Asset Pricing via Graph Neural Networks

Yutong Lu, Gesine Reinert, Mihai Cucuringu

Dept. of Statistics, University of Oxford
yutong.lu@stats.ox.ac.uk

UNIVERSITY OF OXFORD

DEPARTMENT OF STATISTICS

Background

'Why do assets have different returns?' is a long-standing question in financial research. Arbitrage Pricing Theory claims that, if there is no arbitrage opportunity in a diversified market, excess returns of assets approximately follow a multi-factor linear model. A recent study [1] illustrates the effectiveness of machine learning models, which can capture nonlinear patterns, in asset pricing. The dynamic linear multi-factor model is defined as

$$R_t = \beta(c_{t-1})\, F_t + \epsilon_t, \tag{1}$$

where N is the number of assets at time point t, R_t is an $N \times 1$ vector of excess returns, and F_t is an $K \times 1$ vector of risk factors. The asset specific characteristics at time $t-1$ are encoded in the matrix c_{t-1} of size $N \times L$. Finally, the $N \times K$ matrix of factor loadings $\beta(\cdot)$ is a function of c_{t-1}, measuring sensitivity to each risk factor, and ϵ_t is an $N \times 1$ vector capturing the idiosyncratic risk of the assets.

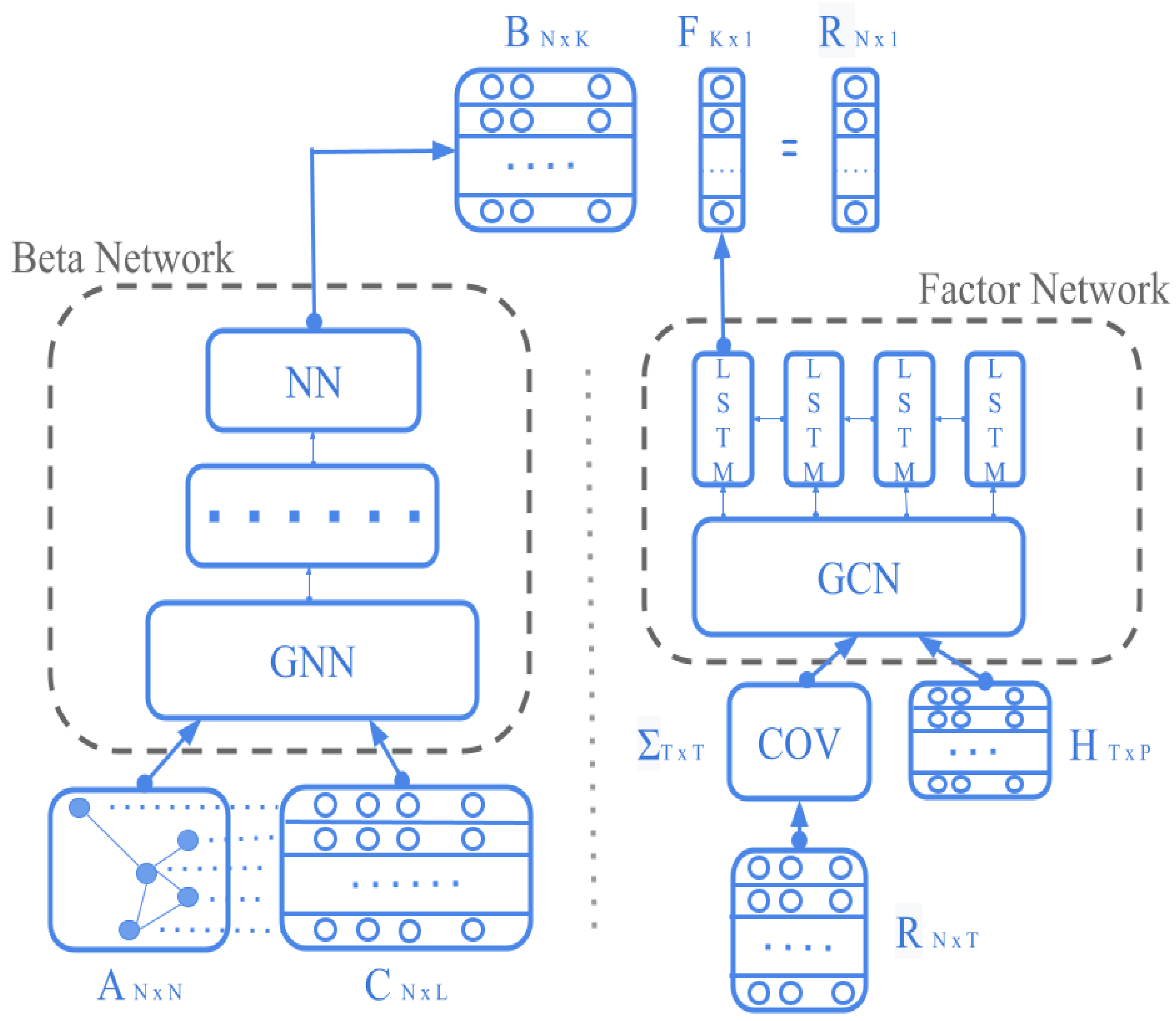
Project Goals

Despite the flexible multi-factor structure, a major challenge is that neither the risk factors nor the risk exposures are known a-priori. Additionally, traditional methods have difficulty in modelling the complex cross-correlation among assets. In this project, we aim to

- Collect asset prices, characteristics and macroeconomic data.
- Construct dynamic networks of assets to model time-varying cross-impact, i.e., employ features of asset i for predicting asset j.
- Develop an asset pricing framework via graph neural network models to incorporate asset specific characteristics, macroeconomic indicators and network effects.

Asset Pricing Framework

- Following the linear factor structure (1), we use graph neural networks to estimate both risk factors and factor loadings, using monthly returns.



- Targeting to minimize both cross-sectional and time series errors, we consider the objective function

