

# Loren Belei

Mestranda em Ciência Ambiental (PROCAM-USP)

[Loren Belei](#)

[tutorial2\\_exercicios\\_script.txt](#)

[aula\\_4\\_-\\_exercicios.txt](#)

[aula\\_5\\_-\\_exerciciosfinal.txt](#)

[aula\\_7b\\_exercicios.txt](#)

[propostas\\_de\\_funcao.docx](#)

## Proposta 1

### Planejamento da função energia

#### Contextualização

O Princípio da Conservação de Energia estabelece que a energia pode ser transformada ou transferida, mas que nunca será criada ou destruída. A conservação pode ser obtida pela redução de desperdícios e aumento da eficiência elétrica dos aparelhos. Um motor elétrico é um conversor de energia baseado em princípios eletromagnéticos. Possui um Efeito Útil que é diferente da Energia de Entrada. Isso porque, ocorrem perdas no processo de geração, sendo impossível que a conversão seja perfeita. É possível desenvolver equipamentos mais eficientes, que apresentam menor taxa de perda, sendo que, desta forma um motor de alto rendimento será capaz de gerar a mesma quantidade de carga, gastando menos energia que um motor padrão. Funções que calculem e comparem essa diferença de perdas e eficiência entre diversos motores podem ser bastante úteis.

#### Entrada

energia (Data, P, R, M)

- Data = banco de dados
- P = Potencia Nominal
- R = Rendimento Nominal
- M = Tipo de motor

## Verificando os parâmetros

- Data é um objeto classe data.frame.
- P é uma coluna de Data, com valores numéricos  $> 0$ .
- R é uma coluna de Data, com valores numéricos no intervalo  $0 \leq R \leq 1$ .
- M é uma coluna de Data, com valores classe “character” ou “factor”.

## Pseudo-código:

1. Condicional que verifica se há pelo menos um valor diferente do tipo numérico no vetor P.
2. Condicional que verifica se há pelo menos um valor diferente do tipo numérico no vetor R.
3. Condicional que verifica se há pelo menos um valor diferente do tipo character ou factor no vetor M.
4. Cria um novo objeto classe data.frame chamado banco, que contém as colunas M, P e R.
5. Cria um objeto chamado “perda” com um valor numérico resultado de P-R.
6. Cria um objeto chamado “p\_perda” com um valor numérico entre 0 e 1, resultado de perda/P\*100.
7. Adiciona os objetos perda e p\_perda como novas colunas no objeto banco.
8. O objeto banco é organizado em forma descendente utilizando os valores da coluna p\_perda.
9. Cria um gráfico de barras usando as colunas p\_perda (eixo y) e M (eixo x) do objeto banco e armazenado no objeto graph\_perda.
10. Cria o objeto r\_perda, que é uma matriz vazia. Os valores do eixo x (posição i) e y (posição j) correspondem aos valores das categorias da coluna M do objeto banco.
11. Objeto r\_perda é preenchido da seguinte forma:
  - 11.1. Cria-se um loop definido como: para cada i da sequência das categorias da coluna M do objeto banco.
  - 11.2. O objeto banco é filtrado tomando como referência os valores da coluna M que tem valores iguais à variável i. Posteriormente, seleciona-se o valor da coluna perda do banco filtrado e armazena-se esse valor em um novo objeto chamado p1.
  - 11.3. Cria-se um segundo loop, dentro do primeiro definido como: para cada j da sequência das categorias da coluna M do objeto banco.
  - 11.4. O objeto banco é filtrado tomando como referência os valores da coluna M que tem valores iguais à variável j. Posteriormente, seleciona-se o valor da coluna perda do banco filtrado e

armazena-se esse valor em um novo objeto chamado p2.

11.5. Cria-se um objeto P3 que é o resultado de p1- p2.

11.6. As posições i e j da matriz r\_perda são preenchidas tomando o valor de p3.

12. Cria-se um objeto, tipo lista, chamado energia que contem:

12.1. Na posição 1 da lista, o objeto banco.

12.2. Na posição 2 da lista, o objeto graph\_perda

12.3. Na posição 3 da lista, o objeto r\_perda

## Saída

Objeto, tipo lista, que contem:

- Banco de dados com os valores inseridos e novos valores calculados para cada um dos motores.
- Gráfico de barras com o percentual de perda do sistema para cada um dos motores do banco organizado em forma decrescente.
- Matriz com a redução de perda comparando cada um dos motores do banco.

## Referências

RAMOS, Mario Cesar do Espirito Santo. Metodologia para avaliação e otimização de motores elétricos de indução trifásicos visando a conservação de energia em aplicações industriais. 2009. Tese (Doutorado em Energia) - Energia, University of São Paulo, São Paulo, 2009. doi:10.11606/T.86.2009.tde-14062011-141907. Acesso em: 2018-05-04.

Comentários Andre Chalom

Olá, Loren!

Acho que sua proposta tem um bom potencial, mas estou tendo bastante dificuldade em entender qual o objetivo dela. Acho que você poderia explicar melhor o que a sua função vai fazer.

