
Questão 1: Priorização de Iniciativas (Regressão Linear)

Contexto: O departamento de Revenue Management busca identificar quais iniciativas devem ser priorizadas para o próximo ano. Como cientista de dados, sua missão é ranquear as iniciativas de acordo com o impacto na alocação de preço de passagens e serviços auxiliares (ancillaries).

Modelo de Estimativa de Receita:

O modelo segue a estrutura:

$$\text{revenue} = \beta_0 + \beta_1 \cdot \text{ticket_price} + \beta_2 \cdot \text{ancillaries_price}$$

Coeficientes:

- Intercepto (β_0): 4.31
- Ticket Price (β_1): 1.32
- Ancillaries Price (β_2): -0.76

Definição dos Grupos:

- **Group A:** Impacta apenas ancillaries (Ticket: 0, Ancillaries: 1).
- **Group B:** Impacta apenas ticket price (Ticket: 1, Ancillaries: 0).
- **Group C:** Não impacta nenhum (Ticket: 0, Ancillaries: 0).
- **Group D:** Impacta ambos (Ticket: 1, Ancillaries: 1).

Pergunta: Qual alternativa representa a ordem correta de priorização (da maior para a menor receita estimada)?

- () B, C, D, A
- () D, C, B, A
- () D, B, C, A
- () B, D, C, A

Questão 2: Upgrade de Cabine (Regressão Logística)

Contexto: Um modelo logístico tenta prever a probabilidade de um cliente fazer upgrade de cabine em função do seu nível (tier) no programa VOLEMOSPass.

Dados do Modelo:

- **Tiers:** 1 a 5 (onde 5 é o nível mais alto).
- **Intercepto:** -1.40.
- **Coeficiente Tier:** 0.47.
- *Nota: Os coeficientes estão em log-odds.*

Pergunta: Qual é a probabilidade de um cliente com **Tier 5** realizar o upgrade?

- () 0.977
 - () 0.721
 - () 0.279
 - () 0.641
-

Questão 3: Alertas ECAM (Teste de Hipótese)

Contexto: O departamento de Melhoria Contínua coletou dados sobre a manutenção semanal do sistema pneumático de aeronaves A320 e seu efeito nos alertas gerados pelo sistema ECAM.

Dados Coletados (após 1 mês):

- **Sem manutenção:** 405 aeronaves / 351 alertas.
- **Com manutenção:** 339 aeronaves / 192 alertas.

Tarefa: Realize um teste de hipótese para duas proporções (two-tailed) para verificar a significância da diferença.

Pergunta: Quais afirmações estão corretas?

- [] O valor da estatística de teste é 28.22.
 - [] O valor da estatística de teste é 9.17.
 - [] Com 95% de confiança, rejeitamos a hipótese alternativa.
 - [] Com 99% de confiança, rejeitamos a hipótese nula de não-significância.
-

Questão 4: Embarque de Passageiros (Distribuição de Poisson)

Contexto: O número médio de passageiros com necessidades especiais em um embarque é de **1.2 passageiros/voo**. Sabe-se que essa variável possui limite inferior 0 e limite superior potencialmente infinito.

Pergunta: Qual a probabilidade de um voo conter **até 2** passageiros com necessidades especiais?

- () 0.662
 - () 0.216
 - () 0.876
 - () 0.301
-

Questão 5: Maximum trips per customer (SQL)

Texto extraído: When a customer buys a ticket on volemos.com it can be one way (OW) or round trip (RT). A trip is defined by the complete route and one flight for each leg of the trip.

Therefore:

- A round trip purchase from SCL to MAD is one trip with origin SCL and destination MAD and consists of two flights (SCL-MAD and MAD-SCL).
- A one way purchase from SCL to MAD is one trip and one flight.
- A trip can have a maximum of two flights.

We have the following tables in our database:

- **vuelos:** table with the tickets issued by each customer during 2022, where each row is defined by a flight (a one way trip has one row and a round trip has two rows).

Tarefa: Generate the query that provides the total number of trips per customer

- **vuelos:**
 - `id_viaje`: INTEGER. Trip id.
 - `customer_id`: INTEGER. customer id.
 - `rt_flag`: STRING. Trip type (indicates if the trip is round trip or one way: ['RT', 'OW']).
 - `ato_org_trip`: STRING. Origin airport code.
 - `ato_des_trip`: STRING. Destination airport code.
 - `dep_date_month`: INTEGER. Flight month.
 - `dep_date_day`: INTEGER. Flight day.

customer 35826188 one round trip

Questão 6: Recorde de Viagens (SQL)

Texto extraído: When a customer buys a ticket on volemos.com it can be one way (OW) or round trip (RT). A trip is defined by the complete route and one flight for each leg of the trip.