$$L_\lambda = \frac{1}{NT}\sum_{t=1}^{T}\sum_{i=1}^{N}(R_{i,t} - \hat{R}_{i,t})^2 + \lambda \cdot \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{T}\sum_{t=1}^{T}(R_{i,t} - \hat{R}_{i,t})\right)^2$$

where λ is a trade-off hyperparameter to balance the weight of the time series errors.

Network of Stocks

We use correlation matrices of stock returns to build dynamic graphs of the market $G_t = (V_t, E_t, A_t, X_t)$, with components given by

- V_t - a collection of vertices, where each vertex is an individual stock
- E_t - a collection of edges, where each edge encodes the connection between two stocks based on the predefined relationship
- A_t - an adjacency matrix representing the graph
- X_t - a matrix of attributes, whose elements encode the features or characteristics of the stocks.

Graph Neural Networks

Given a graph $G = (V, E, A, X)$, graph neural networks (GNN) are a family of neural network (NN) models that take graphs as inputs. The type of GNN mainly used in this project is a Graph convolutional network (GCN) [2]. The propagation rule is given by

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}),$$

where $\tilde{A} = A + I$ and $\tilde{D}$ is the degree matrix of $\tilde{A}$, $W^{(l)}$ is the trainable weight and $\sigma(\cdot)$ is a nonlinear activation function. For the input layer, $l = 0$, $H^{(0)}$ is set to equal the attributes matrix X.

Data

We collect data of US stocks from 1959 to 2016, including

- 94 firm specific characteristics data proposed by [1].
- Daily and monthly prices of all stocks in the characteristics data from CRSP database.
- Monthly macroeconomic indicators from FRED-MD data set [3].

Proof of Concept

Graph convolutional network outperforms conditional autoencoder [1] (benchmark) on both value and equally weighted long-short portfolios constructed based on model forecasts.

- Value weighted portfolios

- Equally weighted portfolios

References

[1] Gu, S., Kelly B., and Xiu, D: Autoencoder asset pricing models, Journal of Econometrics 222.1 (2021): 429-450

[2] Defferrard, M., Bresson, X. and Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering, Advances in Neural Information Processing Systems 29 (2016): 3844-3852

[3] McCracken, M. W., and Ng, S: FRED-MD: A monthly database for macroeconomic research, Journal of Business & Economic Statistics 34.4 (2016): 574-589