Finanças quantitativas e Gerenciamento de Risco – 2020/1

Lista de exercícios 2
Prof. Cristiano Arbex Valle
Entrega: Via Moodle

Instruções

(a) Você não precisa necessariamente usar `R`, pode fazer como quiser: `Python`, `MatLab`, etc.

(b) Não é necessário entregar o código, apenas as respostas finais.

Questão 1 Vamos buscar pares cointegrados, candidatos a uma estratégia de *pairs trading* (arbitragem estatística). O teste de *Engle-Granger* é uma forma de encontrar estes pares. Dadas duas séries temporais X e Y, os passos são os seguintes:

1. Verifique que ambas as séries são $I(1)$.

2. Encontre os coeficientes α e β da regressão linear:

$$Y = \alpha + \beta X + \varepsilon$$

3. Calcule a série temporal de resíduos ε e verifique se ela é $I(0)$. Caso sim, as séries são cointegradas.

Utilize as séries de preços diários entre o primeiro dia de 2018 e o primeiro dia de 2020. Encontre 3 pares cointegrados com o ativo `PETR3` e indique a equação de regressão nestes casos. Plote também os gráficos dos resíduos, incluindo uma linha horizontal com a média dos mesmos. Através de inspeção visual dos mesmos, você acha que uma estratégia de pairs trading nestes pares neste período teria trazido lucro?

Questão 2 Nesta questão vamos simular o modelo de Markowitz com rebalanceamento diário e sem recálculo da estratégia. O universo de ativos será composto por todos os ativos disponibilizados (ativos do `iBov` + `BOVA11`). Utilize os dados do primeiro dia de 2019 até o primeiro dia de 2020 como *in-sample* e do primeiro dia de 2020 até o final como *out-of-sample*. Calcule o $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$ através dos dados históricos *in-sample*. Encontre:

(a) O portfolio de variância mínima global $p1$. Calcule μ_{p1} e σ_{p1}, e plote um gráfico de barras com os pesos de todos os ativos i onde $w_i \geq 0.0001$ ou $w_i \leq -0.0001$.

(b) O portfolio de variância mínima global, sem *shorting*, $p2$. Calcule μ_{p2} e σ_{p2}, e plote um gráfico de barras com os pesos de todos os ativos i onde $w_i \geq 0.0001$.

(c) O portfolio eficiente sem *shorting* e com retorno esperado mínimo 0.3%, $p3$. Novamente, calcule μ_{p3} e σ_{p3} e plote um gráfico de barras com os pesos de todos os ativos i onde $w_i \geq 0.0001$.

(d) Calcule as séries out-of-sample dos três portfolios assumindo um investimento inicial de R$1. Plote um gráfico comparativo com a performance dos mesmos no período *out-of-sample*. Inclua também a série do índice `iBov` - normalize o `iBov` para começar de 1.

(e) Calcule o retorno médio e o desvio padrão diários das séries *out-of-sample* para cada portfolio. Estes valores estão condizentes com os valores calculados *in-sample*?

Questão 3 No problema acima, suponha que você quisesse adicionar novas restrições:

(a) Nenhum ativo pode ter peso maior que 10% ou menor que -10%.

(b) Seja $W \subset N$ o conjunto de ativos que fazem parte do setor de teconologia. Você quer que seu portfolio tenha no máximo 30% em ações deste setor.

Escreva as restrições adicionais que você incluiria no seu modelo para garantir estas condições. Não é necessário resolver o modelo acima com estas restrições adicionais.

Conceitos práticos A simulação é essencial para avaliar estratégias de investimento. Geralmente analisamos como a estratégia teria se comportado ao se considerar um certo conjunto de ativos em um determinado período no passado - este processo é conhecido como **backtesting**. Na próxima questão iremos avaliar portfolios escolhidos via Markowitz através deste processo. Para aqueles que não tem experiência com este tipo de análise, apresento dois conceitos:

- **In-sample** e **out-of-sample**: Ao analisar estratégias no passado, geralmente dividimos o período considerado em dois. O primeiro é o período *in-sample*, que será utilizado para escolher a composição da estratégia (análogo aos conjuntos de treinamento e validação em aprendizado de máquina). O segundo é o período *out-of-sample*, que vem sequencialmente após o período *in-sample* e é análogo ao conjunto de teste. Neste período simulamos o "futuro" ao avaliar como a estratégia teria se comportado.

  Por exemplo: temos um universo de N ativos compostos pelos ativos do índice `iBov` e o ativo adicional `BOVA11`. Temos dados diários entre 2016 e 2019. Podemos utilizar os dois primeiros anos como período *in-sample* e os dois últimos como *out-of-sample*. Utilizamos os dados de todos os ativos em 2016 e 2017 para estimar $\boldsymbol{\mu}$ e $\boldsymbol{\Sigma}$ e em seguida encontrar o vetor de pesos $\boldsymbol{w}$ compondo o portfolio. Simulamos então como o portfolio teria se comportado em 2018 e 2019 ao calcular uma série simulada dos valores do portfolio a cada dia durante este período.

