Bayesian Historical Monte Carlo Simulation

Luca D'Aquanno

Fincite

August 2023

Outline

The Bootstrapping method

- Create future return predictions by random sampling from a time series
- Bootstrapping is a method to obtain a return confidence interval for a certain time horizon

Fincite

The Bootstrapping method: an example

In this framework, users can express their views on the market through a two-step procedure:

- Standardize historical returns
- Incorporate views in the standardized returns:

$$r_{std} = \frac{r_j - \bar{r}}{\hat{\sigma}_r}, \quad r_{new} = (r_{std} \ x \ \sigma_r^{view}) + \mathbb{E}^{view}(r) \qquad (1)$$

- r_{std} are standardized returns
- r_{new} are returns incorporating the user's views on the expected value and the volatility $(\mathbb{E}^{view}(r), \sigma_r^{view})$

Fincite

Drawbacks of the current methodology

Currently, we are assigning equal probability to every joint realization of returns:

- Probabilities associated with the most recent scenarios should obtain more weight
- Probabilities associated with scenarios closer to the user's views should obtain more weight

Fincite

Outline

Fincite

The Entropy pooling method:

Consider a N-dimensional random variable X (stock returns) with T the number of observations:

$$X := \{x_{t,j}\}_{t=1,2,\dots T}^{j=1,2,\dots N}$$

$$X \sim f_X, f_X := \{x_t, p_t\}_{t=1,2,\dots,T}$$

- The distribution of X is defined by a series of joint returns x_t with associated probabilities p_t
- Every x_t (N-length vector for every t) is associated with a scalar p_t (a probability for every joint returns observation)

Fincite

The initial model (prior)

The initial reference model takes into account the time-decay effect by assigning higher weights to the most recent observations:

- $f_X := \{x_t, p_t\}_{t=1,2,\ldots,T}$
- $p = [p_1 \; \ldots \; p_t \; \ldots \; p_T]'$
- $p_{t-1} < p_t$

Fincite

Exponential-decay model for probabilities

The differential equation describing the Exponential-decay[1]:

$$\frac{dp}{dt} = -\lambda p$$

is solved by:

$$p(t) = p_{ew} e^{-\lambda t}$$

- λ is the decay rate
- $\tau_{HL} = \frac{ln(2)}{\lambda}$, $\lambda = \frac{ln(2)}{\tau_{HL}}$

Fincite

[1] Exponential-decay

Time-conditioned probabilities

Each entry p_t of the vector p can be defined as follows:

- $p_t | \tau_{HL} := p_{ew} e^{-\frac{ln(2)}{\tau_{HL}}(|t-T|)}$
- $p_{ew} := 1/\sum_t e^{-\frac{ln(2)}{\tau_{HL}}(|t-T|)}$
- p_{ew} is the equally-weighted probability
- τ_{HL} is approximately the time required for the probability of a scenario to decrease to half of its maximum value in T
- The lower is the half-life parameter τ_{HL}, the higher is the decay rate λ

Fincite

Time-conditioned probabilities: an example

t	$e^{-\lambda\|t-T\|}$	p
1	0,5	0.333
2	1	0.666

- $\tau_{HL} = 1$
- $\lambda = 0,693$
- $T = 2$

It is worth noting:
$p(\tau_{HL}) = \frac{1}{2} p(T)$

Fincite

Prior visualization

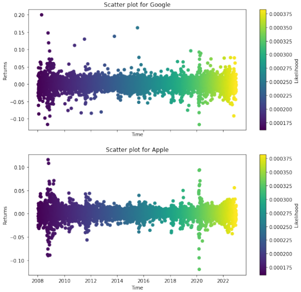

Figure: Google and Iberdrola SA returns scatter plot

From fig.(1) one can notice that most recent returns are getting higher probabilities.

