Portfolio Optimization with Physics-Inspired Graph Neural Networks

Vertiefungsprojekt II (FS23), Master of Eng. in Data Science
Student: Roman Loop
Academic Supervisor: Shao Jü Woo
Project duration: 10. March 2022 to 14. July 2023

Agenda

- Project goals
- Methodology
  - Idea of «Combinatorial Optimization with Physics-Inspired GNN's»
  - Graph Neural Networks (GNN)
- Physics-Inspired GNN approach for low-risk portfolio optimization
  - Idea & Hyptohesis
  - Dataset
  - Methodology
- Evaluation & Conclusion
- Questions & Discussion

Project goals

- Apply the Schuetz et. al approach to a portfolio optimization use case and **find the maximum set of independent stocks**.
- Create a portfolio of indepentent stocks, backtest its performance, and compare it to benchmark portfolios.

3

Methodology

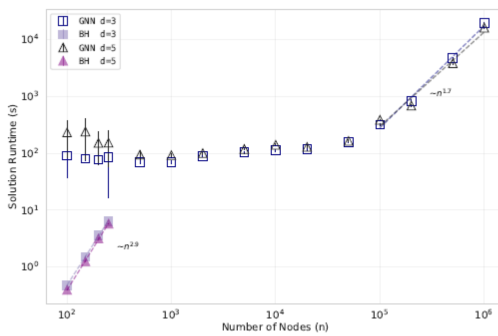
- Idea of «Combinatorial Optimization with Physics-Inspired GNN's»
- Graph Neural Networks (GNN)

4

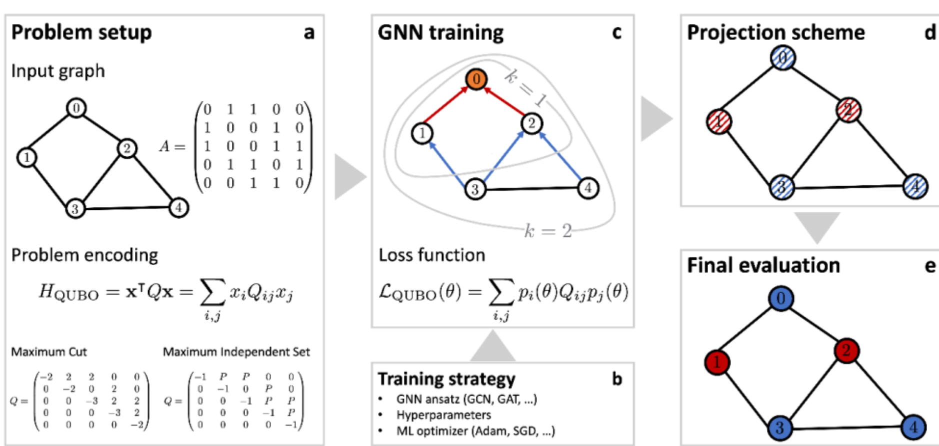
Combinatorial Optimization with Physics-Inspired Graph Neural Networks

Motivation -> Scalability

- Soution runtime ~n$^{1.7}$ vs. ~n$^{2.9}$



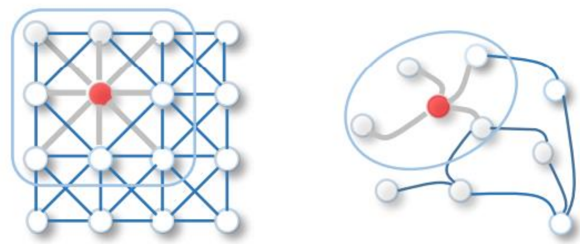
Combinatorial Optimization with Physics-Inspired Graph Neural Networks



End-to-end process from Schuetz et al.
Source : https://arxiv.org/pdf/2107.01188.pdf

