Tactical Asset Allocation with Neural Networks

Semester project

Eder, Ennser, Hartman

25/01/2024

Agenda

Problem Statement

WUTIS

A classical portfolio optimization problem revisited by (advanced) ML algorithms

WHAT

We want to

- Invest in equities, fixed income assets, REITS, gold and cash,
- **and trade in and out positions to take advantage of short to medium term market movements (TAA)**
- in such a way that we generate **alpha**

WHY

The issue with Asset Allocation strategies…

- **Estimation errors** in forecasting expected returns used in the construction of an investment portfolio
- When **optimized,** portfolios tend to **amplify these estimation errors**
- Common **outperformance** of optimized portfolio allocations by **naïve strategies** (equal weights)
- For Tactical Asset Allocation strategies in particular:
  Higher trading costs due to frequent trading

HOW

Suggested Solution

- Model the optimal asset allocation problem with a Neural Network
- Train the model by having it achieve optimal portfolio weights (MSR, MVP, RP, Max R)
- Compare with portfolio **benchmark** index and naïve allocation strategy

Underlying paper: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3812609

Asset classes

W U T I S

A broad variety of assets enables exploitation of (almost) any market phase

	Expansionary	Contractionary
Equities: S&P 500	▪ General appreciating valuation levels, as investors expect growth in dividends and corporate profits	▪ General tendency to decline during economic contractions
Fixed Income: US Treasury Bond	▪ Interest raises lead to bond price declines ▪ Provide stability and income	▪ The safety of government bonds lead to price appreciation ▪ Lower interest rates also benefit debitors
REIT: US REIT Index	▪ Often accompanied by increased demand for commercial and residential properties	▪ Can be strongly impacted by economic downturns ▪ Still provide stable incomes
Gold	▪ Stable addition to every portfolio	▪ Perceived as hedge against inflation
Cash	▪ No real advantages, as no returns provided	▪ Valuable asset during market downturn as provides liquidity to purchase assets at lower prices

Macro indicators

Using various indicators with both similar and contrary impact on the economy makes it more interesting for the model

Offensive (positive correlation)

- US GDP Growth
- US Consumer Price Index
- Consumer Confidence Index
- US Unemployment Rate

10y to 2y US Treasury Yield Spread

- Represents yield curve
- If negative → inverted yield curve → potential recession

- Copper-Gold Ratio
- Lumber Price
- Oil

US High Yield Spreads

- **Rise** indicates increased **risk aversion** and **economic uncertainty → recession**

- Consumer Savings of Disposable Income
- S&P 500 P/E Ratios

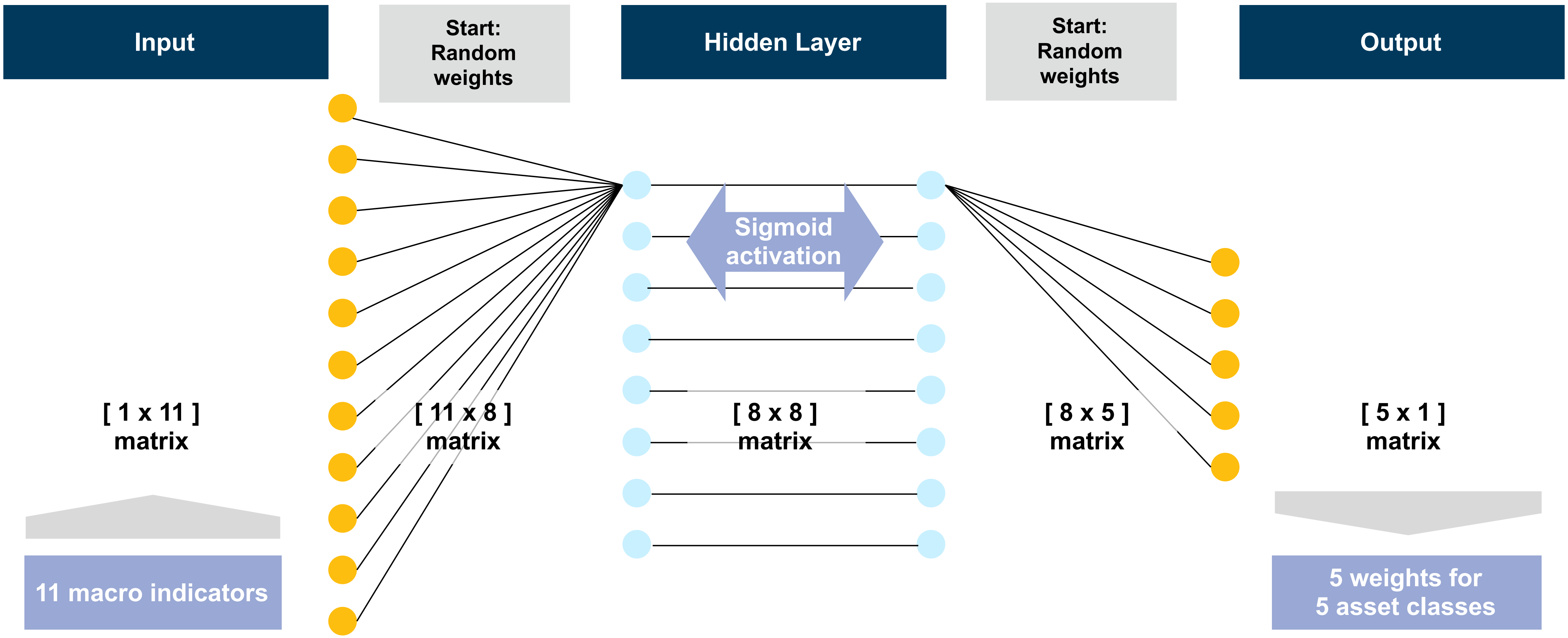
Defensive (negative correlation)

