# Optimization algorithms implementations

### Condition Optimization

#### Random Initialization:

Take a snapshot of the portfolio and sort all fields

For each strategy in the portfolio,

-> for each condition in the strategy

—> for each field type in the condition

—-> if the field type is asset type or time interval, continue

—-> if the condition field type is number field, take the upper and lower bound of it and generate a random number

—> else if the condition field type is select field, generate a random number for one of the options

Notes:

* We can have an initial set of strategies, keep some of them the same, and mutate the rest of them/ randomly initialize

### Strategy optimization

Map each price/volume change for the past n days to a set of strategies that maximize/minimize some value(s)

If the total number of strategy sets is small, we can brute force it. Other we can use a genetic algorithm to search the space

### Genetic algorithm implementation ideas:

* Different types of mutation algorithms
* After crossover, keep the most fit in the population (parents and children)
* Can do multi-objective optimization between sharpe ratio and max drawdown
* Can do single objective optimization on percent gain
* Can approach two ways:
  + Train/validation/test sets
    - More scientifically rigorous
    - Similar to training neural networks
  + Training / test set
    - Easiest to apply to live market data
    - More likely to overfit
* Can train with windows of length ń on the data set of size N where ń is a random number between 0 - N-ń
* Can do variable ń length training
* Can generate two children and when you mutate, use a mutation function on one of the offspring