GNN – Message Passing

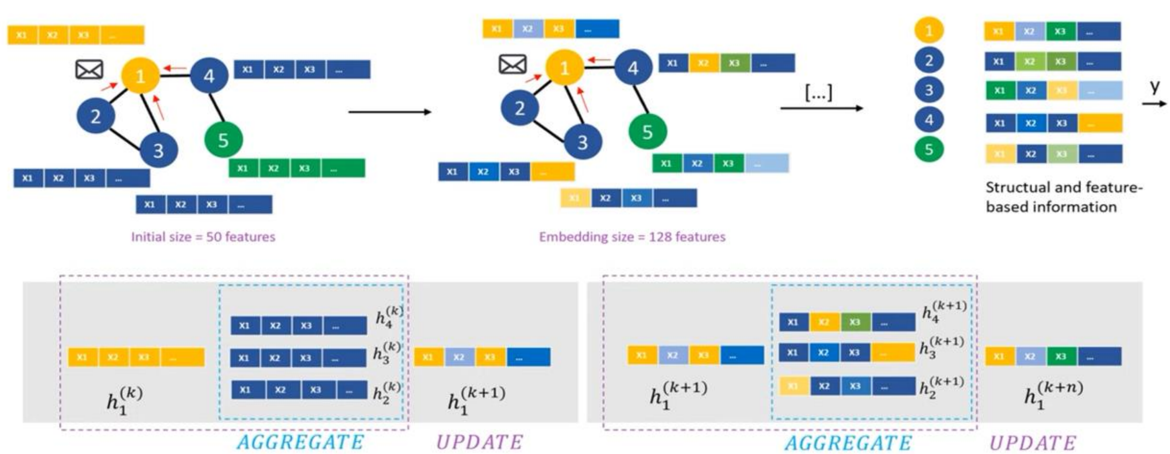
- Message passing layers or MPL build the core of every GNN
- Similar to image convolutions
- Process:
  - Gather information about the neighbourhood of a node
  - Aggregate this information
  - Update the current node embedding with the new information



Source: Wu et al., 2019, A comprehensive Survey on Graph Neural Networks, https://arxiv.org/pdf/1901.00596.pdf

7

Message Passing Layer



Source: DeepFinr, Understanding Graph Neural Networks | Part 2/3, Youtube

8

Physics-Inspired approach for low-risk portfolio optimization

- Hypothesis & Scope
- Dataset
- Data-Preprocessing
- GNN setup & training

Hypothesis & Scope

- **Hypothesis**:
  - A portfolio of uncorrelated stocks is less volatile and therefore reduces the overall risk.
- **Scope**:
  - Asset Universe: S&P500 stocks
- **Problem**:
  - Finding a MIS is NP-hard
  - Existing approximation algorithms (Boppana-Halldorsson) do not scale well – estimated solution runtime ~$n^{2.9}$

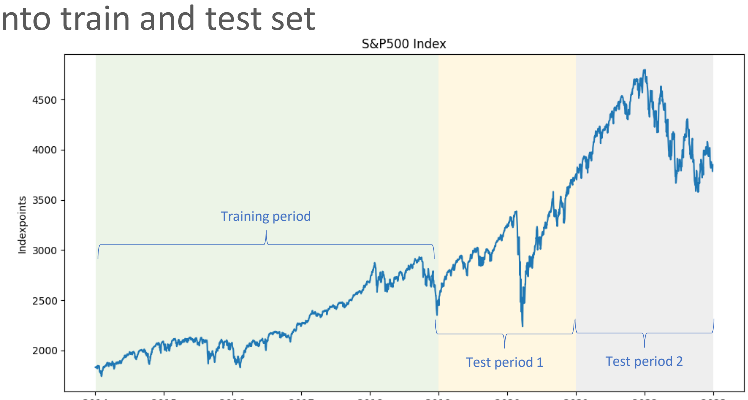
Dataset

- Asset universe: stocks of the S&P500 Index
- Historical prices from 2014 to 2023

The S&P500 is a stock market index tracking the returns of the 500 largest companies listed on US stock exchanges. The index includes about 80% of the US equity market by capitalization and is one of the most widely known stock indices. It includes some of the largest companies in the world such as Apple, Microsoft, Amazon, Johnson & Johnson, ExxonMobil and many others. A full list of the stocks included in the S&P500 can be found on Wikipedia.

