)^2 \tag{2}$$

or more complex functions for classification, the learning of context or other objectives. This error Hamiltonian depends on the data itself, the parameters that compose the set of observable operators $\{\hat{A}_k\}$, and the choice of loss function $\mathcal{L}$. Model tractability can be improved through careful choice of the parameterisation of our operators, and we have developed several variants of this.

We learn the operators $\{\hat{A}_k\}$ through the formalism of quasi-coherent states (Candelori *et al* 2024; Ishiki 2015; Steinacker 2021). Recall that in quantum mechanics a state is a vector of unit norm in a Hilbert space and is represented in bra-ket notation by a ket $|\psi\rangle$. The inner product of two states $|\psi_1\rangle$, $|\psi_2\rangle$ is represented by a bra-ket $\langle\psi_1|\psi_2\rangle$. The expectation value of a Hermitian operator O on a state $|\psi\rangle$ is denoted by $\langle\psi|\,O\,|\psi\rangle$, representing the expected outcome of the measurement corresponding to O on the state $|\psi\rangle$.

Training these models involves iterative updates to the ground state $|\psi_t\rangle$ of the error Hamiltonian and the observables $\hat{A}_k$ to reduce the ground state energy of H until the desired convergence is reached.[1] The specifics of each of these steps depend on the choice of the loss function and how we parameterise $\hat{A}_k$.

Model training

1. Randomly initialise the parameters of $\{\hat{A}_k\}$
2. Iterate over the data and operators until the desired convergence is reached
   a. Generate $H(\boldsymbol{x}_t, \mathcal{L}, \{\hat{A}_k\})$
   b. Holding $\hat{A}_k$ constant, find the ground state $|\psi_t\rangle$ of $H(\boldsymbol{x}_t, \mathcal{L}, \{\hat{A}_k\})$
   c. Holding $|\psi_t\rangle$ constant, calculate the gradients of $H(\boldsymbol{x}_t, \mathcal{L}, \{\hat{A}_k\})$ with respect to $\hat{A}_k$
   d. Update $\hat{A}_k$ via gradient descent

Having suitably trained a set of operators $\hat{A}_k$, consider an arbitrary set of J inputs x_j, $J \subseteq K$, and the corresponding operators $\hat{A}_j$. After solving for the ground state $|\psi_J\rangle$ of the error Hamiltonian computed over this subset, $H(\boldsymbol{x}_j, \mathcal{L}, \{\hat{A}_j\})$, the expected value of any operator $B \in \hat{A}_k$ is computed as $\langle\psi_J|\,B\,|\psi_J\rangle$. This allows us to forecast any of our K operators based on the quasi-coherent state computed over any subset of the operators.[2]

Quantum cognition machine learning applied to financial forecasting

■ **Overview.** We illustrate how QCML can be used to capture complex joint distributions for financial forecasting, while complementing existing machine learning techniques.[3] In order to build confidence in the QCML framework, we begin with a very simple example involving two classic features: momentum and value. There are many definitions of value; here we use revenues scaled by the market capitalisation ('market cap') (see the 'Data and features' section below for details). Our goal is to train both a QCML model and a simple neural network (NN) to predict returns in excess of the returns when momentum and value are considered linearly, and to compare the output of the two approaches. We then move to a second example incorporating additional well-known features often used in equity forecasting and risk management.

■ **Data and features.** We collect daily data on a dynamic set of roughly 1,500 US firms chosen for having the greatest size, liquidity and maturity. We create the following commonly used features[4] using market data sourced from Bloomberg, accounting data sourced from Standard & Poor's (S&P) Capital IQ and securities lending data sourced from S&P Securities Finance.

- Accruals: sign-flipped four-quarter change of:

$$\text{total assets} - \text{working capital} - \text{total liabilities}$$
$$- \text{long-term investments} + \text{long-term debt}$$

scaled by total assets, de-meaned by Global Industry Classification Standard (GICS) industry code and cross-sectionally normalised.

