

# Case Técnico de Data Analysis - iFood

## Visão Geral

Neste case técnico, você deverá desenvolver uma solução baseada em dados para direcionar uma estratégia de cupons como alavanca de crescimento. O desafio permitirá que você demonstre suas habilidades em modelagem, estatística e análise de dados, assim como senso crítico e comunicação de resultados.

Leia o case por completo antes de começar a desenvolver sua solução.

## Contexto

Você foi contratado como Analista de Dados no iFood, uma das maiores empresas de tecnologia da América Latina. Seu primeiro projeto envolve analisar os resultados de um teste A/B realizado recentemente, desenhado para avaliar o impacto de uma estratégia de cupons na retenção de usuários.

Lembre-se de preparar os dados antes da análise utilizando boas práticas do processo de ETL.

## Dados Disponíveis

Você tem a disposição os seguintes datasets:

- Pedidos (`order.json`): `https://data-architect-test-source.s3-sa-east-1.amazonaws.com/order.json.gz`

- Usuários (consumers.csv): <https://data-architect-test-source.s3-sa-east-1.amazonaws.com/consumer.csv.gz>
- Merchants (restaurant.csv): <https://data-architect-test-source.s3-sa-east-1.amazonaws.com/restaurant.csv.gz>
- Marcação de usuários que participaram do teste A/B (ab\_test\_ref.csv): [https://data-architect-test-source.s3-sa-east-1.amazonaws.com/ab\\_test\\_ref.tar.gz](https://data-architect-test-source.s3-sa-east-1.amazonaws.com/ab_test_ref.tar.gz)

As tabelas estão detalhadas no final desse documento.

## O Desafio

1. No iFood, várias áreas utilizam testes A/B para avaliar o impacto de ações em diferentes métricas. Esses testes permitem validar hipóteses de crescimento e a viabilidade de novas funcionalidades em um grupo restrito de usuários. Nos dados fornecidos nesse case você encontrará uma marcação de usuários, separando-os entre grupo teste e controle de uma campanha de cupons, que disponibilizou para os usuários do grupo teste um cupom especial.
  - a) Defina os indicadores relevantes para mensurar o sucesso da campanha e analise se ela teve impacto significativo dentro do período avaliado.
  - b) Faça uma análise de viabilidade financeira dessa iniciativa como alavanca de crescimento, adotando as premissas que julgar necessárias (explícite as premissas adotadas).
  - c) Recomende oportunidades de melhoria nessa ação e desenhe uma nova proposta de teste A/B para validar essas hipóteses.

2. A criação de segmentações permite agrupar usuários de acordo com características e comportamentos similares, possibilitando criar estratégias direcionadas de acordo com o perfil de cada público, facilitando a personalização e incentivando o engajamento, retenção, além de otimização de recursos. Segmentações de usuários são muito utilizadas pelos times de Data, mas a área em que você atua ainda não tem segmentos bem definidos e cada área de Negócio utiliza conceitos diferentes. Por isso, você precisa:
- a) Definir as segmentações que fazem sentido especificamente para o teste A/B que está analisando.
  - b) Estabelecer quais serão os critérios utilizados para cada segmento sugerido no item a). Utilize os critérios/ferramentas que achar necessários, mas lembre-se de explicar o racional utilizado na criação.
  - c) Analisar os resultados do teste A/B com base nos segmentos definidos nos itens a) e b).
3. Com base na análise que realizou nas questões 1 e 2, sugira os próximos passos que o iFood deve tomar. Lembre-se que você precisa defender suas sugestões para as lideranças de Negócio, por isso não esqueça de incluir uma previsão de impacto (financeiro ou não) caso o iFood siga com a sua recomendação. Fique à vontade para sugerir melhorias no processo/teste e para propor diferentes estratégias de acordo com cada segmento de usuário.

## Instruções de Entrega

1. Crie um repositório público ou privado no GitHub.
2. Desenvolva sua solução.
3. Atualize o README com instruções de execução.
4. Envie o link do seu repositório.

Seu repositório deverá conter:

1. **Notebooks de processamento e análise de dados**
  2. **Relatório/ apresentação final**
    - Relatório com suas conclusões e sugestões claramente delineadas
    - Destinado a líderes de negócio tecnicamente leigos
    - Arquivo PDF
- Utilize ferramentas de sua escolha, mas sugerimos fortemente que o processamento e análise sejam realizados em Python/PySpark. A nossa recomendação é utilizar o Databricks Community Edition, Google Colab ou um ambiente local de desenvolvimento. É fundamental que possamos replicar o seu código, portanto, caso opte por desenvolver a solução em um ambiente local, lembre-se de incluir em seu repositório todas as instruções e arquivos necessários para a reprodução.