Step 1 (Train / Test Split)

- Transform the price time series into returns
- Split into train and test set



Step 2 (Compute the correlation matrices)

- **Pearson's** correlation coefficient
  - Widely used, measures linear correaltion
  - Mathematically defined as "the covariance between two vectors normalized by the product of their standard deviations"
- **Distance** correlation coefficient
  - Measures nonlinear correlations
  - Rather than assessing how two variables tend to covary in their distance from their respective means, the distance correlation assesses how they tend to covary in their distances from all other points
- **Quantile** correlation coefficent
  - Measures of the overall sensitivity of a conditional quantile of a random variable
    to changes in the other variable
  - Geometric mean of two quantile regression slopes

Step 3 (Binarize the correlation matrix)

Binarization threshold = 0.35

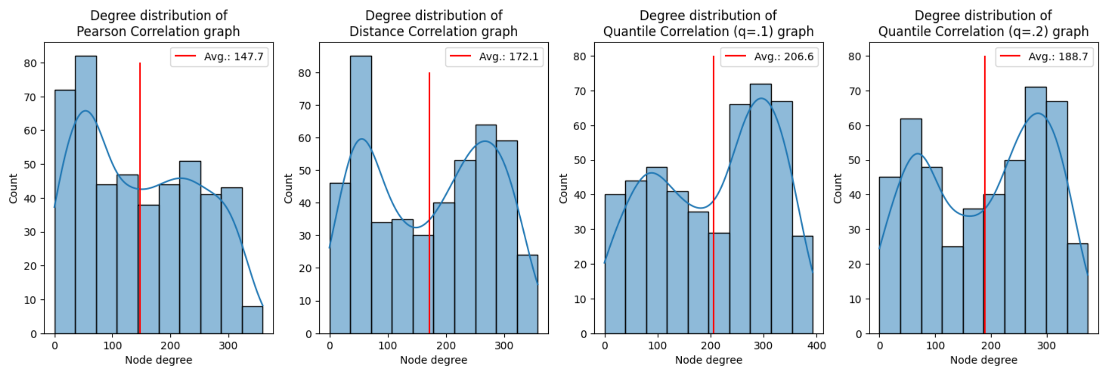
	AAPL	ADBE	...	XEL	ZTS
AAPL	1.000	0.406	...	0.212	0.298
ADBE	0.406	1.000	...	0.293	0.363
...	...	...	...	...	...
XEL	0.212	0.293	...	1.000	0.141
ZTS	0.298	0.363	...	0.141	1.000

	AAPL	ADBE	...	XEL	ZTS
AAPL	1	1	...	0	0
ADBE	1	1	...	0	1
...	...	...	...	...	...
XEL	0	0	...	1	0
ZTS	0	1	...	0	1

4x

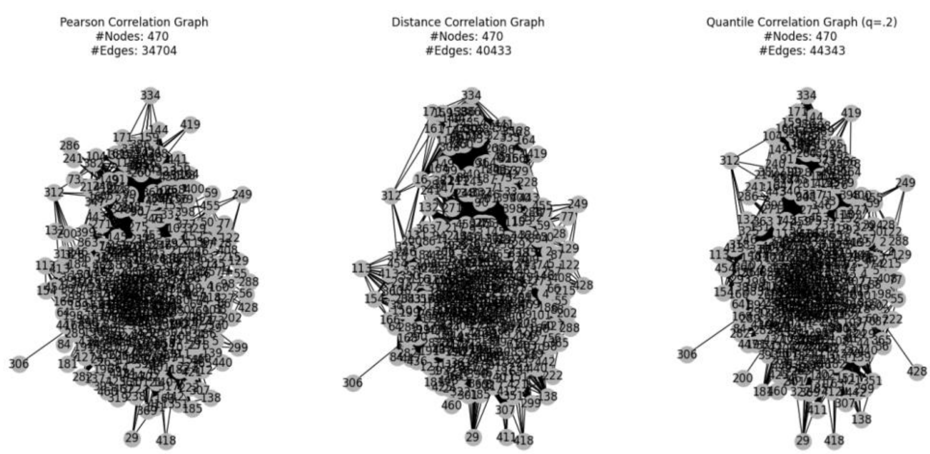
Step 4 (Generate the graphs)

- **NetworkX library**
- **30-50 times higher Ø node degrees**



16

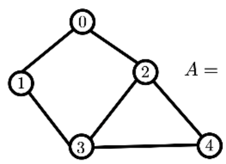
Step 4 (Generate the graphs)



17

Step 4 (Generate QUBO matrix)

Input graph



$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Problem encoding

$$H_{\text{QUBO}} = \mathbf{x}^{\intercal} Q \mathbf{x} = \sum_{i,j} x_i Q_{ij} x_j$$

