Appendix to the Final Project: Investment Replica

1 Hedge Fund Index Replication

1.1 Why Index Replication?

Hedge Funds are quite inaccessible, both from the point of view of really high entrance fees (generally at least € 100000) and simply because they are sometimes private. Moreover, they have high commission fees (2%) and performance fees (20%, which has to be paid whenever the fund's profits exceed a threshold, typically put around 8%).
These are somewhat justified by the fact that, except for the 2007 financial crisis, hedge funds have consistently been able to generate high returns for their investors.

A first expected reason to invest time in index cloning is therefore the desire to replicate said returns, while being free from the constraints imposed by the hedge funds structure.
Another motivation is Risk Management, either because we have invested in some fund and we want to hedge against possible losses, or simply because we want to understand the risk structure of a certain inaccessible product.
Finally, we could also want to invest in a very large and illiquid index, which would be impossible to replicate physically.

1.2 What is a Future?

Futures are standardized contracts, traded in an exchange trade market. They are a legal agreement between two counterparties, where one of the two agrees to buy (sell) at maturity from (to) the other a certain asset or security at a price which is determined upon signing the contract.
To neglect counterparty's risk, the exchange demands both counterparties to pay a maintenance margin as the (daily) difference in the position's value. As we will see, this factor has to be taken into account when leveraging Futures contracts.

1.3 Why Futures?

There are two main reasons why Futures are used for Investment Replica: their high availability, given they're the most liquid financial derivative in the market, and their leverage property. Indeed, since when we hold a Future contract, we just need to pay the money corresponding to the margins, we can financially expose ourselves more than what our capital would allow us. Other advantages are their tipically lower transaction costs, when compared to the spot market, and their safety as standardized contracts.
However we must careful, this is why we set a maximum leverage threshold of 200%, as required by UCITS/MIFID regulations, that results in a 100% capital exposure on Futures, 20% kept to pay margins and 80% that can be invested in bonds.

2 Data Analysis

Here we offer some more observations and comments on the data analysis performed.

- We removed the LLL1 Future, because its value was constant after a certain date.
- We tried adding three spreads: between 2y and 10y German bonds, between 2y and 10y US bonds, and between the Nasdaq 100 and the S&P 500, so between US Tech and general US Equity.

- Based on the study of the correlation coefficients shown in the slides, we removed all the assets related to government bonds and the Gold future, as they had really low correlation with the Hedge Fund index.
  We can comment that it looks like hedge funds do not significantly invest in AAA government bonds, or in general assets that are safer and less profitable, like gold. Indeed, hedge funds' strategy is to take advantage of bearish and bullish markets, that is why they do not invest in products which are more stable.
- Looking at the assets' values, we decided to normalize the training set to stabilize the variance of the data.
- Since the hedge funds generally lost quite some money in the financial crisis period, we decided to remove these data points from the dataset and start the training procedure from the 1st week of 2010. This decision is justified by the fact that the negative returns of the period from 2007 to 2009 are not really representative of the typical positive returns of hedge funds, and this would cause our model to learn unwanted dynamics in the first sliding windows.

3 Lasso Regression

3.1 How do we choose the lasso function regularization parameters?

The **'lasso'** function on Matlab gives to possibility to feed into it a chosen Lambda vector.
However, after some tries, we decided it was better to just set two inputs of the function: 'NumLambda' and 'LambdaRatio'. We increased 'NumLambda' to 150, from the default 100, to have a finer grid of regularization constants, sacrificing some computational time for a better result. We also incremented 'LambdaRatio' to 1e-3, from the default 1e-4, to include stronger regularization values.

3.2 How do we set the Sliding Window?

In our code we give the possibility of choosing between two different sliding windows, which simply represents the length of the training set for the new week's portfolio weights.

A sliding window of 2 months (= 8 weeks) follows more closely the behaviour of the target index, with a trade-off of having more oscillating weights. This makes sense: every time we shift the window we are changing the 12.5% of the training data, so the slope fitted by the Lasso regression will be quite different from the previous one.
The higher weights' volatility implies higher trading costs. However, this window choice could be preferred by a firm used to performing high frequency trading, or in general a bank that wishes to hedge its position on some hedge funds and is therefore willing to sacrifice some costs for a better risk management; banks are even facilitated in this, as they have lower transaction fees and can perform "in-house" trading.