Legend

- Equities: S&P 500
- Fixed Income: US Treasury Bond
- REIT: US REIT Index
- Gold
- Cash

Setup of the Neural Network

This simple yet powerful setup is not dependent on any premade ML/DL libraries



Mechanics and Training

The sigmoid function helps the NN better understand how macro indicators and optimal portfolio weights fit together

Calculations

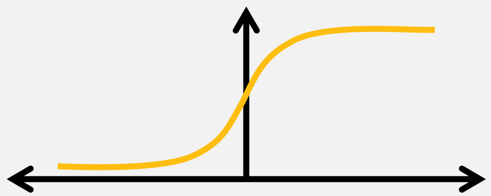
- Mathematical function that **maps** inputs to a value **between 0 and 1,** making it useful for **binary classification** and **logistic regression** problems
- **In NN**, it is used as an **activation function** in the **neurons**, introducing **non-linearity** into model
- **Allows the NN to learn more complex decision boundaries**

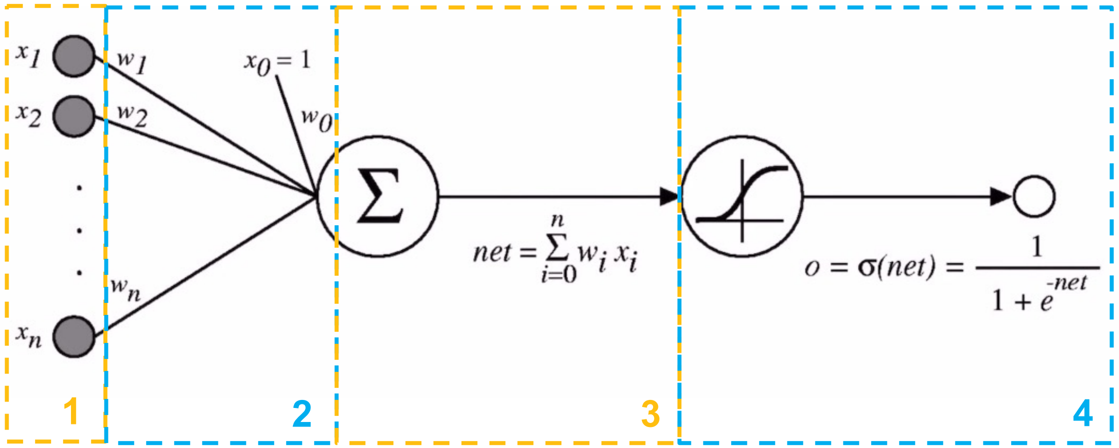
Sigmoid Function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Calculation of values:

$$a_{l+1} = \sigma(W_l a_l + b_l)$$





Training

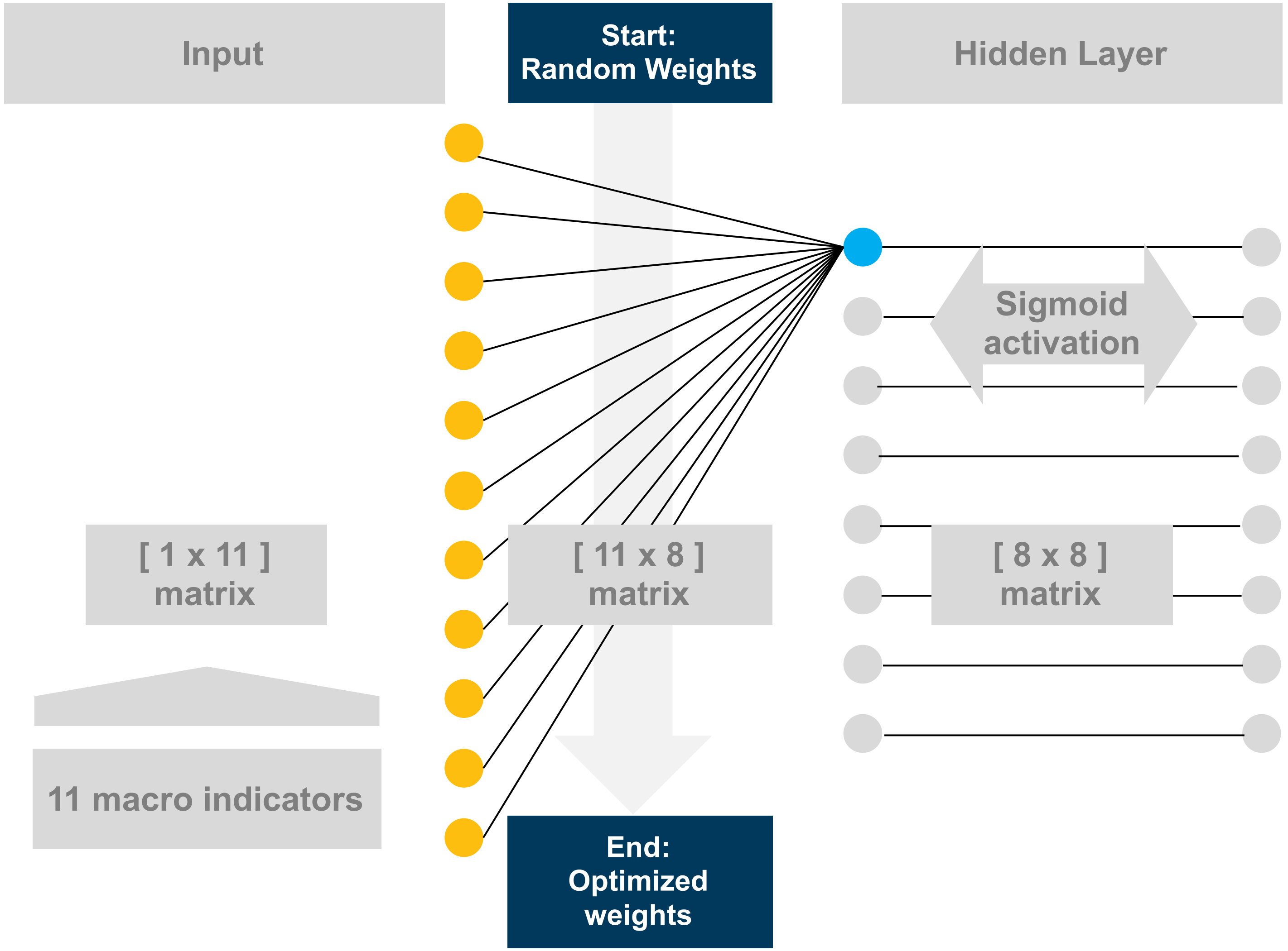
- **Optimal portfolio weights** are calculated on a **rolling window** given a **strategy** (Max Sharpe, Max Profit, Equal Weights, Risk Parity)
- These weights are fed to the Neural Network as **targets to optimize** for
- For each window in the timeframe, the Network takes the selected macroeconomic indicators as an input to optimize the weight targets using **back propagation**
- The result are **two weight matrices** that represent the **mapping** of input values to the optimized output values

Neural Network Predictions: One step

- (1) The **macroeconomic indicators** $x_1, x_2, \ldots, x_n$ for one **corresponding** day are sorted and parsed to the model in a matrix
- (2) This input matrix is **multiplied** by the corresponding **weight matrix**.
- (3) The result is a matrix of the **weighted sums**. Each element in this matrix represents the **aggregated impact** of the input indicators, considering for the **significance assigned by the weights**
- (4) These weighted sums are then traversed through the **activation function**, in our case the sigmoid function.