Maximum Cut

$$Q = \begin{pmatrix} -2 & 2 & 2 & 0 & 0 \\ 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & -3 & 2 & 2 \\ 0 & 0 & 0 & -3 & 2 \\ 0 & 0 & 0 & 0 & -2 \end{pmatrix}$$

Maximum Independent Set

$$\boxed{Q = \begin{pmatrix} -1 & P & P & 0 & 0 \\ 0 & -1 & 0 & P & 0 \\ 0 & 0 & -1 & P & P \\ 0 & 0 & 0 & -1 & P \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix}}$$

- Problem encoding in *Q* marix
- Used the proposed *Q* matrix by the authors
  - rewards -1
  - penalty 2
- *«Handcrafted» Q* matrix uses Sortino ratios as rewards

GNN setup

- **Problem**: Proposed model by Schuetz et. al (two-layer GraphConv architecture, ReLU) did not work for my graph structures.
  - Graphs have **30-50** times higher average node degrees!
- Testing new architectures (SAGE, Graph-Attention, GraphConv)
  - Grid search with Ray Tune
  - 60 combinations of architectures and hyperparameter sets tested
  - Used heuristics from Schuetz et al. for dimension sizing
    - Large graph ($n > 10^5$): $d_0 = int(\sqrt{n})$
    - Small graph ($n < 10^5$): $d_0 = int(\sqrt[3]{n})$
    - $d_1 = int(d_0/2)$

GNN setup

Run time	Best Loss	MIS Size	# Violations	dropout-rate	learning-rate	Model / Architecture
1’074.99	-38.98	39	0	0.05	0.0001	SAGE_2L_Model
1’151.59	-37.00	37	0	0.05	0.0010	SAGE_2L_Model
2’064.49	-36.00	36	0	0.05	0.0010	GAT_1L_2H_Model
1’159.50	-35.00	35	0	0.10	0.0010	SAGE_2L_Model
648.04	-33.00	33	0	0.10	0.0010	SAGE_1L_Model
56.58	0.00	0	0	0.00	0.0010	GCN_2L_Model
166.25	0.00	0	0	0.00	0.0001	GCN_2L_Model
18.03	0.00	0	0	0.10	0.0001	SAGE_2L_Model
176.96	0.00	0	0	0.00	0.0010	GAT_2L_1H_Model
500.55	0.00	0	0	0.00	0.0001	GAT_2L_1H_Model

Our Model

Model Schuetz et al.