- Beta: beta estimated with a rolling 252-day time series, relative to the S&P 500 index.
- EBITDA to TEV: prior four quarters' earnings before interest, taxes, depreciation and amortisation (EBITDA), scaled by total enterprise value (TEV), de-meaned by GICS industry code and cross-sectionally normalised.
- Momentum: returns from 21 days ago to 252 days ago, cross-sectionally normalised.
- Operating efficiency: prior four quarters' revenues, scaled by total assets, de-meaned by GICS industry code and cross-sectionally normalised.
- Profit margin: prior four quarters' net income, scaled by prior four quarters' revenues, de-meaned by GICS industry code and cross-sectionally normalised.
- Short utilisation: sign-flipped lender value on loan (averaged with a 10-day half-life), scaled by active lendable value (also averaged with a 10-day half-life), cross-sectionally normalised.
- Size: log of market cap, cross-sectionally normalised.
- Value: prior four quarters' revenues, scaled by market cap, de-meaned by GICS industry code and cross-sectionally normalised.
- GICS dummies: dummy variables based on GICS industry group membership.

■ **Model setup and training.** The models are trained using daily data from January 2008 through August 2013. While both the NN and QCML models can be updated online or through rolling/expanding retraining, for simplicity we keep the parameters for both models static after the initial training.

As our target variable for model training we use the 15-day forward log returns, projected away from model input features as well as beta, size and the GICS dummies.[5] After projection, target returns are cross-sectionally normalised. Although forecasts learned from models using such a residualised

[1] *The ground state $|\psi_t\rangle$ is the eigenstate associated to the lowest eigenvalue of the error Hamiltonian and is also called the quasi-coherent state.*

[2] *Note this also provides a natural way to deal with missing data: as J can be any subset of K, any missing inputs can simply be excluded from the error Hamiltonian calculated for forecasting, rather than being prefilled in some manner. These missing inputs could also themselves be forecast given $|\psi_J\rangle$.*

[3] *We do not make any claims of superiority in these examples but wish to show that the framework is efficacious and unique.*

[4] *Hou* et al *(2020) used their proposed q-factor model to test the robustness of various features proposed in the financial literature, and their paper also serves as an excellent survey of such features.*

[5] *If you wish to solve for portfolio weights that maximise expected returns, with a penalty for expected portfolio variance, while maintaining zero exposure to a set of controls, then for weights w, forecast f, asset variance V, risk aversion μ and controls M, you need to solve for the w that minimises $-w^{\mathrm{T}} f + 0.5\mu w^{\mathrm{T}} V w$ such that $w^{\mathrm{T}} M = 0$. The solution is $w = V^{-1} R f$, where $R = I - M(M^{\mathrm{T}} V^{-1} M)^{-1} M^{\mathrm{T}} V^{-1}$. Thus R is a projection operator that projects away from M, consistent with our desired investment process. Also, Rf is the residual from an inverse-variance-weighted regression of f on M.*

A. Sharpe ratios of returns from the NN and QCML forecasts for the momentum/value models

Period	NN	QCML	Combination
Sep 2013–Jun 2024	0.58	0.69	0.75
Sep 2013–Apr 2017	1.91	0.80	1.61
May 2017–Dec 2020	−0.30	0.73	0.27
Jan 2021–Jun 2024	0.11	0.53	0.33

Strategy portfolios have zero exposure to momentum, value, beta, size and GICS industry groups, and thus include no linear contribution these features make to returns

target variable are likely to have a small average correlation with all the features that the target returns have been projected away from (which includes the input features), we project the final aggregate forecasts away from the relevant input and control features in order to focus on performance after controlling for linear effects.

To create more robust forecasts, we partition stocks into randomised groups of approximately 50, train individual NN and QCML models over each subset, and average forecasts for each stock across 100 different such partitions.

The NN architecture and training approach we use adheres fairly closely to that recommended by Gu *et al* (2020), other than our more complex approach to ensembling. Our NN is implemented in PyTorch and contains three hidden layers of 32, 16 and 8 nodes, respectively, with batch normalisation applied prior to rectified linear unit activation. We use a simple mean square error loss function and train using the Adam optimiser.

Our QCML model uses a Hilbert space dimensionality of 32, and operator representation and training techniques proprietary to Qognitive, Inc., but following the general gradient descent approach outlined in the section titled 'Quantum cognition machine learning', implemented on classical hardware.

Model evaluation. To test performance, we use covariance estimated daily from daily returns[6] to produce Markowitz-optimal investment portfolios, where portfolio weight $w = V^{-1} f$ given covariance V and forecast f.[7] Given that the forecasts have been projected away from input and control features, these investment portfolios will also have no exposure to those features.[8]

Investment portfolios are smoothed over 15 days to proxy for the fact that realistic investment portfolios can only gradually begin trading on new forecasts. We compute daily returns to the smoothed portfolios and analyse portfolio performance from September 2013 through March 2024.