Concordo que “Funções que calculem e comparem essa diferença de perdas e eficiência entre diversos motores podem ser bastante úteis”, mas como a sua função especificamente vai fazer essas comparações? Venda mais sua ideia na descrição 😊

Fora isso, alguns comentários sobre o protótipo e pseudo-código:

1. Não entendi qual o uso do parâmetro “Data”. Os dados que você vai trabalhar não estão nos vetores P, R e M? Então não seria melhor se a função fosse apenas **energia(P, R, M)**?
2. O valor de R é expresso na mesma escala de P, ou é uma porcentagem? Se é uma porcentagem, a perda não deveria ser  $P - (R * P)$ ?
3. O que acontece com a sua função se mais de um valor de M for igual? Ou seja, se existir mais de uma entrada com o mesmo tipo de motor?
4. Veja que gráficos não podem ser retornados de uma função. Você pode criar uma função

que retorna uma lista com x, y e z, e além disso faz um gráfico.

## Proposta 2

# Planejamento da função mtcor

## Contextualização

Entender o comportamento entre duas variáveis é um método comumente utilizado na pesquisa. Entre os métodos mais utilizados encontram-se a correlação de Pearson para dados numéricos e o teste de qui quadrado para dados categóricos.

No entanto, estudos que requerem entender o comportamento entre pares de variáveis podem ter um grau de complexidade muito elevado quando o número de variáveis a comparar é grande. Portanto, é necessário alguma função que otimize a comparação entre pares de variáveis e também que facilite a interpretação destas.

### Entrada

mtcor (Data, Fun)

- Data = banco de dados
- Fun = caráter

## Verificando os parâmetros

- Data recebe um objeto classe data.frame.
- Fun deve ser igual a "cor", mas existe a opção de selecionar "chisq".

## Pseudo-código:

1. Condicional que verifica se há mais de duas colunas no objeto Data (Para continuar precisa ter mais de duas colunas).
2. Condicional que verifica se a informação inserida em Fun é classe character e corresponde a "cor" ou "chisq".

3. Condicional que dependendo do argumento inserido em Fun ("cor" ou "chisq") vai selecionar todas as colunas classe numeric (se seleciona "cor") ou todas as colunas classe character ou factor (se seleciona "chisq") do objeto Data.
4. O objeto Data vai ser substituído com as colunas selecionadas anteriormente.
5. Cria-se uma matriz vazia chamada cor\_mat, onde os eixos "y" e "x" da matriz serão os nomes das colunas do objeto Data.
6. A matriz cor\_mat vai ser preenchida da seguinte forma:
  - 6.1. Cria-se um loop definido como: para cada i da sequência dos nomes das colunas do objeto Data.
  - 6.2. Seleciona-se a coluna do objeto Data com o nome igual ao valor de i. Posteriormente, esta coluna é armazenada num novo objeto chamado c1.
  - 6.3. Cria-se um segundo, dentro do primeiro, definido como: para cada j da sequência dos nomes das colunas do objeto Data.
  - 6.4. Seleciona-se a coluna do objeto Data com o nome igual ao valor de j. Posteriormente, esta coluna é armazenada num novo objeto chamado c2.
  - 6.5. Cria-se uma condicional que verifica:
    - 6.5.1. Se Fun é igual a "cor", vai ser aplicada uma correlação de Pearson entre c1 e c2. Posteriormente o valor da correlação é armazenado em c3.
    - 6.5.2. Se Fun é igual a "chisq" vai ser aplicado um teste de qui quadrado entre c1 e c2. Posteriormente o valor de p resultante é armazenado em c3.
  - 6.6. As posições i e j da matriz cor\_mat serão preenchidas com o valor de c3.
7. Com base na matriz cor\_mat, vai ser gerado um gráfico de matriz de correlações e armazenado no objeto graph\_cor
8. Com base na matriz cor\_mat, vai ser gerado um resumo com o número de variáveis usadas, quantas correlações foram testadas, correlações positivas, negativas ou associações não significativas (para o teste de qui quadrado) e armazenado no objeto summary\_cor.

## Saída:

- Será mostrado um print de summary\_cor.
- Gráfico com a matriz de correlações com as comparações por pares de todas as colunas do banco de dados usado.

Comentários Andre Chalom

Essa função é interessante e tem muito potencial de ser aplicada por usuários de várias áreas! Acho que ela está mais clara do que a primeira proposta. Algumas dicas pra deixar a função mais legal:

1. Números em R podem ser da classe "numeric" ou da classe "integer"! Compare class(c(0.1, 0.2)) com class(1:2)
2. Sua função retorna uma correlação quando "cor" é selecionado, e um valor-p quando

“chisq” é selecionado. Será que isso não vai confundir os usuários? Eu sugeriria adicionar um parâmetro chamado “return.p” que escolhe se o que vai ser retornado (e plotado) vai ser o valor da estatística (correlação ou chi-quadrado) se return.p for FALSE, ou o valor p (de um teste de correlação ou de um teste chi-quadrado) se o return.p for TRUE. O que vc acha?

Boa progRamação!

# Trabalho Final - Função

## Código da Função mtcOR

[trabalho\\_final\\_funcao\\_mtcOR.r.txt](#)

## Help da Função

[trabalho\\_final\\_help.docx](#)

From:

<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2018:alunos:trabalho\\_final:loren.belei:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2018:alunos:trabalho_final:loren.belei:start)

Last update: **2020/08/12 06:04**