14.07.2023 VP2: Portfolio Optimization with Physics-Inspired Graph Neural Networks 20

20

GNN setup

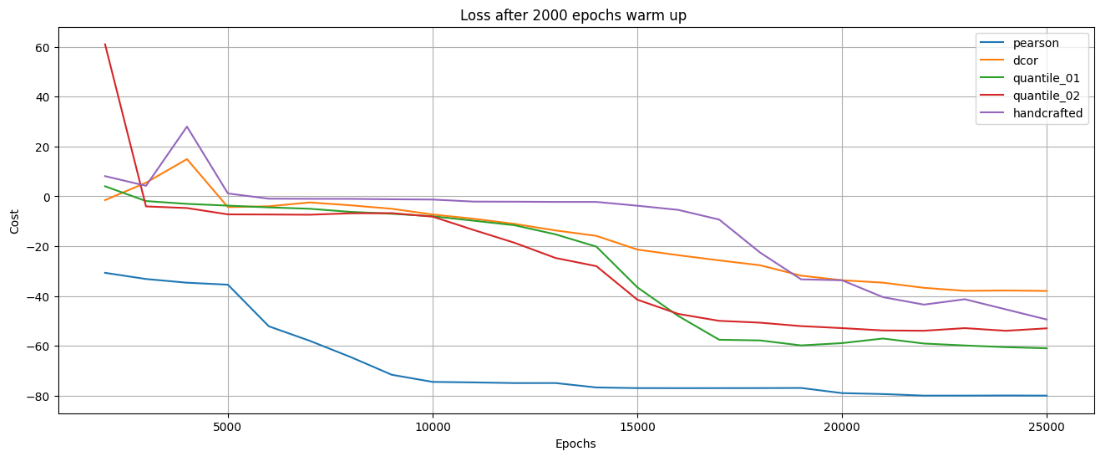
- Final Model: two-layer SAGEConv

Model architecture	Hyperparameters
SAGE_2L_Model((conv1): SAGEConv((feat_drop): Dropout(p=0.0, inplace=False) (fc_pool): Linear(in_features=22, out_features=22, bias=True) (fc_neigh): Linear(in_features=22, out_features=11, bias=False) (fc_self): Linear(in_features=22, out_features=11, bias=True)) (conv2): SAGEConv((feat_drop): Dropout(p=0.0, inplace=False) (fc_pool): Linear(in_features=11, out_features=11, bias=True) (fc_neigh): Linear(in_features=11, out_features=1, bias=False) (fc_self): Linear(in_features=11, out_features=1, bias=True)))	{'lr': 0.0001, 'dim_embedding': 22, 'hidden_dim': 11, 'dropout': 0.05, 'number_classes': 1, 'prob_threshold': 0.5, 'number_epochs': 25000, 'tolerance': 0.0001, 'patience': 1000, 'model': 'SAGE_2L_Model'}

14.07.2023 VP2: Portfolio Optimization with Physics-Inspired Graph Neural Networks 21

21

GNN training



22

Model Evaluation

- GNN approximated MIS
- Backtest optimized portfolio

23

GNN approximated MIS

- Runtime
  - Avg. GNN training time = 230 seconds
  - Avg. runtime of traditional solver = 4.5 seconds
- Approximated MIS:

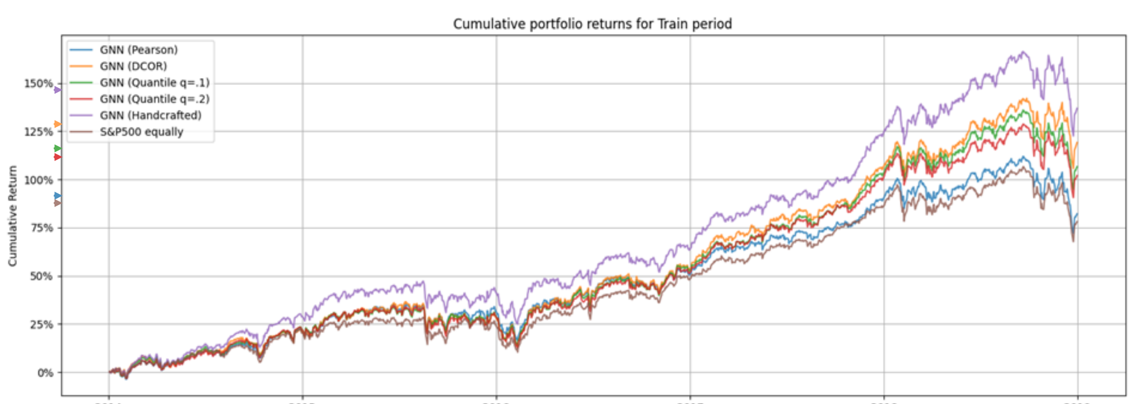
	GNN			Boppana-Halldorsson		
	MIS size	Violations	Difference	MIS size	Violations	Difference
Pearson graph	80	0	+6	74	0	-6
DCOR graph	38	0	-10	48	0	+10
Quantile (q=.1) graph	61	0	+2	59	0	-2
Quantile (q=.2) graph	59	0	+2	57	0	-2