We do not remove transaction costs since these forecasts are presented not as a stand-alone investment strategy but rather as capturing complex attributes of the joint distribution that could be additive to an existing strategy.

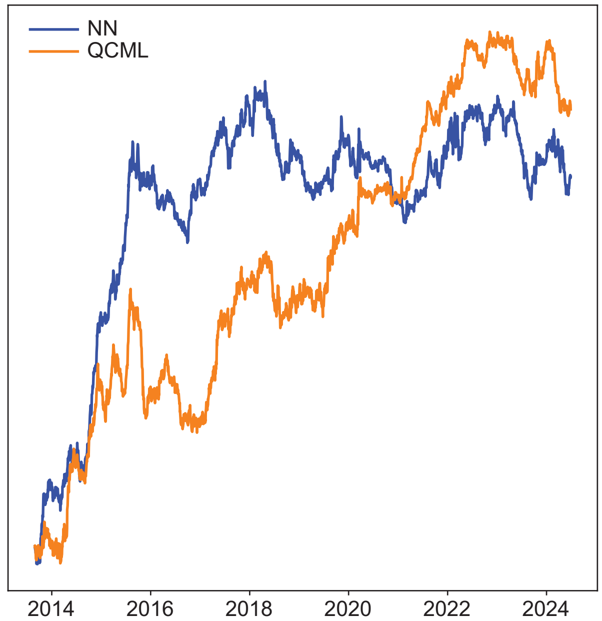
Momentum/value model. For our first example, the model input features are momentum and value, as defined in the section titled 'Data and features'. Thus, the target returns and final forecasts are projected away from momentum, value, beta, size and the GICS dummies.

[6] *Covariance is estimated using a technique proprietary to Duality Group.*

[7] *Note that forecasts here have already been projected away from inputs and controls as explained in the section titled 'Model setup and training' and in footnote 5.*

[8] *Given a matrix of input and control features M, we have $f^{\mathrm{T}} M = 0$ and $w^{\mathrm{T}} M = 0$.*

1 Cumulative returns from the NN and QCML forecasts for the momentum/value models



Strategy portfolios have zero exposure to momentum, value, beta, size and GICS industry groups, and thus include no linear contribution those features make to returns. From a visual examination of the performance, it is clear the models are picking up on similar, but distinct, underlying patterns. The correlation of realised returns is 0.41. Returns have been scaled by full-sample realised volatility

In table A we show the Sharpe ratios of each of the forecasts, as well as the Sharpe ratios of an equal-risk-weighted combination of the two.[9] We find that both the NN and QCML models produce moderately positive signals, which are in excess of the returns when the inputs are considered linearly. However, the correlation of the return streams generated by the two approaches is only 0.41 over the full test period, and we see that an equal-risk-weighted combination of the two approaches outperforms either individual approach.

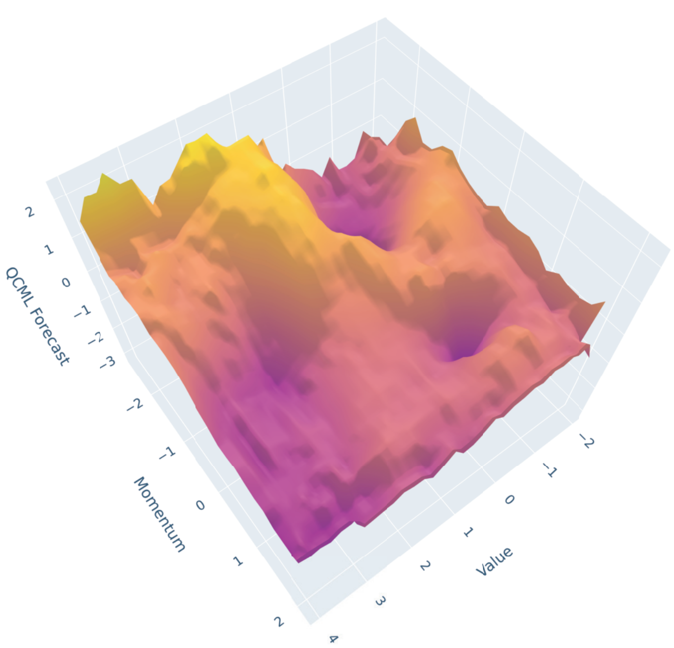
We plot the cumulative returns from the NN and QCML forecasts in figure 1. From a visual examination of the performance, it is clear the models are picking up on similar, but distinct, underlying patterns.