Backpropagation

Applying advanced concepts of linear algebra and calculus, optimal weights are derived in a back-to-front approach



Backpropagation

Error computation:

- Calculate the difference between the predicted output and the actual targets given to the model.

$$\delta_1^m = \hat{y}_d - y_d$$

Backward pass:

- Propagate this error backward through the Neural Network using the sigmoid derivative to adjust the weights to minimize the prediction error.

$$g'(x) = \frac{\partial \sigma(x)}{\partial x} = \sigma(x)(1 - \sigma(x))$$

Weight Updating:

- Utilizing the gradient computed, the weights are updated systematically to enhance the network's ability to make more accurate predictions over time

$$w_{i,j} = w_{i,j} - \alpha \frac{\partial E}{\partial g_j} \sigma'(g_j)$$

Optimization

W U T I S

22,500 iterations is what it took our NN to find the most ideal parameters

Optimized variables

Learning Rate

- Value between 0 and 1
- The **higher** the rate, **the faster the network learns** from previous iterations
- The **higher** the rate, **the lower is generalization capability**

Weight thresholds

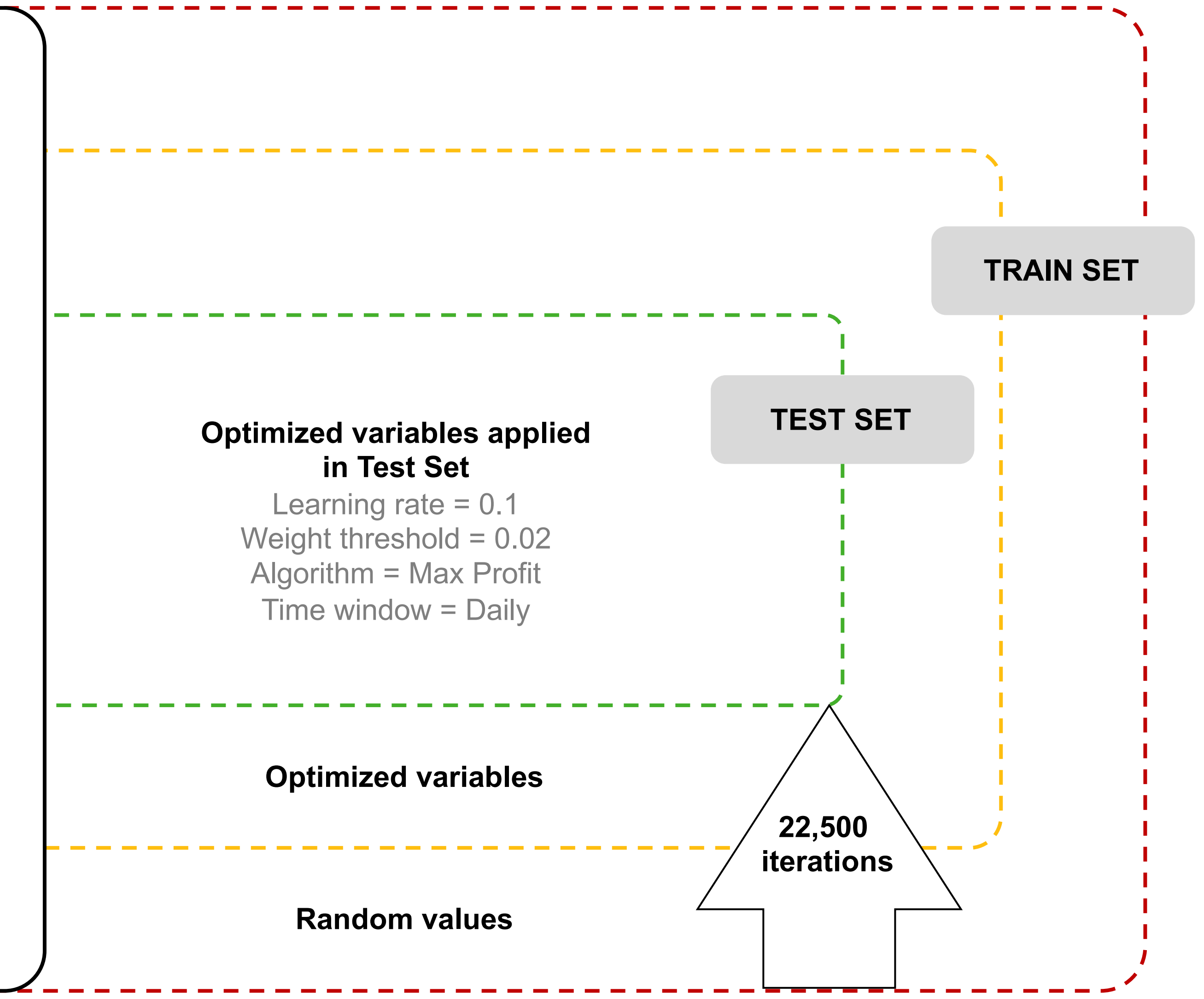
- Typically, in TAA, the portfolio is **spread across all asset classes**
- To ensure the model **does not prefer 100% equity,** different thresholds for **weight bounds** are defined