- **Rebalanceamento**: Suponha que na simulação você assuma um investimento inicial de $C =$ R\$100000. O preço do primeiro dia *out-of-sample* é P_{i1}. Com o peso w_i em cada ativo, podemos calcular o número de ações x_i a serem compradas:

  $$x_i = \frac{w_i C}{P_{i1}}$$

  O cálculo acima possui duas simplificações: não leva em conta custos de negociação e nem o fato de que x_i pode não ser inteiro (exemplo comprar $x_i = 10.28$ ações?). Mais pra frente veremos como tratar estas situações.

  Ao iniciar um teste out-of-sample, outra decisão a ser tomada é de quanto em quanto tempo atualizaremos o portfolio:

  - *Buy-and-hold*: Você mantém as posições x_i durante todo o período *out-of-sample*, e calcula qual teria sido o valor do portfolio V_t para cada dia out-of-sample $t = 1, \ldots, T$:

    $$V_t = \sum_{i=1}^{N} x_i P_{it} \quad t = 1, \ldots, T$$

  - *Rebalanceamento sem recálculo da estratégia*: Imagine dois ativos `A` e `B` em 3 períodos com preços $P_{At} = (10, 11, 12)$ e $P_{Bt} = (5, 4, 5)$. Considere também $w_A = w_B = 0.5$ e $C = 100$. No dia 1, você calcula que deve comprar $x_A = 5$ e $x_B = 10$ ações.

    Com as alterações de preço no dia 2, o valor investido em `A` e `B` é respectivamente $x_A \cdot P_{A2} = 55$ e $x_B \cdot P_{B2} = 44$. O portfolio vale 99 e os pesos atualizados são $w_A = 55.55\%$ e $w_B = 44.44\%$ - ou seja, o portfolio não possui mais os pesos originais.

    No rebalanceamento, você busca fazer alterações no portfolio para obter novamente os pesos desejados. Sem recálculo da estratégia, manteremos os mesmos pesos decididos anteriormente: $w_A = w_B = 0.5$. Recalculamos $x_A = \frac{0.5 \cdot 99}{11} = 4.5$ e $x_B = \frac{0.5 \cdot 99}{4} = 12.375$. É como se vendêssemos 0.5 ações de `A` e comprássemos 2.375 ações de `B`.

    No terceiro dia, o valor final do portfolio é $4.5 \cdot 12 + 12.375 \cdot 5 = 115.875$. Note que podemos decidir de quanto em quanto tempo o rebalanceamento será feito: todos os dias, a cada 5 dias, etc. Também não precisa ser necessariamente em intervalos regulares.

– *Rebalanceamento com recálculo de estrátegia*: O rebalanceamento acima assumiu que reutilizaríamos os pesos iniciais. Porém, no dia 2 do exemplo acima, temos novas informações que podem ser utilizadas para tomar possivelmente outra decisão. Voltando ao exemplo mais geral acima, suponha que tivéssemos U dias *in-sample*. No dia 2 *out-of-sample*, podemos recalcular a estratégia com Markowitz utilizando os $U + 1$ dias que agora fazem parte do "passado". Podemos também adotar uma janela deslizante, eliminando o primeiro dia *in-sample* para manter o mesmo tamanho U. Se, por exemplo, $U = 500$ e adotamos uma janela deslizante e rebalanceamento diário, o primeiro vetor de pesos é obtido com os dias 1 a 500, o segundo com os dias 2 a 501, e assim vai.

Dicas Se tiver pouca experiência gerando gráficos ou resolvendo problemas de otimização, dê uma olhada no material que disponibilizei em `R` no Moodle. Há um *wrapper* para geração de gráficos e outro para problemas de otimização. Por exemplo, imagine que `weights` seja um vetor em `R` com os pesos diferentes de zero, e que `assets` seja um vetor de strings com os nomes destes ativos. Com o *wrapper* você pode gerar um gráfico de barras com ângulo de 90 graus nos nomes das abscissas através dos comandos:

```
ggOptions                 = getGGOptions();
ggOptions$barChart        = 1;
ggOptions$height          = 6;
ggOptions$width           = 10;
ggOptions$axisTitleSize   = 16;
ggOptions$axisXLabelSize = 8;
ggOptions$axisYLabelSize = 14;
ggOptions$xTitle          = "Ativo";
ggOptions$yTitle          = "Peso";
ggOptions$percentageInYAxis = 1;
ggOptions$axisXLabelAngle = 90;
ggOptions$title           = "Grafico de barras";
ggOptions$saveGraphics    = 1;
ggOptions$imageName       = "teste.png";
plotGraph(weights, xValues = assets, ggOptions = ggOptions);
```

O comando `saveGraphics` indica se o gráfico será salvo no arquivo especificado em `imageName` ou se será exibido diretamente no RStudio.