**2018 OR & Analytics Student Team Competition**
Final Problem Statement – Revised 11/10/17

# 1) Background

Portfolio optimization refers to the process of finding the optimal proportion of each asset in an investor's portfolio, given a particular investment objective. The two main criteria are to maximize return for a given level of risk or, equivalently, to minimize risk for a given level of expected return. In order to control risk efficiently an investor will try to construct a portfolio that is well diversified, i.e. has as little systematic (non-diversifiable) risk as possible and as many uncorrelated assets as possible to protect the portfolio as a whole against losses in individual assets. A perfectly diversified, long-only (no short selling allowed) portfolio, will only be left with (systematic) market risk that can't be avoided. The traditional Markowitz Mean-Variance Optimization framework offers a way to allocate stocks by considering a trade-off between risk and expected return. Further diversification can be achieved by avoiding concentration in countries, company sizes, sectors, etc.

Institutional investors are generally required to construct portfolios using additional constraints such as, limits on the number of unique assets in the portfolio or how many assets outside a specified benchmark can or must be added. These additional constraints are designed to control the costs that occur when an asset is bought, sold or held while also creating a portfolio that is concentrated enough to outperform the market.

Principal Global Investors has a large base of asset under management, mainly from our pension and insurance business as well from institutional clients. In order to operate profitably these assets get invested in a wide range of asset classes and strategies. One of those asset classes is equities, where portfolio construction is the main point of focus. The following problem is an accurate representation of what challenges an equity portfolio manager faces on a daily basis.

# 2) Problem Definition

Your challenge is to develop an equity portfolio optimization framework that improves risk adjusted excess returns. The optimization framework will minimize the trade-off between risk, in the form of tracking error, and expected excess return through time, subject to several constraints. The portfolio will only consist of long positions, meaning the decision variable (proportion of the portfolio held in each asset) is always positive. The portfolio must also be fully invested at all times. In other words, the proportions of all assets must sum to 100%. In order to avoid concentration in individual assets, we will limit the proportion that can be held in each individual asset. To ensure diversification, there will be constraints on sectors, company sizes (market capitalization), and sensitivity to the equity market (Beta). So far, this represents a basic Markowitz portfolio optimization problem that can be solved using widely open-source or commercial quadratic programming solvers. This Markowitz implementation was also used to create the baseline data that is provided and should serve as a starting point to further add the following extensions to the problem.

As mentioned before, the number of distinct assets in a portfolio is often a requirement in portfolio construction. Therefore, we would like you to add a constraint to limit the target number of distinct assets in the portfolio to a predefined range. We will also introduce an active share constraint which controls how closely we follow a chosen benchmark index in terms of active weights (equation 1, difference between portfolio weight and benchmark weight), as well as a tracking error (standard deviation of active weights) constraint which also controls how closely we follow a benchmark, but in terms of a risk measure rather than return. These final three extensions to the Markowitz problem alter the problem definition considerably. This formulation can no longer be solved with traditional convex optimization techniques. We therefore challenge you to come up with a creative solution for handling these constraints. You must consider the dimensionality of the problem and the associated time it will require to compute a solution. Think about applying heuristics rather than exact implementations as a trade-off for the time it will take to compute a solution. We also want you to think about scalability of the problem, i.e. what would happen if your input data increased dramatically in size, e.g. 10-20x? This would typically happen if we want to run this problem on a global equity universe.

## 3) Define Notations

Input descriptions:

Risk ($\Omega$) : Return covariance matrix for all assets in the universe, which contains the relationships between each pair of historical return series. $\Omega$ is a symmetric matrix where the diagonal elements are the variances of the asset returns, the off-diagonal elements are covariances. As the matrix is symmetric we will only be providing the upper half of the matrix.

Alpha Score ($\alpha$): Expected return score, higher values are expected to have higher future returns.

Date: Date stamp when portfolio optimizations will be performed, also used for corresponding covariance matrices.

Identifier: SEDOL (**S**tock **E**xchange **D**aily **O**fficial **L**ist), unique stock identifier

Sector: Information about the industrial sector each asset belongs to

Beta: Measure of an asset's return sensitivity relative to the equity market return as a whole

Name: Asset's name

Benchmark weights ($w_{bench}$): Proportion of each asset in the Benchmark

MCAPQ: Market Cap Quintile, Market Cap is a measure of the asset's capitalisation size relative to the market. Used to distinguish large companies from small ones. A rank of 1 indicates it is part of the largest 20% of companies (by benchmark weight). A rank of 5 indicates it is part of the smallest 20% of companies.

