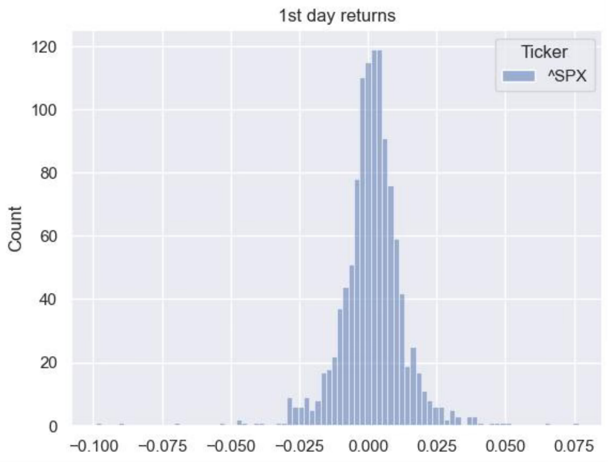
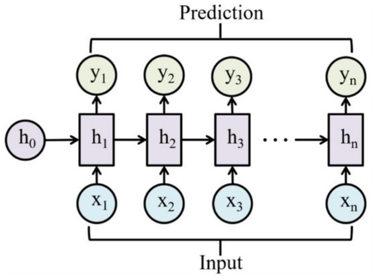
WUTIS



Convolutional Recurrent Neural Networks for Volatility predictions under Payday Anomaly assumption



Semester project proposal

Denys Shkola

Payday anomaly: peak liquidity times

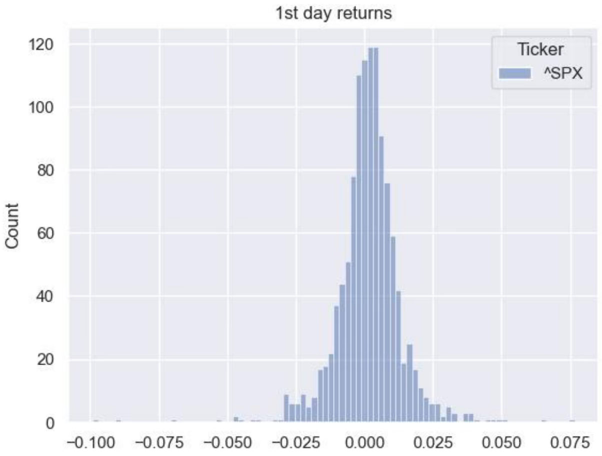
How the payday anomaly could be translated to short and intermediate period of trades

Overview

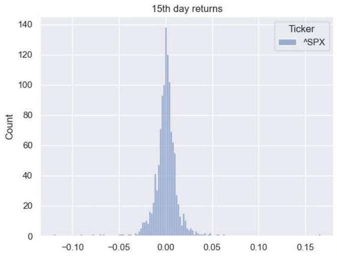
- 1st day of month returns for SPX show higher mean with lower VaR results amid having lower kurtosis
- Trade idea: leveraged positions for 1st month expiry date: CFD, custom futures/ options
- The analogically similar assumption can be used observed and used in shorter trading cycles

Criticism

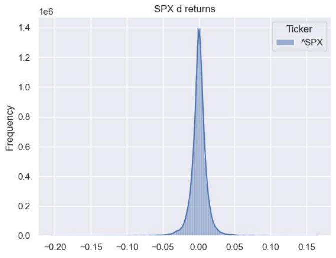
- The relation should be observed for the target market, as first choices are: stock, crypto markets
- Due to small amount of data available for 1st day returns, similar pattern may be analysed for shorter timeframes with statistical and data science/deep approaches
- High-leveraged position might still be too risky to hold, if some insurance is needed - VIX futures suggestion



(a) mean: ~0.14%, kurtosis: ~10



(b) mean: ~0.05%, kurtosis: ~31



(c) mean: ~0.05%, kurtosis: ~31

SPX distributions of daily returns from 1927-12-30 to 2025-03-20: (a) for the 1st day of month, (b) 15th day of month, (c) whole dataset daily returns distribution

Convolutional Recurrent Neural Network for Time-Series data

How CRNN can be used for modelling of future forecasts and utilisation in derivatives trading

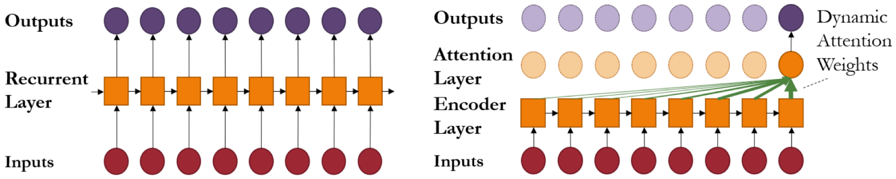
What CRNN?