On the contrary, a sliding window of 1 year is, in our case, less precise in the index tracking, but it reduces re-balancing frequency. This lower weights' volatility is preferable for private clients, that can in this way limit the trading fees spent for the tracking, while still obtaining a quite satisfactory replication result.

We also tried in our project sliding windows of 2 and 3 years, but the results were not successful.

3.3 How do we choose a weekly turnover threshold?

The first idea in order to set a "turnover threshold" was to train the whole model, look at the empirical distribution of weekly turnovers and then choose a certain quantile; however, this approach was discarded as we would have committed data snooping.

We decided to set the weekly threshold by looking at what real world Hedge Funds usually do and, excluding High Frequency Firms, we have discovered that the weekly turnover could vary between 5% and 20% depending on the market conditions and on the strategies adopted. Specifically, we chose to look at the total absolute weekly threshold and to take an educated guess to limit the allowed trading costs to 0.4 bps per week, which represents a weekly maximum of a 10% turnover in our portfolio. Another choice could be to fix a threshold as a percentage of the current portfolio exposure.

Refining the idea above, we can put an additional check: if between today and last week the target index had a consistent price variation, we assume that it is in a high volatility moment and so we can allow the turnover to surpass the threshold, so that the clone can follow the target's behaviour. Instead, if we are in a more stable situation, we try to limit the turnover to have lower trading costs, without sacrificing too much the replica error. In particular, assuming the returns to be normally distributed, we defined the price variation of the index as "high" when it is greater than the 90% quantile (taking into account the last month data).

Another further addition is to limit really small individual trades: if the exposure to a certain asset changes just slightly (again we reasoned in terms of a total weekly turnover of 10%, in this case divided by the number of futures used) from one week to the other, we decide to revert this small change, as it should not significantly impact the model's tracking performance. This could be a sound procedure from an economical point of view, as often, additionally to a variable trading fee, there is a fixed commission to be paid to trade assets, especially for private actors having to deal with brokers.

4 Kalman Filter

A Kalman filter is an algorithm that provides estimates of unknown variables by using a series of measurements observed over time. It can optimally combine prior information with new measurements to produce more accurate and reliable estimates, making it valuable for tracking and predictive applications.
The main reason why we have decided to explore also this path is that the Kalman Filter operates in a recursive manner, so that each weights prediction starts from the previous one, making it highly efficient for real-time processing.
We must note, however, that the Kalman Filter implemented by us does not respect the max 200% leverage constraint.

4.1 Hyper-parameter Tuning

The main task of the Kalman Filter was keeping a good balance between avoiding overfitting and keeping a low Tracking Error Volatility.
We implemented a for loop to find the optimal value of Q, the process noise, minimizing a metric specifically created as the sum of the mean Gross Exposure and the TEV, both scaled using a min-max scaler.
We also tried different values of the variable R, the observation noise, but using a too small value leads to over-fitting and a high Gross Exposure.

5 Bayesian Linear Regression

In the Bayesian viewpoint, we formulate linear regression using probability distributions rather than point estimates. For example the response y is not estimated as a single value, rather it is assumed to be drawn from a probability distribution.

As a first approach we tried with a non hierarchical model, which means that all parameters are at the same level and are estimated independently. In order to obtain the estimates for the coefficients we implemented a Stan model that uses Markov Chain Monte Carlo (MCMC) methods, particularly the No-U-Turn Sampler (NUTS), a variant of the Hamiltonian Monte Carlo (HMC) algorithm, to draw samples from the posterior distribution of the parameters.
We chose some priors such that they are not too informative and did some tuning based on the quality of the results by trying different types of priors. Some evaluations were made also with respect to the choice of the rolling window, that in the end was taken equal to 156 weeks (3 years).

Secondly, we took into consideration a hierarchical model, in which the data model depends on certain parameters. This allows to obtain a more complex model and to have more degrees of freedom for the regression parameters.

In conclusion it can be stated that the advantages of Bayesian inference for assessing model uncertainty include the ability to propagate uncertainties and capture parameter variation across experiments. On the other hand, a disadvantage is the need to make assumptions and approximations when computing the posterior distribution. Another aspect to keep in mind is that the MCMC methods are computationally expensive and obtaining a result can take hours.