## **Cr terios de Avalia  o**

- Capacidade de manipula  o e processamento de dados.
- Qualidade e organiza  o do c digo.
- An lise cr tica e solu  o de problemas usando dados.
- Conhecimento de conceitos estat sticos e de experimenta  o.
- Documenta  o clara das premissas e resultados.
- Comunica  o eficaz dos insights obtidos, tanto escrita quanto visual.
- Proatividade em sugerir melhorias e solu  es.

Boa sorte!

## Descrição das tabelas de dados disponíveis

### Pedidos (order.json)

<https://data-architect-test-source.s3-sa-east-1.amazonaws.com/order.json.gz>

Contém dados de cerca de 3.6 milhões de pedidos realizados entre dez/18 e jan/19. Cada pedido possui um `order_id` e os seguintes atributos complementares:

- `cpf (string)`: Cadastro de Pessoa Física do usuário que realizou o pedido
- `customer_id (string)`: Identificador do usuário
- `customer_name (string)`: Primeiro nome do usuário
- `delivery_address_city (string)`: Cidade de entrega do pedido
- `delivery_address_country (string)`: País da entrega
- `delivery_address_district (string)`: Bairro da entrega
- `delivery_address_external_id (string)`: Identificador do endereço de entrega
- `delivery_address_latitude (float)`: Latitude do endereço de entrega
- `delivery_address_longitude (float)`: Longitude do endereço de entrega
- `delivery_address_state (string)`: Estado da entrega
- `delivery_address_zip_code (string)`: CEP da entrega
- `items (array[json])`: Itens que compõem o pedido, bem como informações complementares como preço unitário, quantidade, etc.

- `merchant_id` (string): Identificador do restaurante
- `merchant_latitude` (float): Latitude do restaurante
- `merchant_longitude` (float): Longitude do restaurante
- `merchant_timezone` (string): Fuso horário em que o restaurante está localizado
- `order_created_at` (timestamp): Data e hora em que o pedido foi criado
- `order_id` (string): Identificador do pedido
- `order_scheduled` (bool): Flag indicando se o pedido foi agendado ou não (pedidos agendados são aqueles que o usuário escolheu uma data e hora para a entrega)
- `order_total_amount` (float): Valor total do pedido em Reais
- `origin_platform` (string): Sistema operacional do dispositivo do usuário
- `order_scheduled_date` (timestamp): Data e horário para entrega do pedido agendado

## **Usuários** (`consumers.csv`)

<https://data-architect-test-source.s3-sa-east-1.amazonaws.com/consumer.csv.gz>

Contém dados de cerca de 806k usuários do iFood. Cada usuário possui um `customer_id` e os seguintes atributos complementares:

- `customer_id` (string): Identificador do usuário
- `language` (string): Idioma do usuário
- `created_at` (timestamp): Data e hora em que o usuário foi criado
- `active` (bool): Flag indicando se o usuário está ativo ou não

- `customer_name` (string): Primeiro nome do usuário
- `customer_phone_area` (string): Código de área do telefone do usuário
- `customer_phone_number` (string): Número do telefone do usuário

## **Merchants** (`restaurant.csv`)

<https://data-architect-test-source.s3-sa-east-1.amazonaws.com/restaurant.csv.gz>

Contém dados de cerca de 7k restaurantes do iFood. Cada restaurante possui um `id` e os seguintes atributos complementares:

- `id` (string): Identificador do restaurante
- `created_at` (timestamp): Data e hora em que o restaurante foi criado
- `enabled` (bool): Flag indicando se o restaurante está ativo no iFood ou não
- `price_range` (int): Classificação de preço do restaurante
- `average_ticket` (float): Ticket médio dos pedidos no restaurante
- `delivery_time` (float): Tempo padrão de entrega para pedidos no restaurante
- `minimum_order_value` (float): Valor mínimo para pedidos no restaurante
- `merchant_zip_code` (string): CEP do restaurante
- `merchant_city` (string): Cidade do restaurante
- `merchant_state` (string): Estado do restaurante
- `merchant_country` (string): País do restaurante



## **Marcação de usuários que participaram do teste A/B (ab\_test\_ref.csv)**

`https://data-architect-test-source.s3-sa-east-1.amazonaws.com/ab_test_ref.tar.gz`

Contém uma marcação indicando se um usuário participou do teste A/B em questão. Assim como a base de usuários, cada usuário possui um `customer_id`. Os campos são:

- `customer_id` (string): Identificador do usuário
- `is_target` (string): Grupo ao qual o usuário pertence ('target' ou 'control').