As these simple models have only two inputs and a single forecast, we can also plot surfaces of the forecasts as a function of the inputs, to visually compare what the NN and QCML models learn. We plot the QCML forecast relative to value and momentum in figure 2 and the equivalent for the NN forecast in figure 3.

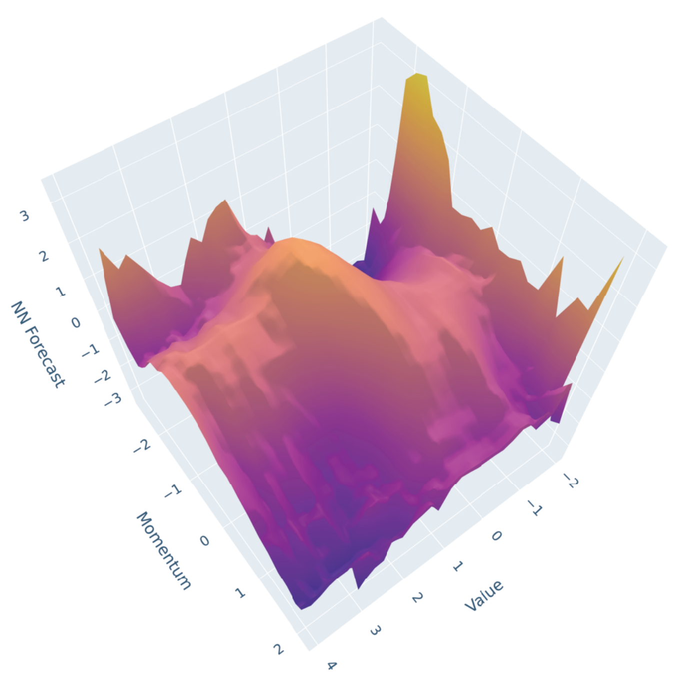
We notice some similarities between the surfaces – both models predict strong returns for stocks with moderately high value and moderately low momentum, and poor returns for those with high momentum and high value or with low momentum and low value. In both cases we could interpret the models as having learned that value is much stronger among low-momentum (loser) stocks, and that the momentum effect is weaker among high-value stocks and stronger among low-value (expensive) stocks.

[9] *The risk scalar for each strategy is the standard deviation over an expanding-window sample of forecast returns, lagged by two days.*

2 The QCML momentum/value model forecast as a function of momentum and value



3 The NN momentum/value model forecast as a function of momentum and value



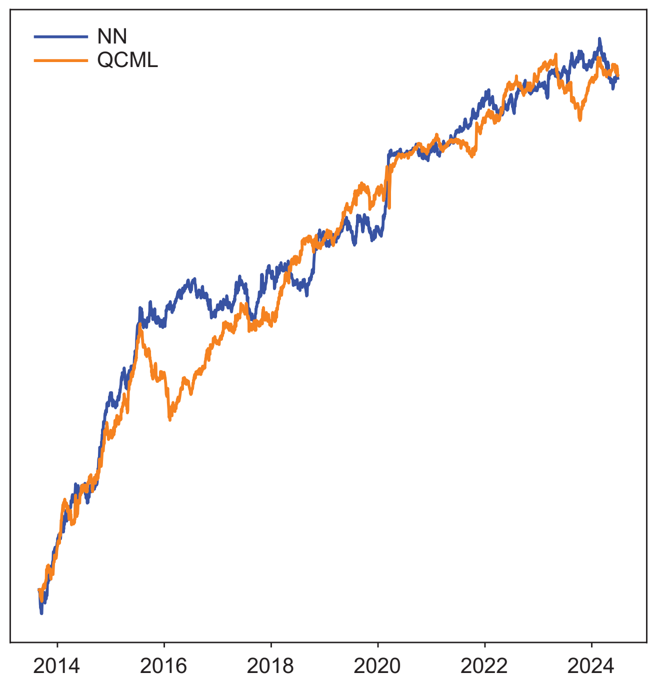
This is consistent with previously documented findings for patterns in the interaction of value and momentum stocks. For example, Asness (1997) documented a similar pattern despite using a different value metric (book value scaled by market cap) and having a data sample independent in time from our training sample.