=> It is possible to find good MIS approximations with a GNN

25

Backtest optimized portfolio

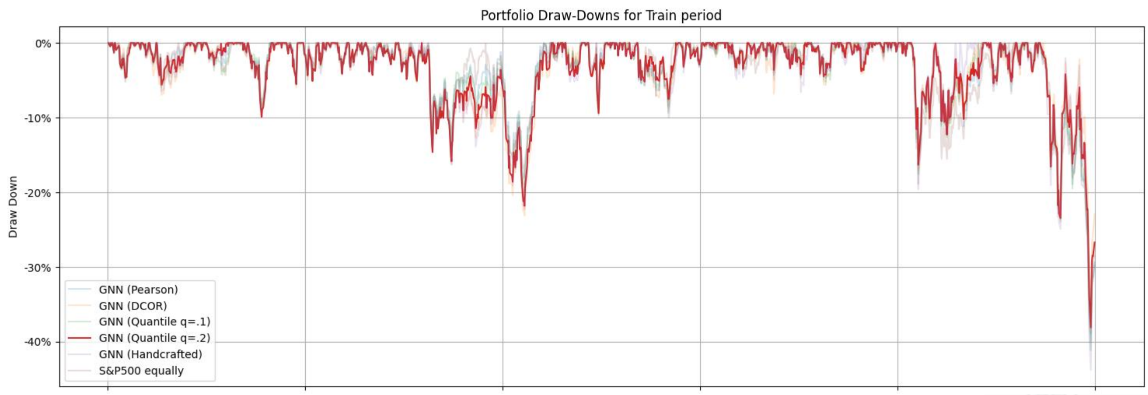
- **Returns** during **training** period