Portfolio optimization algo

- Determines the **target weights** pursued in **training: MSR, GMV, RP, EW, Max Profit**

Time window

- Frequency of rebalancing and predicting
  - Daily (3d, 5d)
  - Monthly
  - Quarterly



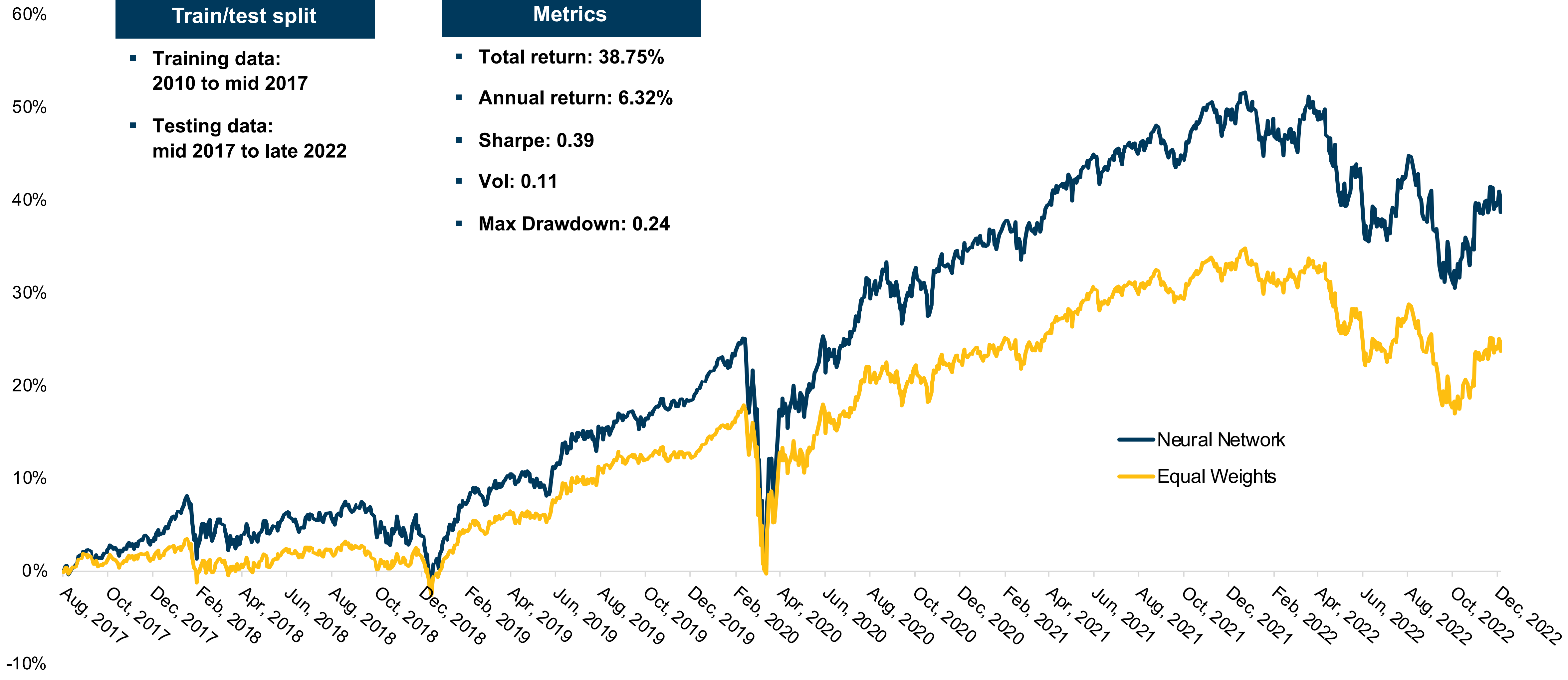
Model Performance

Train/test split

- **Training data:**
  2010 to mid 2017
- **Testing data:**
  mid 2017 to late 2022

