

# **MVP Pipeline de Dados**

Pesquisa sobre aparelhos celulares

Edmilson Prata da Silva

PUC-RJ

MBA em Ciência de Dados e Analytics

Disciplina de Engenharia de Dados (40530010057\_20250\_01)

Santa Cruz de La Sierra, Bolívia

29 de Março de 2025

## Sumário

1. Escopo do Trabalho.....	3
1.1. Definição do Problema.....	3
1.2. Criação de uma Pipeline de Dados.....	3
2. Fonte de Dados.....	4
2.1. Descrição da fonte de dados.....	4
2.2. Features da fonte de dados.....	4
3. Modelo e Catálogo de Dados.....	5
3.1. Modelagem de Dados.....	5
3.2. Catálogo de Dados.....	6
3.3. Métricas.....	7
3.3.1. Métricas de Preço por Empresa.....	7
3.3.2. Métricas Técnicas por Empresa.....	7
3.3.3. Métricas de Desempenho – TOP TEN.....	7
3.3.4. Métricas de Segmentação.....	7
4. A Pipeline de Dados.....	8
4.1 Criação da base de dados.....	8
4.1.1. Camada Bronze.....	8
4.1.1. Camada Silver.....	8
4.1.1. Camada Gold.....	8
4.2. ETLs.....	8
4.2.1. ETL de coleta de dados (camada bronze).....	8
4.2.2. ETL de análise e tratamento de dados (camada silver).....	9
4.2.3. ETL de otimização de dados (métricas – camada gold).....	10
5. Autoavaliação:.....	10
5.1. Métricas de Preço por Empresa.....	10
5.2. Métricas Técnicas por Empresa.....	11
5.3. Métricas de Desempenho – TOP TEN.....	13
5.4. Métricas de Segmentação.....	14
5.5. Conclusões.....	16
6. Evidências do Trabalho.....	17

# 1. Escopo do Trabalho

## 1.1. Definição do Problema

O objetivo deste trabalho é responder a questionamentos de negócio para uma empresa que deseja iniciar no ramo de fabricação de aparelhos celulares. Para esta finalidade, deseja-se obter insights sobre este ramo de atividade e, desta forma, definir as características para o lançamento de produtos com maior potencial de venda e lucratividade, acompanhando as tendências de mercado. Para isto, uma base de dados pública, a ser detalhada mais à frente, será utilizada para responder às seguintes questões de negócio:

### a) Quais características técnicas dos aparelhos mais influenciam na definição do preço?

É preciso descobrir quais são as configurações mais relevantes do ponto de vista da precificação. Com isso, pode-se levantar custos dos investimentos e esses dados apoiarão na definição de uma política de preços dos produtos a serem projetados e fabricados. Determinadas características podem ter maior influência no preço com um custo menor que outros. Isso pode gerar uma vantagem, potencializando estas características para atrair os clientes e tornar o produto mais competitivo.

### b) Qual é a variação existente nas características técnicas dos aparelhos?

Saber qual a faixa de valores de cada configuração, ou seja, o quanto varia cada característica técnica, tais como memória e CPU, permite encontrar o range adequado para definição de cada uma das características dos modelos a serem fabricados. Será possível conhecer o mínimo e o máximo de cada característica e também como se distribuem os modelos concorrentes nestas faixas de valores. Estes dados apoiaram na definição da configuração ideal para os modelos de entrada e os modelos mais sofisticados, no mesmo patamar ou superando as configurações dos concorrentes.

## 1.2. Criação de uma Pipeline de Dados

Para dar suporte ao trabalho de análise de dados, necessário ao atendimento das necessidades descritas, será construída uma Pipeline de Dados utilizando o ambiente de Cloud chamado Databricks, em sua versão Community, que é gratuita. Esta versão, por ser *free*, possui restrições de recursos de hardware, mas é adequada e suficiente para o trabalho que será realizado.

O Databricks pode ser acessado na internet<sup>1</sup>. Esta é uma plataforma de nuvem completa que oferece suporte à banco de dados e permite a criação de componentes de ETL (Extração, Transformação e Carga) com praticidade e agilidade.

## 2. Fonte de Dados

### 2.1. Descrição da fonte de dados

A fonte de dados utilizada foi obtida no site Kaggle<sup>2</sup>, que é bastante conhecido por profissionais das áreas de *Data Science* e *Analytics*. Este site oferece bases de dados de todos os tipos de temas, de dados científicos a dados para negócios. É uma base comunitária bastante respeitada e que oferece dados de qualidade de todo o mundo.