28

Backtest optimized portfolio

- **Drawdowns** during **training** period

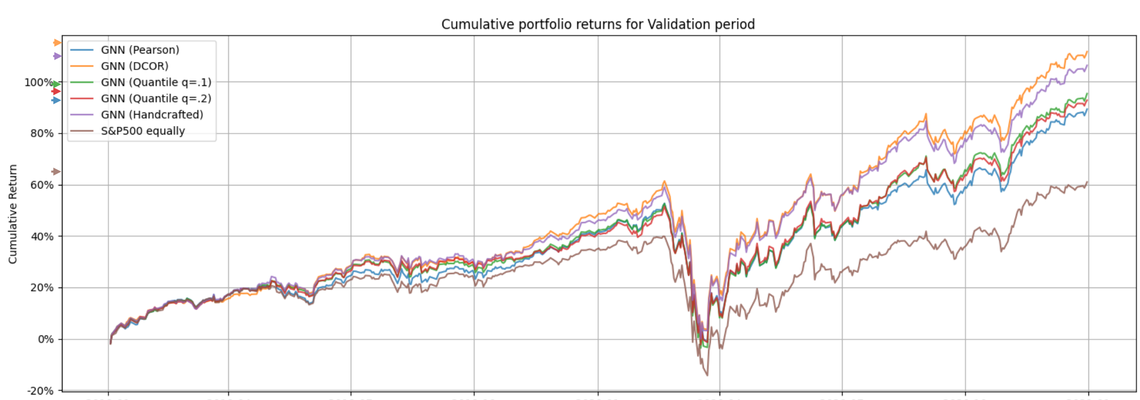


14.07.2023 VP2: Portfolio Optimization with Physics-Inspired Graph Neural Networks 29

29

Backtest optimized portfolio

- **Returns** during **first test** period

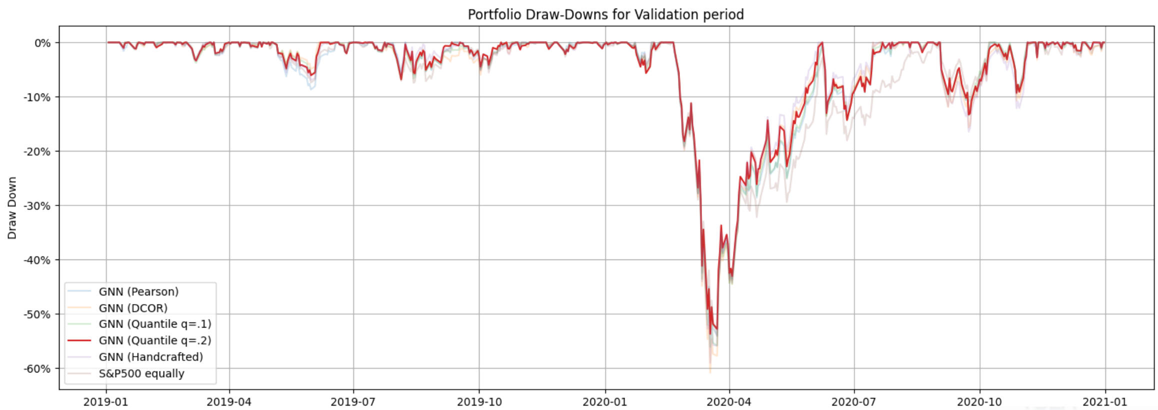


14.07.2023 VP2: Portfolio Optimization with Physics-Inspired Graph Neural Networks 30

30

Backtest optimized portfolio

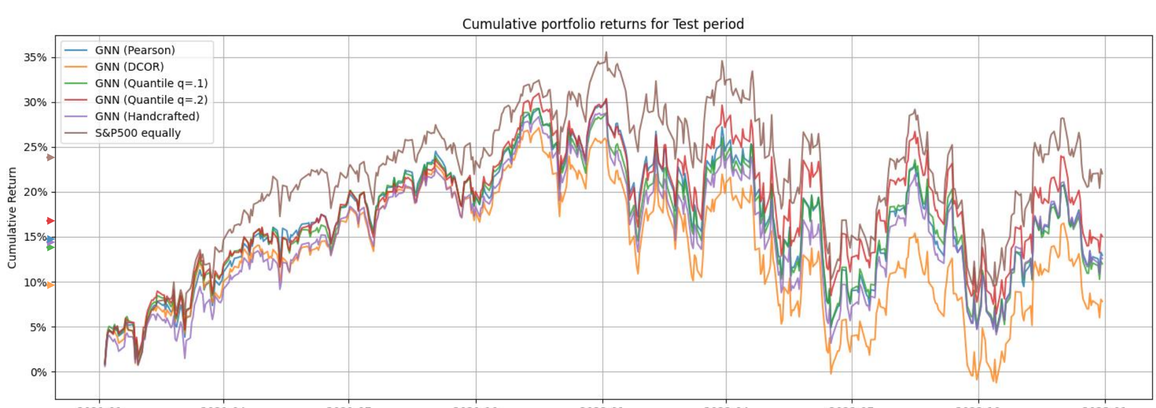
- **Drawdowns** during **first test** period



31

Backtest optimized portfolio

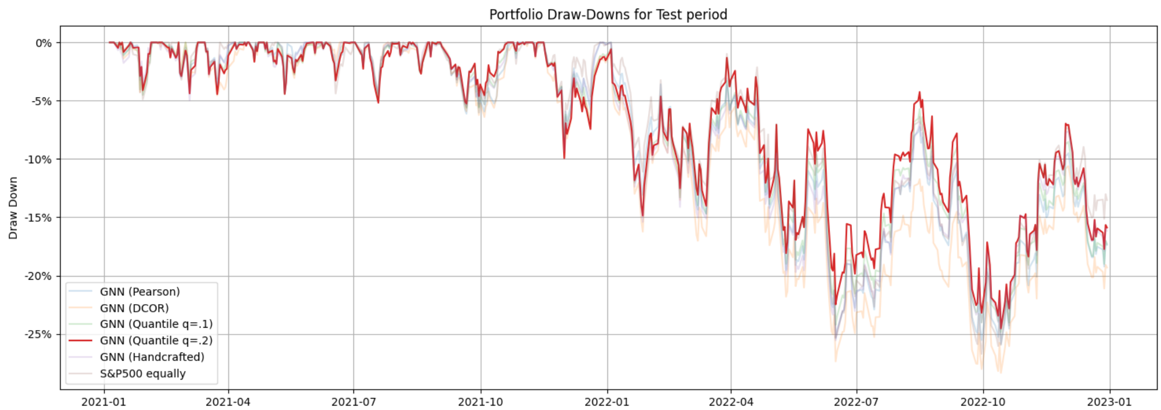
- **Returns** during **second test** period



32

Backtest optimized portfolio

- **Drawdowns** during **second test** period



14.07.2023 VP2: Portfolio Optimization with Physics-Inspired Graph Neural Networks 33

33

Conclusion

14.07.2023 VP2: Portfolio Optimization with Physics-Inspired Graph Neural Networks 35

35

Conclusion

- Overall, good results; successfully applied the approach of Schuetz et al. to a portfolio optimization problem with real data; **biggest achievement: robust GNN mode**l that finds good MIS on different graph structures
- Scalability of the Schuetz et al. approach could not be tested; would be interesting to apply my SAGE model to a larger asset universe.
- Overall **promising results from the backtesting**; but reducing the portfolio's volatility was only partially achieved; further tests with a larger and more diversified asset universe.
- More sophisticated back test and asset allocation method (walk forward backtest)

36

Questions?

37