B. Sharpe ratios of returns from the NN and QCML forecasts for the extended models

Period	NN	QCML	Combination
Sep 2013–Jun 2024	1.25	1.25	1.39
Sep 2013–Apr 2017	1.91	1.72	1.81
May 2017–Dec 2020	1.05	1.26	1.48
Jan 2021–Jun 2024	0.64	0.63	0.78

Strategy portfolios have zero exposure to all input and control features (all features defined in the section titled 'Data and features'), and thus include no linear contribution these features make to returns

4 Cumulative returns from the NN and QCML forecasts for the extended models



Strategy portfolios have zero exposure to all input and control features (all features defined in the section titled 'Data and features'), and thus include no linear contribution those features make to returns. The correlation of realised returns is 0.32. Returns have been scaled by full-sample realised volatility

Of course, the details of the NN and QCML forecast surfaces differ, and both approaches are likely to learn an effect that is noisy relative to the truth. However, the fact that QCML learns something both unique and reasonable when compared with an NN helps to build our confidence in QCML as a forecasting technique.

Extended model. We now extend to the more interesting case of several input features. For these models, in addition to momentum and value, our input features include accruals, EBITDA to TEV, operating efficiency, profit margin, short utilisation and size. As before, the target returns and final forecasts are projected away from inputs and controls, which in this case are all the features defined in the section titled 'Data and features'.

In table B we show the Sharpe ratios of each of the forecasts, as well as the Sharpe ratios of an equal-risk-weighted combination of the two (formed as in the prior section).

In this case, both the NN and QCML models produce strongly positive signals, which are in excess of the returns when the inputs are considered linearly. In addition, the correlation of the return streams generated by the

C. Sharpe ratios of returns from the linear forecasts							
Period	Accruals	EBITDA to TEV	Momentum	Operating efficiency	Profit margin	Short utilisation	Value
Sep 2013–Jun 2024	0.29	0.04	0.65	1.36	0.27	0.93	0.25
Sep 2013–Apr 2017	0.51	0.86	0.96	1.56	0.85	1.90	1.05
May 2017–Dec 2020	−0.30	−1.48	0.47	1.12	−0.24	−0.29	−0.92
Jan 2021–Jun 2024	0.84	1.19	0.53	1.42	0.24	1.30	1.15

Strategy portfolios are projected away from beta, size and the GICS dummies

D. Sharpe ratios of returns from equal-risk combined linear forecasts for momentum and value only, from the same linear forecasts but with either the NN or QCML momentum/value forecasts added, and from all momentum/value strategies				
Period	Linear strategies	Linear strategies + NN	Linear strategies + QCML	All strategies
Sep 2013–Jun 2024	0.91	0.99	1.11	1.07
Sep 2013–Apr 2017	1.84	2.46	1.87	2.25
May 2017–Dec 2020	−0.72	−0.71	0.02	−0.13
Jan 2021–Jun 2024	1.54	0.99	1.42	0.97

The QCML forecast provides better diversification than the NN forecast

E. Sharpe ratios of returns from equal-risk combined linear forecasts for all features, from the same linear forecasts but with either the NN or QCML extended-model forecasts added, and from all extended strategies				
Period	Linear strategies	Linear strategies + NN	Linear strategies + QCML	All strategies
Sep 2013–Jun 2024	1.17	1.38	1.43	1.58
Sep 2013–Apr 2017	2.31	2.57	2.66	2.82
May 2017–Dec 2020	−0.66	−0.32	−0.26	0.03
Jan 2021–Jun 2024	1.77	1.81	1.82	1.83

The QCML forecast provides better diversification than the NN forecast, but the best Sharpe ratio is achieved by using both the NN and QCML forecasts

two approaches is even lower than in the previous example, being 0.32 over the full test period, and we see that an equal-risk-weighted combination of the two approaches outperforms either individual approach.

We plot the cumulative returns from the NN and QCML forecasts in figure 4.