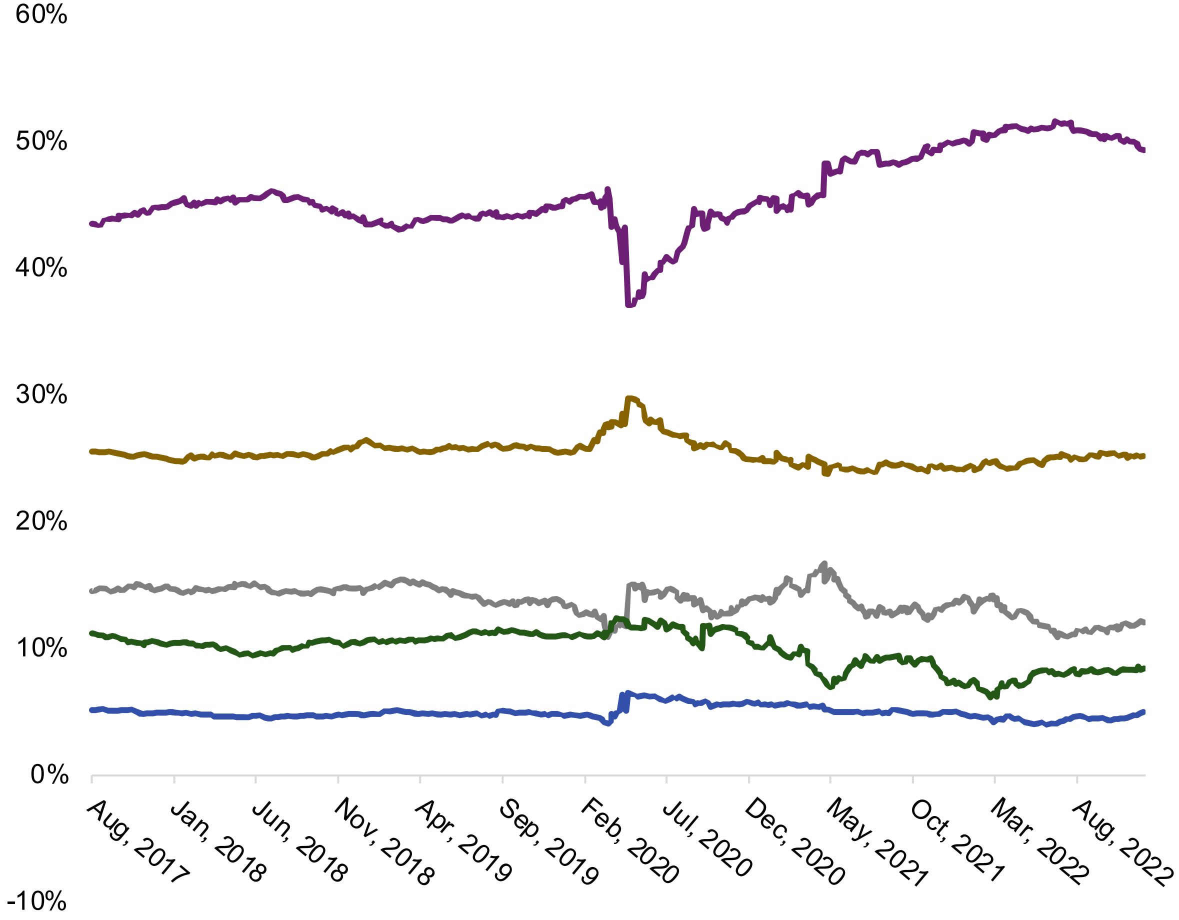
Metrics

- **Total return: 38.75%**
- **Annual return: 6.32%**
- **Sharpe: 0.39**
- **Vol: 0.11**
- **Max Drawdown: 0.24**



Interpretation

The NN seems to have (partially) understood the impact of some of the most common indicators on the market



Asset	Macro indicators (standardized)	|ρ|
Equities: S&P 500	US CPI	0.94
	WTI Oil	0.86
	US GDP	0.73
Gold	Copper-Gold-Ratio	0.83
	High-Yield Spread	0.77
	WTI Oil	-0.68
REIT: US REIT Index	Consumer Confidence	0.63
	US CPI	-0.64
Cash	US CPI	-0.87
	WTI Oil	-0.83
Fixed Income: US Treasury Bond	US Unemployment	0.86
	Consumer Savings	0.84
	WTI Oil	-0.67