Defining the user's views

To mitigate numerical instability, the time series of each instrument will be normalized with the z-score method:

- $Z_j = \frac{X_j - \bar{X}_j}{\sigma(X_j)}$
- Denoting with X_j the time series of the j-th portfolio's instrument returns
- $Z := \{z_{t,j}\}_{t=1,2,...T}^{j=1,2,...N}$
- $f_Z := \{z_t, p_t\}_{t=1,2,...,T}$

Defining the user's views

- Views (V) are represented as expressions of the expectation of arbitrary functions v(X) of returns [2]

$$V := \left\{ \mathbb{E}_p \left(v(X) \right) \geq v_*^{std} \right\} \tag{2}$$

- Where v_*^{std} is a threshold value that determines the intensity of the view

[2] in [1] several instances of how such functions v(X) can be defined are presented.

Fincite

Defining the user's views: a simple example

The function $v(X)$ maps the risk drivers X in their standardized version Z:

- $v(X) := Z'$
- $\mathbb{E}_p\Big(v(X)\Big) := v(X)p$

Fincite

Defining the user's views: a simple example

Considering also views for volatilities, $\Sigma_k(Z)$ denotes the matrix of volatilities over a rolling window k for instruments' standardized time series:

- $v_1(X) := Z'$ (constraints matrix rows for views on expected returns)
- $v_2(X) := \Sigma_k(Z)$ (constraints matrix rows for views on expected volatilities)
- $\mathbb{E}_p\bigg(v_j(X)\bigg) := v_j(X)p, \; j = 1, 2$

Defining the user's views: a simple example

Assuming two stocks with an expected return of -16% and -20%, and an expected volatility of 26% and 29% respectively. One can express the views as follows:

- $v_* = [-0.16, 0.26, -0.20, 0.29]'$ is the user views' vector
- $v_*(\bar{X}) = [-0.16, -0, 20]$ is the returns' view intensity
- $v_*(\sigma) = [0.26, 0.29]$ is the volatilities' view intensity
- Then is necessary to define: $f(v_*) \rightarrow v_*^{std}$

Fincite

Defining the user's views: a simple example

To define $f(v_*) \rightarrow v_*^{std}$, one can follow the following steps:

1. Fit a probability distribution for expected returns and volatilities
   - A chi-square $\chi_{df,\lambda}$ for expected returns (fitted on X rolling average time series)
   - A log-normal $\mathcal{LN}_{u,\sigma}$ for rolling volatilities (fitted on X rolling volatility time series)
2. Take the value of the cumulative distribution function evaluated at the view intensity to extract the quantile:
   - $q_u, q_\sigma = \chi_{df,\lambda}\left(v_*(\bar{X})\right), \mathcal{LN}_{u,\sigma}\left(v_*(\sigma)\right)$

Defining the user's views: a simple example

Then v_*^{std} can be defined as the value corresponding to quantiles q_u, q_σ in $U_k(Z)$ and $\Sigma_k(Z)$:

$$v_*^{std} = [F_{U_k}^{-1}(q_u), F_{\Sigma_k}^{-1}(q_\sigma)]$$

- F_{U_k} is the empirical cumulative distribution function for Z rolling average (U_k)
- F_{Σ_k} is the empirical cumulative distribution function for Z rolling volatilities

Fincite

Defining the user's views: a simple example

The constraints matrix is:

$$v(X) = \begin{bmatrix} v_1(x_{1,\mathrm{G}}) & \cdots & v_1(x_{t,\mathrm{G}}) & \cdots & v_1(x_{T,\mathrm{G}}) \\ v_2(x_{1,\mathrm{G}}) & \cdots & v_2(x_{t,\mathrm{G}}) & \cdots & v_2(x_{T,\mathrm{G}}) \\ v_1(x_{1,\mathrm{I}}) & \cdots & v_1(x_{t,\mathrm{I}}) & \cdots & v_1(x_{T,\mathrm{I}}) \\ v_2(x_{1,\mathrm{I}}) & \cdots & v_2(x_{t,\mathrm{I}}) & \cdots & v_2(x_{T,\mathrm{I}}) \end{bmatrix}$$

- G, I= Google, Iberdrola SA
- $v_1(x_{t,j}) = z_{t,j}$ (standardized return of j at time t)
- $v_2(x_{t,j}) = \Sigma_k(z_{t,j})$ (rolling volatility of z_j at time t)

Fincite

Defining the user's views: a simple example

The constraint for Google's (bearish) view on expected return is defined in this way:

- $v_1(X_G) = [v_1(x_{1,\mathrm{G}}) \quad \cdots \quad v_1(x_{t,\mathrm{G}}) \quad \cdots \quad v_1(x_{T,\mathrm{G}})]$
- $\mathbb{E}_p\Big(v_1(X_G)\Big) \leq v_*^{std}(\bar{X}_G)$

That can also written as follows:

- $Z_G' p \leq F_{U_{k,G}}^{-1}\Big(\chi_{df_G,\lambda_G}(-0.13)\Big)$
- Similarly we can define all the remaining constraints

Fincite

Posterior Visualization

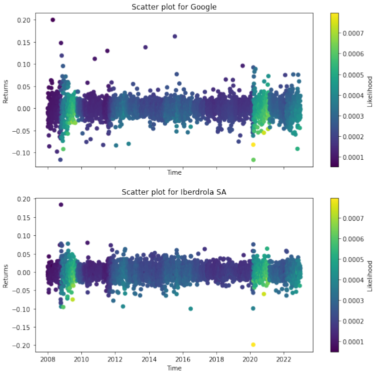

Figure: Google and Iberdrola SA return scatter plot

From fig.(2) one can notice that returns closer to the user views are getting higher probabilities

Fincite

Computing the posterior probability distribution

- Our ultimate goal is to compute a posterior distribution f_X^{post}, that departs from the prior: $f_X := \{x_t, p_t | \tau_{HL}\}_{t=1,2,\ldots,T}$
- To take into account the views, the posterior distribution f_X^{post} is defined by new probabilities p_t^{post} for the same scenario outcomes x_t:

$$f_X^{post} := \{x_t, p_t^{post}\}_{t=1,2,\ldots,T}$$

Fincite

Computing the posterior probability distribution

To compute p^{post}, we must rely on the relative entropy $\xi(p^{post}, p)$ as a measure of the distance between p and p^{post}:

$$\xi(p^{post}, p) := (p^{post})' \left(ln(p^{post}) - ln(p) \right) \qquad (3)$$

Computing the posterior probability distribution

We then define the posterior as the distribution that is closest to the prior, as measured by (3), which satisfies the views (2):

$$\underset{p^{post} \in V}{\text{argmin}_{\text{p}^{\text{post}}}} \; \xi(p^{post}, p), \tag{4}$$

using the Exponential-decay model as the prior for probabilities ($\text{p} = p|\tau_{HL}$).

Fincite

Entropy minimization: defining constraints

To formulate the optimization problem, we must define both inequality and equality constraints:

$$V := \left\{ Fq \geq f, Hq = h \right\},$$

using a vector q as the counterpart of p^{post} posterior probabilities.

Entropy minimization: defining constraints

- $F = v(X)$
- $q = [q_1 \ \ \dots \ \ q_t \ \ \dots \ \ q_T]'$ is a vector collecting the probability for each scenario x_t at time step t
- H is the counterpart of F for equality constraints: $\sum_t q_t = 1$ is specified, using the vector $H = [1 \ \ \dots \ \ 1 \ \ \dots \ \ 1]$ and the scalar $h = 1$
- $f = v_*$ is the vector collecting the value for each view intensity

Fincite

Entropy minimization

The optimization problem is:

$$\underset{Fq \leq f, Hq = h}{\text{argmin}_q} \sum_{t=1}^{T} q_t(ln(q_t) - ln(p_t)) \tag{5}$$

and the Lagrangian function can be expressed in the vectorial notation as:

$$L(q, \lambda_1, \lambda_2) = q'(ln(q) - ln(p)) + \lambda_1'(Fq - f) + \lambda_2'(Hq - h) \tag{6}$$

Fincite

Entropy minimization

The Lagrange multipliers λ_1' and λ_2' are row vectors where the number of rows equals the number of inequality and equality constraints.