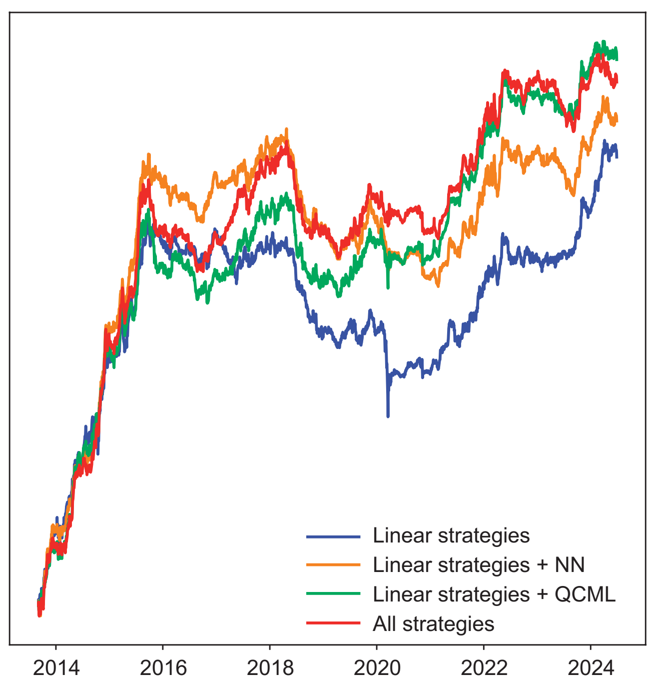
■ **Combining nonlinear and linear models.** We next demonstrate the ability of QCML forecasts to improve on linear forecasts, beyond the improvement seen from NN forecasts. We do this by taking equal-risk-weighted combinations of linear forecasts with the nonlinear NN and QCML forecasts.

For each previously discussed model (the momentum/value and extended models), we produce linear forecasts from all input features except size, which is maintained as a control. We produce our linear forecast by taking the linear features and projecting them away from beta, size and the GICS dummies. This is not a statement regarding which linear features should be included in a forecast, how to best form hedged portfolios of linear features or how to appropriately weight a diverse set of forecasts. It is meant solely to provide a simple demonstration of the ability of QCML forecasts to improve linear models. We show the Sharpe ratios of each of the linear forecasts in table C.

In table D we show the Sharpe ratios of returns from equal-risk combined linear forecasts for momentum and value features only, from the same linear forecasts but with either the NN or QCML momentum/value forecasts added, and from all four strategies (momentum, value, momentum/value NN and momentum/value QCML). Both the NN and QCML forecasts improve the Sharpe ratio of the linear forecasts alone, with a slightly greater improvement seen from adding the QCML forecast.

In table E we show the Sharpe ratios of returns from equal-risk combined linear forecasts for all features, from the same linear forecasts but with either the NN or QCML extended-model forecasts added, and from all nine strategies (the seven linear forecasts, extended NN and extended QCML). Again, both the NN and QCML forecasts improve the Sharpe ratio of the linear forecasts alone, with a slightly greater improvement seen from adding the QCML forecast. Here, the best Sharpe ratio is achieved by using both the NN and QCML forecasts.

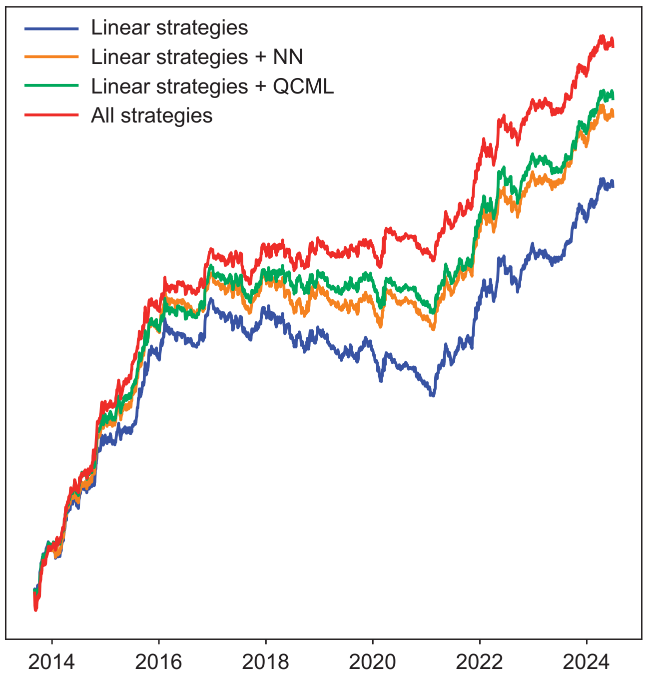
5 Cumulative returns from equal-risk combined linear forecasts for momentum and value only, from the same linear forecasts but with either the NN or QCML momentum/value forecasts added, and from all momentum/value strategies



Returns have been scaled by full-sample realised volatility

We plot the cumulative returns from the linear and enhanced forecasts for the momentum/value models in figure 5, and we plot the cumulative returns from the linear and enhanced forecasts for the extended models in figure 6.