4 Weekly Returns ($r$): 4 weekly forward returns (the return from the date in question over the subsequent 4 weeks) for each asset, these returns can only be used to calculate the portfolio performance metrics after the optimizations (i.e. it cannot be part of the optimization itself) since this would introduce look-ahead bias. The only proxy for future returns that can be used within the objective function, is the Alpha score.
Terminology:

Long Position: is the buying of a security such as a stock, commodity or currency with the expectation that the asset will rise in value. A long position is the opposite of a short (or short position).

Institutional Investor: is an entity which pools money to purchase securities, real property, and other investment assets or originate loans. Institutional investors include banks, insurance companies, pensions, hedge funds, REITs, investment managers, endowments, and mutual funds.

Variable definitions:

$\boldsymbol{w}$ : total portfolio
$w_i$ : proportion (weight) of overall portfolio held in asset i
$d_i$ : Active weight held in asset i, describes the difference between portfolio weight and benchmark weight for each asset and is a measure for how actively a portfolio manager diverts from a benchmark, see equation 1.

Scaling parameters:
$\lambda$ : tuning parameters to influence how dominant parts of the objective function behave. This is a constant. Typical $\lambda s$ range from 0 (pure minimum variance), to 10 (very aggressively tilted towards targeting expected return). For the provided baseline case, a $\lambda = 1$ was used.

## 4) Mathematical Formulation

We define the problem of portfolio selection as follows:

$$d_i = w_i - w_{bench\,i} \tag{1}$$

Markowitz Formulation

$$\min_{d} \bar{\sigma} = \min_{d}(d^T \Omega\, d - \lambda\, d^T \alpha) \tag{2}$$

$s.t.$

$$w_i \geq 0 \quad \forall\, i \tag{3}$$

$$\sum_i w_i = 1 \tag{4}$$

$$-0.05 \leq d_i \leq 0.05 \quad \forall\, i \tag{5}$$

$$-0.1 \leq \sum_{i\, \in\, sector\ j} d_i \leq 0.1\ \forall\, j \tag{6}$$

$$-0.1 \leq \sum_{i \in MCAPQ\ k} d_i \leq 0.1\ \forall\, k \tag{7}$$

$$-0.1 \leq \sum_i d_i * \beta_i \leq 0.1 \tag{8}$$

Extensions to the traditional problem

$$50 \leq card(w_i \neq 0) \leq 70 \tag{9}$$

$$0.6 \leq 1 - \sum_i min(w_i\,, w_{bench\,i}) \leq 1 \tag{10}$$

$$0.05 \leq \sqrt{d^T\, \Omega\, d} \leq 0.1 \tag{11}$$

Constraints Description:

1. Active weight Definition
2. Markowitz Objective Function balancing risk (tracking error) and excess return
3. Long-only (no negative weights): minimum asset weight = 0%
4. No leverage, i.e., entire portfolio is invested : $\sum weights = 100\%$

5. Min/Max active weights per asset: Benchmark weight +/- 5%
6. Sector active weight: +/- 10%
7. Market Cap Quintile active weight: +/- 10%
8. Beta (β) active weight: -0.1 ≤ β ≤ 0.1
9. Cardinality: Target range on number of stocks (consider $w_i$ as 0 if $w_i \leq 0.001$ ) between 50 to 70
10. $0.6 \leq Active\ share \leq 1$
11. Tracking Error (TE): $5\% \leq TE \leq 10\%$

## 5) Datasets

Data provided is an extract of the constituents of the S&P500 at pre-specified dates. The data extract is taken every 4 weeks over a 10-year horizon, starting 2007-Jan-03. The number of data points per date will vary slightly.

- Time series data includes the following fields:
  - Date
  - Identifier (SEDOL (**S**tock **E**xchange **D**aily **O**fficial **L**ist))
  - Sector
  - Beta
  - Alpha Score
  - Name
  - Benchmark (S&P 500) weight ($w_{bench_i}$)
  - Market cap quintile ($MCAPQ_i$)

- Historical risk model data (upper half of the covariance matrix) used for optimization done on the corresponding date.