Entropy minimization

The first order condition for q reads:

- $\frac{dL}{dq} = ln(q) - ln(p) + 1 + F'\lambda_1 - H'\lambda_2$
- $\frac{dL}{dq} = [0 \quad \dots \quad 0 \quad \dots \quad 0]'$

and solving for q:

$$q(\lambda_1, \lambda_2) = e^{ln(p) - 1 - F'\lambda_1 - H'\lambda_2} \tag{7}$$

Fincite

The Duality principle

Given the convexity of $\xi(q, p)$ in (4) for a posterior q and a fixed prior p, then:

$$\xi(q, p) \leq L(q, \lambda_1, \lambda_2), \forall\ \lambda_1 \geq 0.$$

Taking the minimum of both sides with respect to q, we get

$$\min_q \xi(q, p) \leq \min_q L(q, \lambda_1, \lambda_2).$$

Taking the maximum of both sides with respect to λ, we get:

$$\max_{\lambda_1 \geq 0, \lambda_2} \min_q \xi(q, p) \leq \max_{\lambda_1 \geq 0, \lambda_2} \min_q L(q, \lambda_1, \lambda_2)$$

The Duality principle

It is worth noting that, according to (5):

- If $\lambda_1 < 0$ then the constraint $Fq \leq f$ is violated
- λ_2 does not need to be constrained in order to satisfy $Hq = h$

Fincite

The Duality principle

The dual function is given by definition:

$$G(\lambda_1, \lambda_2) := \min_q L(q, \lambda_1, \lambda_2)$$

therefore, we have:

$$\max_{\lambda_1 \geq 0, \lambda_2} G(\lambda_1, \lambda_2) \geq \min_q \xi(q, p)$$

- The solution of the dual problem is an upper bound for the solution of the primal problem
- If the objective function is strictly convex, then the minimization problem has a unique solution:

$$\max_{\lambda_1 \geq 0, \lambda_2} G(\lambda_1, \lambda_2) = \min_q \xi(q, p)$$

Fincite

Entropy minimization

According to (7) the Lagrange dual function can be expressed as:

$$G(\lambda_1, \lambda_2) := L(q(\lambda_1, \lambda_2), \lambda_1, \lambda_2). \tag{8}$$

The two vectors of Lagrange multipliers (λ_1,λ_2) result from maximizing the Lagrange dual function:

$$(\lambda_1^*, \lambda_2^*) := \underset{\lambda_1 \geq 0,}{\text{argmax}}_{\lambda_1,\lambda_2} \left\{ G(\lambda_1, \lambda_2) \right\}$$

Fincite

KKT conditions

To ensure that λ_1, λ_2 are solving the optimization problem (5), one need to check the KKT conditions:

$$\max_{\lambda_1 \geq 0, \lambda_2} \min_{q} L(q(\lambda_1 \lambda_2), p),$$

such that:

- $\frac{dL(q,p)}{dq} = 0$
- $\lambda_1(Fq - f) = 0$
- $Fq - f \leq 0$
- $Hq - h = 0$

Fincite

KKT conditions

In the minimization case we have:

$$\min_{\lambda_1 \leq 0, \lambda_2} - \min_{q} L(q(\lambda_1 \lambda_2), p)$$

such that:

- $\frac{dL(q,p)}{dq} = 0$
- $\lambda_1 (Fq - f) = 0$
- $Fq - f \geq 0$
- $Hq - h = 0$

Fincite

Entropy minimization

Finally, we can define the set of posterior probabilities as:

$$p^{post} := q(\lambda_1^*, \lambda_2^*) \tag{9}$$

Outline

Fincite

Monte Carlo simulation with historical bootstrapping

Assuming we want to evaluate the expected performance of our portfolio over the next year:

1. Generate $\{x_{n,t}, p_t^{post}\}_{t=1,2,\ldots,252}^{n=1,2,\ldots,N}$ scenarios, for t time steps and n simulations
2. For each time step t, calculate the portfolio return for each simulated scenario n
   1. $r_{n,ptf,t} = x_{n,t} w_t$
   2. with w_t portfolio weights at time step t
3. Compute the cumulative return over the time period, for each simulated scenario:
   1. $\{R_{n,t}\}_{t=1,2\ldots,252}^{n=1,2,\ldots,N} = \sum_{t=1}^{252} \prod_{i=1}^{t} (1 + r_{n,ptf,t}) - 1$