6 Cumulative returns from equal-risk combined linear forecasts for all features, from the same linear forecasts but with either the NN or QCML extended-model forecasts added, and from all extended strategies



Returns have been scaled by full-sample realised volatility

Conclusion

We developed and demonstrated the use of a new machine learning paradigm using principles of quantum cognition, which we call QCML. In a world with an immense proliferation of data sets, the need to address large feature sets paired with small observation sets will become more and more pressing. QCML achieves a logarithmic economy of data representation, making it well suited to meet this challenge – as well as the challenge of nonstationarity, which reduces the amount of relevant data available.

With classic statistical learning, every time you add a variable to a model the uncertainty grows. This leads to difficulty or instability in forming statistical estimates and creates a requirement for judicious choices of model variables. In quantum systems, by contrast, the uncertainty of the whole system can be less than the uncertainty of the components (see Musaelian *et al* (2024) and the references therein).

Here, we demonstrated an application of QCML to forecasting stock returns in excess of the linear contribution to returns of the input features. We benchmarked forecasts produced by QCML against forecasts produced by a neural network and showed that, even for relatively simple systems and a classical hardware implementation, QCML can offer an advantage over or complement the forecasts produced by the neural network.

In future work, we plan to continue studying the properties and applications of QCML. Candelori *et al* (2024) shows one such research direction, where the authors extend QCML to apply it to manifold learning, specifically to the estimation of the intrinsic dimension of data sets, demonstrating the practicality of the proposed method on synthetic manifold benchmarks as well as real data sets. We are also exploring ways to integrate QCML with existing statistical techniques and working towards a practical implementation on quantum hardware. ■

Ryan Samson, Vahagn Kirakosyan and Luca Candelori are directors of research at Qognitive in Miami Beach, Florida, while Kharen Musaelian and Dario Villani are the chief science officer and the chief executive officer, respectively, there. Jeffrey Berger is chief technology officer at Qognitive, based in New York. Candelori is also associate professor in the mathematics department at Wayne State University, and Musaelian and Villani are, in addition to their Qognitive roles, chief investment officer and chief executive officer of the Duality Group, respectively.
Email: ryan.samson@qognitive.io, vahagn.kirakosyan@qognitive.io,
luca.candelori@qognitive.io, kharen@qognitive.io,
dario@qognitive.io, jeff.berger@qognitive.io.

REFERENCES

Aerts D and S Aerts, 1995
Applications of quantum statistics in psychological studies of decision processes
Foundations of Science 1(1), pages 85–97

Asness CS, 1997
The interaction of value and momentum strategies
Financial Analysts Journal 53(2), pages 29–36

Busemeyer J and PD Bruza, 2012
Quantum Models of Cognition and Decision
Cambridge University Press

Candelori L et al, 2024
Robust estimation of the intrinsic dimension of data sets with quantum cognition machine learning
Preprint, available at https://arxiv.org/pdf/2409.12805

Dawid A and Y LeCun, 2022
Introduction to latent variable energy-based models: a path towards autonomous machine intelligence
Les Houches Summer School Lecture Notes

Gu S, B Kelly and D Xiu, 2020
Empirical asset pricing via machine learning
Review of Financial Studies 33(5), pages 2,223–2,273

Hou K, C Xue and L Zhang, 2020
Replicating anomalies
Review of Financial Studies 33(5), pages 2,019–2,133

Ishiki G, 2015
Matrix geometry and coherent states
Physical Review D 92(4), article 046009

Khrennikov A, 2006
Quantum-like brain: 'interference of minds'
Biosystems 84(3), pages 225–241

Musaelian K et al, 2024
Quantum cognition machine learning: AI needs quantum
White Paper, www.qognitive.io/QCML%20-%20Qognitive,%20Inc.pdf

Pothos EM and JR Busemeyer, 2022
Quantum cognition
Annual Review of Psychology 73(1), pages 749–778

Steinacker HC, 2021
Quantum (matrix) geometry and quasi-coherent states
Journal of Physics A: Mathematical and Theoretical 54(5), article 055401