- Historical returns for each 4 week period in the 10-year horizon for each stock, in percentage, e.g. 0.05 = 5% (part of the evaluation criteria, see section 7. Included in the results template)

## 6) Simulation Process

A sequence of portfolio rebalances has to be performed on the whole set of data provided. The rebalance dates are pre-specified by the data provided. Every rebalance will result in a new set of weights for each asset.

Institutional investors are required to rebalance their portfolios at regular intervals, or more frequently if market conditions change and therefore portfolio diversification worsens. Typical rebalance intervals are daily, weekly, monthly, quarterly, semi-annually or annually. The more frequently a portfolio gets rebalanced, the closer it reflects the present market conditions and therefore holds less risk and a higher chance to outperform. Unfortunately the cost associated with rebalancing the portfolio cannot be neglected and makes or breaks a portfolio manager's performance. It therefore is necessary to find that sweet-spot between rebalancing as frequently as necessary to capture the potential excess return and minimizing the cost of doing so. For this project we want you to simulate this behaviour of a portfolio manager, by rebalancing your portfolios at pre-defined dates throughout a 10 year history. Every 4 weeks you will be running an optimization based on a new dataset. You will be responsible for managing how many assets in your portfolio change from rebalance to rebalance, i.e. the portfolio turnover. To reflect the costs of trading, the turnover number will be used to reduce your overall returns before you calculate how well your portfolio performed over time. One way of controlling turnover is by introducing further constraints. While rebalancing, you can only use information available to you up until that point in time, i.e. you cannot look into the future. The outcome of your simulation will be a set of assets and the proportion they make up in the portfolio for each rebalance date (and the resulting time series of portfolio returns). This information will be used in the final step to evaluate how well your portfolio performed compared to a benchmark. The first portfolio composition (your starting point) will be provided. Any further rebalance will be performed by you.

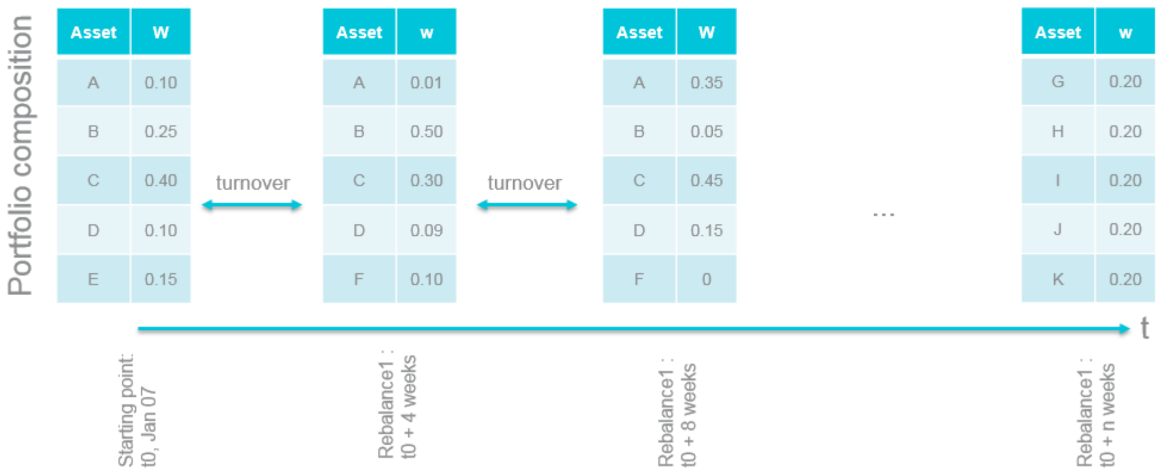


*Figure 1: Portfolio simulation timeline*

## 7) Evaluation Criteria

After all the simulations have been performed the provided 4 weekly return numbers will be used to compute overall portfolio returns. The returns will be aggregated to a total return for the whole portfolio per rebalance date based on each asset's return and its weight in the portfolio on the rebalance date and will be turnover adjusted. In reality, rebalancing is not free. The turnover adjustment represents a proxy for the costs that would occur for trading stocks to rebalance the portfolio. Every 100% in portfolio turnover will reduce the subsequent overall portfolio return by 0.5%.

Start off with calculating the overall turnover numbers (equation 13) for your portfolio for every rebalance date t. Then calculate the overall return for your portfolio (equation 14) for every date t using your calculated portfolio weights and the 1 month returns ($r$) provided to you. In your final step, using equation 15, calculate your turnover adjusted overall portfolio returns for each date t.

