

Rafael de Souza Carvalho Melhem



Mestrando do departamento de Ecologia - IBUSP

Meus exercícios

Exercícios

Propostas de Trabalho final

Plano A: Teleportador

Contextualização

Não seria incrível de vez em quando se teletransportar para longe dos problemas? Ou para perto das alegrias?

Pois é, essa audaciosa função do R é motivada por essa ideia de aproximação de realidades.

Matematicamente, a menor distância entre dois pontos é uma reta. Já no mundo real, isso não é tão direto.

O próprio termo “distância” pode ser ampliado, sendo medido não só em metros mas também em segundos, calorias, bytes, reais ou qualquer outro custo.

A internet é consagrada por driblar essas situações, transmitindo informações praticamente em tempo real.

Mas haja abstração para comparar um emoticon com um abraço em uma pessoa querida, ou então uma foto 360º com a visão do alto de uma montanha.

Como, então, vencer essa multifacetada distância?

Não utilizaremos pás, logo o número de metros será o mesmo (senão maior).

Sem chance de sermos codificados em bytes (texto escrito em abril de 2018), então próxima ideia.

Vamos de avião para poupar nosso tempo e energia, que beleza!

Agora só falta reduzir o preço...

É aí que entra nossa função “teleport”, encurtando a distância financeira entre nós e nossos destinos.

Ela irá acessar sites via API, organizar e apresentar os valores.

O foco será em sites que reúnem informações de outros sites, potencializando nossas buscas.

Serão feitas buscas a cada 10 min, enquanto o computador estiver ligado e conectado à internet.

A busca retornará o valor mais baixo disponível no momento e respectivas informações adicionais (trecho, data e site onde é possível fazer a compra).

O valor mais baixo da busca será armazenado e comparado aos anteriores.

Se for o valor mais baixo já encontrado, então ele será apresentado numa janela pop-up.

Juntamente com o valor, serão apresentadas as informações adicionais listadas acima.

A partir daí, cabe à pessoa entrar no site, realizar a compra e se teletransportar.

Outra funcionalidade seria retornar um boxplot do trecho, com os valores em função dos 12 meses do ano.



A ideia é ajudar no planejamento pessoal, estimando épocas com maior probabilidade de preço baixo. Outras abordagens estatísticas podem ser acopladas a essa funcionalidade posteriormente.

Entrada: `teleport(n, classe, origem, destino, dia, mes, conexoes, max)`

`n`: número de pessoas a serem teleportadas (integer: $1 < n < 8$) default=1

`classe`: tipos de cabine (character: "economica", "executiva", "primeira classe")

default="economica"

`origem`: local de partida do voo (character: código aeroportuário padrão IATA) sem default

`destino`: local de chegada do voo (character: código aeroportuário padrão IATA) default=local mais barato (opção de sites)

`dia`: dia de partida do voo (date: dd/mm/aaaa) sem default

`mes`: mês de partida do voo (date: mm/aaaa) default=mês mais barato nos próximos 12 meses (opção de sites)

`conexoes`: numero de máximo de conexões do voo (integer: $0 < conexoes < 3$) default=3

`max`: valor máximo de preço para que esse seja apresentado (numeric com duas casas decimais: $max > 0$) sem default

Verificando os argumentos:

`n`: é um número inteiro entre 1 e 8? Se não, mensagem de aviso: "n informado não reconhecido, calculando para 1 pessoa"

`classe`: é um caractere de um dos 3 tipos de cabine? Se não, mensagem de aviso: "classe informada não reconhecida, calculando para classe econômica"

`origem`: é um caractere contido no código aeroportuário IATA? Se não, mensagem de erro: "origem deve ser um caractere com três letras maiúsculas no padrão IATA".

`destino`: é um caractere contido no código aeroportuário IATA? Se não, mensagem de aviso: "destino informado não reconhecido, calculando para o destino mais barato"

`dia`: é uma data no padrão dd/mm/aaaa ou mes foi preenchido? Se não, mensagem de aviso: "data de ida não informada, calculando para o mês mais barato"

`mes`: é uma data no padrão mm/aaaa? Se não, mensagem de aviso: "mês não informado, calculando para o mês mais barato"

`conexoes`: é um número inteiro entre 0 e 3? Se não, mensagem de aviso: "conexoes informado não reconhecido, calculando para até 3 conexões"

`max`: é um número maior que zero e com duas casas decimais? Se não, mensagem de aviso: "max informado não reconhecido, calculando sem limite máximo de preço"

Saída:

Janela pop-up com o valor de interesse recém encontrado e respectivas informações adicionais (trecho, data e site onde é possível fazer a compra). Boxplot do trecho pesquisado, com o histórico de valores encontrados em função dos 12 meses do ano.

Plano B: Desempilhando palmitos

No primeiro exercício sobre reamostragem e simulação, discutimos se os palmitos estavam

distribuídos aleatoriamente na floresta.

Para isso, calculamos as distâncias médias dos vizinhos mais próximos (mnn) de 1000 amostras. Sendo uma delas a partir de dados coletados (mnn.1) e as outras 999 a partir de dados simulados. A partir daí, observamos que mnn.1 era significativamente diferente da mnn média das simulações e concluímos que a distribuição não era aleatória.

No entanto, foram observados dados de distância iguais ou muito próximos a zero nas simulações, influenciando as mnn, logo, nossa conclusão.

Partindo do princípio que diz: “Dois palmitos não podem ocupar o mesmo local numa mesma amostra”, a ideia dessa função é criar simulações de agrupamento mais fiéis à realidade.

A função poderia se extrapolada para outros grupos de seres vivos, incluindo vegetais, animais sésseis e colônias microbióticas.

Ainda em busca de aumentar a abrangência da função, seria incluída a possibilidade de funções diferentes de `runif()` para o cenário nulo, de acordo com a hipótese a ser testada.

From:

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

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2018:alunos:trabalho_final:rafael.melhem:start&rev=1597223093



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