Therefore:

- A round trip purchase from SCL to MAD is one trip with origin SCL and destination MAD and consists of two flights (SCL-MAD and MAD-SCL).
- A one way purchase from SCL to MAD is one trip and one flight.
- A trip can have a maximum of two flights.

We have the following tables in our database:

- **vuelos:** table with the tickets issued by each customer during 2022, where each row is defined by a flight (a one way trip has one row and a round trip has two rows).

(Input tables schema):

- **vuelos:**
 - `id_viaje`: INTEGER. Trip id.
 - `customer_id`: INTEGER. customer id.
 - `rt_flag`: STRING. Trip type (indicates if the trip is round trip or one way: ['RT', 'OW']).
 - `ato_org_trip`: STRING. Origin airport code.
 - `ato_des_trip`: STRING. Destination airport code.
 - `dep_date_month`: INTEGER. Flight month.
 - `dep_date_day`: INTEGER. Flight day.

Exemplo de dados mencionado: "In this sample you can see that customer 35826188 has 2 trips (one one way and one round trip)." (*Isso confirma que o cliente aparece 3 vezes na tabela para totalizar 2 viagens*).

Tarefa: Generate the query that provides the max number of trips and how many people reached the max.

Questão 7: Função latam (Lógica Simples)

Questão 7: Função latam

Enunciado: Escreva uma função que recebe um parâmetro inteiro N e retorna uma lista de comprimento N, contendo em cada posição de 1 a N o seguinte:

- Posição múltipla de 3: 'LAN'
- Posição múltipla de 5: 'TAM'
- Posição múltipla de 3 e 5: 'LATAM'
- Resto das posições: a própria posição

Exemplo:

Entrada: 15

Saída: [1, 2, 'LAN', 4, 'TAM', 'LAN', 7, 8, 'LAN', 'TAM', 11, 'LAN', 13, 14, 'LATAM']

```
def latam(n):
```

Questão 8: Operações Incrementais em Array

Enunciado: A operação é definida com uma lista de três elementos onde cada posição corresponde ao seguinte:

- 0: elemento inicial
- 1: elemento final
- 2: valor a ser somado

A operação é aplicada em um array de zeros de comprimento n, com $n \leq$ último elemento. Escreva a função que, a partir de uma lista de operações, as aplica incrementalmente e retorna o valor máximo da lista final.

Exemplo: E

Entrada: n=10,

operações = [1, 2, 5]

resultado = [5, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0]

```
def arrayManipulation(n, queries):
```

Questão 9: Tornar o Array Positivo (`getMinOperations`)

Enunciado: Dado um array `arr` de n inteiros, modifique o array para torná-lo positivo usando a seguinte operação: selecione um índice i ($0 \leq i < n$) e um valor x ($-10^{18} \leq x \leq 10^{18}$) e altere `arr[i]` para x .

Um array é considerado positivo quando a soma de cada sub-array de comprimento maior que 1 é não-negativa. Formalmente, a condição $\sum_{i=l}^r arr[i] \geq 0$ deve ser mantida para todos os valores de l e r onde $0 \leq l < r < n$.

Determine o número mínimo de operações necessárias para tornar o array positivo.

Exemplo:

`arr = [2, 5, -8, -1, 2]`

Assumindo 0 indexacao, pega $i=2$ e $x=10$. O array modificado é `arr'=[2,5,10,-1,2]`

Agora cada sub array de comp. maior que tem uma soma não negativa. Retorne o numero de operacoes

Saída: 1 (Ao mudar -8 para 10, todas as somas de sub-arrays tornam-se ≥ 0).

Restrições

$1 \leq n \leq 10^5$

$-10^9 \leq arr[i] \leq 10^9$ onde $0 \leq i < n$

```
def getMinOperations(arr):
```

Questão 10: Classificador Random Forest (Scikit-Learn)

Contexto: Você deve treinar um modelo de Random Forest para prever uma variável binária (0/1) com base em um conjunto de dados numéricos e categóricos. O objetivo é maximizar a Acurácia no conjunto de teste.

Esquema de Dados (Schema): A tabela contém as seguintes colunas:

- **feat_0** a **feat_4** (float): Atributos numéricos.
- **label** (int): Alvo da predição (0 ou 1).

Arquivos : train.csv,test.csv,sample_submissions.csv

Entregáveis: 1. Um notebook Jupyter bem comentado e anotado com Markdown. 2.

Arquivo **submissions.csv** com apenas uma coluna: **label**