Portfolio turnover for period $t$ is defined as:

$$w_{i,t}^{Pre} = \frac{w_{i,t-1} * (1 + r_{i,t-1})}{\sum_i w_{i,t-1} * (1 + r_{i,t-1})} \qquad (12) \qquad turnover_t = \sum_i \left| w_{i,t} - w_{i,t}^{Pre} \right| \qquad (13)$$

Where $w_{i,t-1}$ and $w_{i,t}$ are your calculated, post-rebalance, portfolio weights at two consecutive rebalance dates and $r_{i,t-1}$ are the 4 week forward returns from your previous rebalance date. To incorporate time variance in the weights since the previous rebalance, due to underlying price movements, we have to adjust $w_{i,t-1}$ first by using the returns in equation 12. Thus $w_{i,t}^{Pre}$ represents the weights of the portfolio from time period *t*-1, adjusted for returns in that prior period, prior to the rebalancing that occurs in period *t*.

Overall return for period *t* of your portfolio is defined as:

$$r_{Opt\ t} = \sum_i w_{i,t} * r_{i,t} \qquad (14)$$

Adjusted returns (return adjusted = 4 weekly return – cost penalty * turnover) for period *t*:

$$r_{Txadj\ t} = r_{Opt\ t} - 0.5\% * turnover_t \qquad (15)$$

<u>Metrics</u>:

- Adjusted information ratio

$$IR = \frac{\prod^t (1 + r_{Txadj\ t}) - \prod^t (1 + r_{bench\ t})}{std(r_{Txadj\ t} - r_{bench\ t})} \qquad (16)$$

  Calculate $r_{bench}$ for every date t, using the provided benchmark weights $w_{bench}$ and the 1 month returns $(r)$. *std*() represents the standard deviation of the difference of the adjusted returns and the benchmark returns, where the standard deviation is taken over all the time periods in the simulation.

- Simulation run time (< 3 min per rebalance date, on a standard 8 core machine)
- Turnover
- Elegance and implementability of solution and framework

## 8) Submission

- Performance statistics such as: Cumulative return for portfolio and benchmark (see equation 17), annualized return for portfolio and benchmark (see equation 18), annualized excess return for portfolio vs. benchmark (see equation 19) where n = total number of rebalances and 13 the number of rebalances per year, annualized tracking error (see equation 20), Sharpe ratios for portfolio and benchmark (see equation 21) and information ratio (see equation 16). Performance numbers must be calculated using transaction cost adjusted returns (see equation 15). Feel free to include additional output metrics you consider valuable.

$$R_{cumul} = \prod^{t}(1 + r_t) - 1 \qquad (17) \qquad\qquad R_{anual} = (1 + R_{cumul})^{13/n} - 1 \qquad (18)$$

$$R_{excess,anual} = R_{anual,portfolio} - R_{anual,benchmark} \qquad (19)$$

$$TE = \sqrt{13} * std(r_{Txadj\,t} - r_{bench\,t}) \qquad (20) \qquad\qquad SR = \frac{\prod^{t}(1 + r_t) - 1}{std(r_t)} \qquad (21)$$

- Portfolio rebalance output data:
  - Date of rebalance
  - SEDOLs selected
  - Associated weights for each SEDOL
- Complete codebase to run the simulation
- Documentation, including details on any libraries or commercial packages used
- Specify any additional techniques used, e.g. parallel processing (no distributed computing)
- A template will be provided that will determine the format for submitting your results. Included in the results template is the 'FOUR_WEEKLY_RETURN' data, which can only be used to calculate the performance statistics.

<u>Highly Recommended</u>:
Attention has to be paid to whether all constraints will be implementable in the chosen framework, non-traditional implementations might be necessary, i.e. non-convex. Furthermore, the project will involve a large data set, therefore a programming framework should be chosen that can handle large amounts of data points. It is also highly recommended to try different $\lambda$ values in your objective functions to get a feeling for how much your risk levels scale to achieve higher returns. If you find yourself in a situation where your trading

costs/ turnover gets too high, consider implementing a constraint based on equation 13, to limit your turnover to a certain threshold. In case you encounter any data issues, i.e. missing entries, replace them with dummy entries, in case of the covariance matrix, set 0 for the off-diagonal and 0.5 for the diagonal elements.

Bonus Question:
In the previous discussion, $\alpha$ and $\Omega$ have been assumed to be deterministic. In reality, they are estimates. How would you handle them if they are treated as stochastic variables and reflect the uncertainty in the estimates?