

# Algorithms for optimal combination of strategies

QPlum

December 7, 2014

## Abstract

In this paper we discuss various approaches to combining strategies and testing for optimality given the mandate of a user.

## 1 Nomenclature and problem definition

We are given a matrix of log-returns of  $K$  strategies,  $SR$  of dimensions  $N, K$ . We need to find an algorithm to construct a matrix  $W$  of dimensions  $N, K$  where the interpretation of the row  $W[i, :]$  is the fraction of the portfolio assigned to each of the strategies using all the data available till row  $i$ , which is day( $i$ ) in our context.

If the strategies are based on individual products, which is a subcase and not always applicable, assume we have a matrix of log returns of the product  $RP$  of dimensions  $N, K$  where the interpretation of  $RP[i, j]$  is that it is the log return of the  $j$ th product on the  $i$ th day. We also have a ranking function  $F_{rank}$  that when applied to the returns of two strategies allows us to rank whether strategy A is better than strategy B.  $F_{rank}$  is based on the mandate of the user and performance and risk parameters of the strategy, like annualized returns, annualized standard deviation of returns, max drawdown of returns,  $Var_10$  and turnover.

We also assume a function  $F_{Risk}$  which gives an estimate of the probability of losses of the strategy.

## 2 Approach

$$W[i, j] = \frac{1}{K} \quad (1)$$

$$W[i, j] = \frac{1}{F_{Risk}(SR[(i - t_1) : i, j])} \quad (2)$$

$$W[i, j] = \frac{1}{F_{Risk}(PR[(i - t_1) : i, j])} \quad (3)$$

$$W[i, j] = \frac{\max(0, (SR[(i - t_2) : i, j]))}{F_{Risk}(SR[(i - t_1) : i, j])} \quad (4)$$

$$W[i, j] = e^{(\sum_{l=(i-t_3)}^i SR[l, j])} \quad (5)$$

We have used equality loosely here. In all cases “equals to” means “proportional to”.

1 assigns equal weighting.

2 assigns weights inversely proportional to the risk of the strategy in the past  $t_1$  period.

3 is the traditional strategy independent risk weighting approach, where the allocation is inversely proportional to the variability of the product. Perhaps the Risk function here should be different than the strategy Risk function used, because log-returns of most products tend to be more normally distributed than the returns of a strategy.

4 takes into account the performance of the strategy and not just the risk measure.

5 borrows from best expert learning, or Zinkevich’s convex optimization work. It assigns weights proportional to exponent of the profit. Zinkevich’s algorithm also talks about the constant that is raised to the exponent.

We can try out these methods of combining strategies and see what achieves the highest rank for the given  $F_{rank}$  function.

### 3 Miscellaeous implementation details

We should take into account that we never use future data into allocation at past prices, like the allocation decided with all the data of day(i), will be implemented on day(i+1), and hence we should consider the weights used on day(i+1) as weights computed using all data available till day(i-1).

The Risk (  $X[i-t :i]$  )function has been written in a way that seems to give equal importance to the earliest data point  $X[i-t]$ as the most recent data point  $X[i]$ . In practice, one might find expressions that give exponentially low importance to the past data points as more intuitive.