Convolutional Recurrent Neural Network is a mixture of RNN and Convolutional models which is mainly used for video processing. In the big dataset of different indicators and market data it could observe information from data more deeply and enhance basic RNN usage.

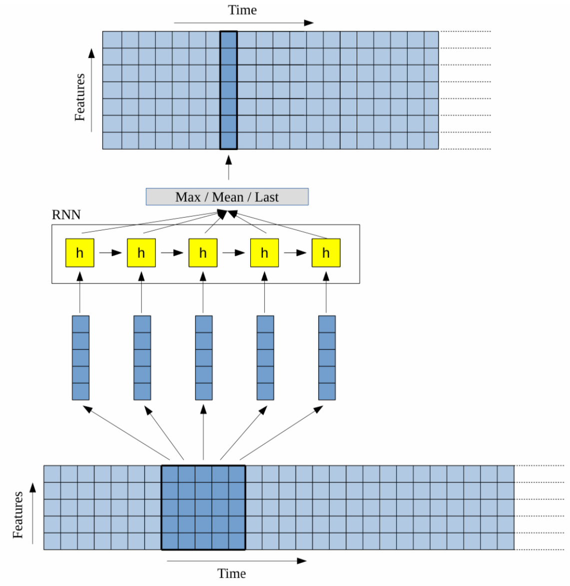
Why CRNN?

Since both RNN and CNN were used for predicting market data, why not use the CRNN which is a mixture of them both?

CRNNs are more lightweight than attention mechanism models, but still have more scalability than regular RNN based layers and can be trained on larger datasets. Training and prediction times are also important and in such case CRNN might be superior to big and bulky transformer models



(b) RNN Model. (c) Attention-based Model.



(b) Extraction of features from a time window using a CRNN layer. First, the time window is fed frame by frame into a recurrent layer. Then, the hidden states of the recurrent layer along the different frames of the window are used to compute the extracted features, by applying a max/mean operator or simply by taking the vector of hidden states of the last time frame in the window. Note that it is possible to extract features from the window using the outputs of the recurrent layer in each time step instead of the hidden states.

Main Model construction and Trading Overview

Construction of the model and technical methodology for creating market trades with it

Scenario

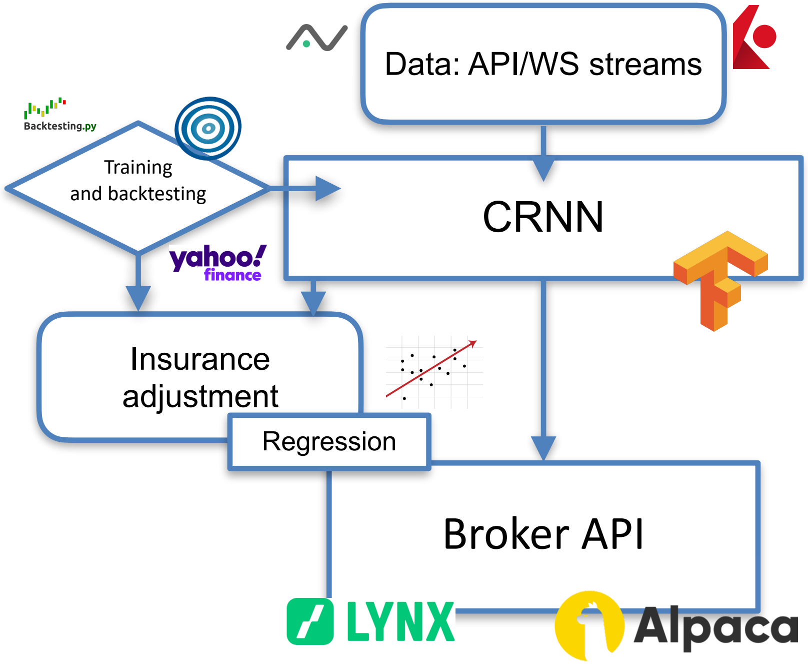
Technical indicators derived from market data, market sentiment data[1] and volatility contracts parameters are fed into the model. After predicting future volatility (and possibly returns) the position of the asset plus risk insurance (amount of which could be optimised with regression techniques) is constructed and the orders are placed via broker API.

Model

Convolutional Recurrent Neural Network is assembled into model with Convolutional + Dense layers to predict future volatility (and returns).

Technical approach and Data

- Backtesting: YFinance data, Backtesting.Py, Alpaca
- Data: AlphaVantage(data steams), broker APIs, IBKR API(specific markets)
- Model implementation: TensorFlow, hyper parameter optimization(optuna)
- Brokers: Alpaca(easily available API), Lynx(broad range of assets)



Source: [1]DeepVol: Volatility Forecasting from High-Frequency Data with Dilated Causal Convolutions, [2]Time Series Forecasting With Deep Learning, Volatility Forecasting for High-Frequency Financial Data Based on Web Search Index and Deep Learning Model