Fincite

Monte Carlo simulation with historical bootstrapping

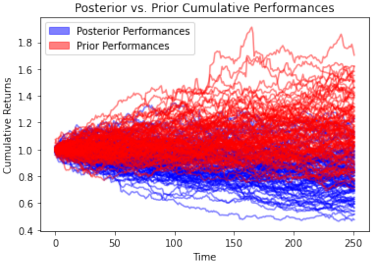

Figure: 100 paths of 1-euro investment in the Google-Iberdrola portfolio over 1 year, views' vector= $v_* = [-0.16, 0.26, -0.20, 0.29]'$

For this simulation, the portfolio weights are constant and equally weighted ($\frac{1}{2}$ Google, $\frac{1}{2}$ Iberdrola SA).

Fincite

Monte Carlo simulation with historical bootstrapping

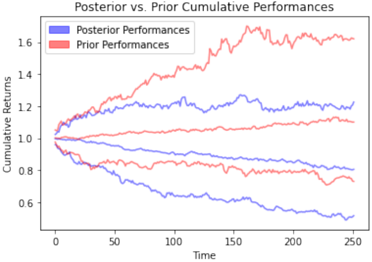

Figure: Quantiles of 1-euro investment in the Google-Iberdrola portfolio over 1 year

The last two figures, outline that negative views have a detrimental influence on portfolio performances

Fincite

Example: Not Coherent Views

There can be an edge case in which there is not an optimal solution for the user views:

- Consider a negative view of Google's expected return (-0.20%) and a positive one for Iberdrola SA ($+0.20\%$)
- The optimization algorithm fails to find an optimal solution that fits both views
- In this case, a probability distribution will be fitted for every instrument's time series of returns

Fincite

Example: Not Coherent Views

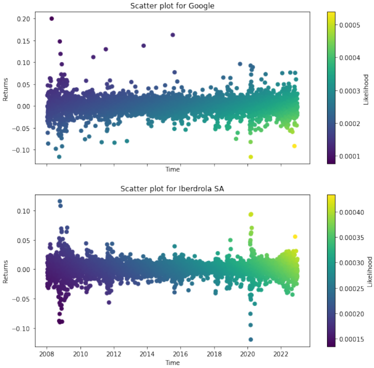

Figure: Google and Iberdrola SA return scatter plot

In Fig.(5), the probability distributions assign different weights to positive and negative returns.

Fincite

Example: Not Coherent Views

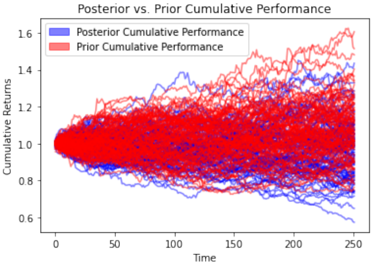

Figure: 100 paths of 1-euro investment in the Google-Iberdrola portfolio over 1 year

From Fig.(6), one can notice that a positive view for Iberdrola SA partially mitigates the negative view for Google.

Example: Not Coherent Views

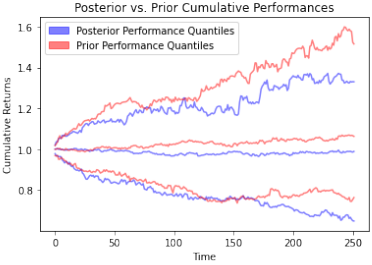

Figure: Quantiles of 1-euro investment in the Google-Iberdrola portfolio over 1 year

Additionally one can also observe the quantile comparison (7) for the new views on the portfolio's instruments.

Advantages of Entropy pooling approach

1. Incorporate views
   - Entropy pooling can incorporate subjective views or beliefs about different risk factors
2. Considers Non-Normal Distributions
   - It does not rely on normality in the risk factors distributions, allowing for non-Gaussian return distribution assumptions
3. Addresses Multidimensionality
   - It accounts for multiple risk factors simultaneously, which can be especially useful in complex portfolios

Fincite

[1] Attilio Meucci. "Mixing probabilities, priors and kernels via entropy pooling". In: *GARP Risk Professional* (2011), pp. 32–36.

Fincite