O conjunto de dados utilizado para este trabalho, segundo a descrição e detalhamento contido no próprio site<sup>3</sup>, segue conforme citação a seguir, em versão traduzida do original, em inglês:

*“Este conjunto de dados contém especificações detalhadas e preços oficiais de lançamento de vários modelos de celulares de diferentes empresas. Ele fornece insights sobre hardware de smartphones, tendências de preços e competitividade de marcas em vários países. O conjunto de dados inclui recursos importantes como RAM, especificações da câmera, capacidade da bateria, detalhes do processador e tamanho da tela.”*

A documentação ressalta também que um aspecto importante do conjunto de dados são as informações de preços. Estes preços, na verdade, são os preços oficiais de lançamento dos aparelhos celulares. Para esta pesquisa, utilizaremos o valor em dólar, apesar do conjunto de dados conter preços de alguns outros países. Estes preços podem oferecer insights valiosos sobre tendências de mercado e, devido a esta característica, a base foi escolhida para apoiar este trabalho de pesquisa.

### 2.2. Features da fonte de dados

Nesta seção serão detalhados os campos, ou *features*, do conjunto de dados obtido no site do Kaggle para a realização do trabalho desta pesquisa, conforme segue:

1. **Company Name:** A marca ou fabricante do telefone celular.

---

1 Endereço do Databricks na internet: <https://community.cloud.databricks.com>

2 Endereço do Kaggle na internet: <https://www.kaggle.com/>

3 Endereço do conjunto de dados na internet: <https://www.kaggle.com/datasets/abdulmalik1518/mobiles-dataset-2025/data>

2. **Model Name:** O modelo específico do smartphone.
3. **Mobile Weight:** O peso do celular (em gramas).
4. **RAM:** A quantidade de memória de acesso aleatório (RAM) no dispositivo (em GB).
5. **Front Camera:** A resolução da câmera frontal (selfie) (em MP).
6. **Back Camera:** A resolução da câmera traseira principal (em MP).
7. **Processor:** O chipset ou processador usado no dispositivo.
8. **Battery Capacity:** O tamanho da bateria do smartphone (em mAh).
9. **Screen Size:** O tamanho da tela do smartphone (em polegadas).
10. **Launch Price** (Paquistão, Índia, China, EUA, Dubai): O preço oficial de lançamento do celular no respectivo país no momento de seu lançamento. Os preços variam de acordo com o ano em que o celular foi lançado.
11. **Launched Year:** O ano em que o celular foi lançado oficialmente.

### 3. Modelo e Catálogo de Dados

#### 3.1. Modelagem de Dados

A partir da fonte de dados já apresentada, constrói-se o modelo de dados em Esquema Estrela. O Esquema Estrela consiste em uma tabela de fatos central e várias tabelas de dimensões relacionadas. Neste caso, a tabela de fatos será “smartphones”, e as tabelas de dimensões serão “company”, “model” e “price”, conforme abaixo:

- Tabela de Fatos “smartphones”: Contém métricas e chaves estrangeiras para as tabelas de dimensões.
- Tabelas de Dimensão “company”: Armazena informações sobre as empresas fabricantes.
- Tabelas de Dimensão “model”: Armazena informações sobre os modelos dos smartphones.
- Tabelas de Dimensão “price”: Armazena informações sobre os preços de lançamento em diferentes países.

## 3.2. Catálogo de Dados

A seguir, será apresentada uma descrição detalhada dos dados e seus domínios, contendo valores mínimos e máximos esperados para dados numéricos e possíveis categorias para dados categóricos, entre outras informações relevantes sobre os dados.

### A. Tabela de Fatos “silver.smartphones”:

1. smartphone\_id (PK): Chave primária.
2. company\_id (FK): Chave estrangeira referenciando a tabela company.
3. model\_id (FK): Chave estrangeira referenciando a tabela model.
4. price\_id (FK): Chave estrangeira referenciando a tabela price.

### B. Tabelas de Dimensão “silver.company”:

1. company\_id (PK): Chave primária.
2. company\_name: Nome da empresa fabricante.

### C. Tabelas de Dimensão “silver.model”:

1. model\_id: Chave primária.
2. model\_name: Nome do modelo.
3. mobile\_weight: Peso em gramas.
4. Ram: Quantidade de memória RAM.
5. front\_camera: Resolução da câmera frontal.
6. back\_camera: Resolução da câmera traseira.
7. Processor: Processador do aparelho.
8. battery\_capacity: Capacidade da bateria.
9. screen\_size: Tamanho da tela.
10. launched\_year: Ano de lançamento.

### D. Tabelas de Dimensão “silver.price”:

1. price\_id (PK): Chave primária.
2. model\_id (FK): Chave estrangeira referenciando a tabela model.

3. country: País onde o preço foi registrado.
4. launched\_price: Preço de lançamento no país.

### **3.3. Métricas**

Com o modelo apresentado, é possível colher métricas detalhadas com potencial relevância para o negócio, a fim de proporcionar uma análise rica dos dados e produzir insights para as tomadas de decisões. Com base nessas métricas, é possível identificar oportunidades de negócio, entender o comportamento do consumidor, dos concorrentes e tomar decisões estratégicas. A seguir, serão apresentadas as métricas cobrindo aspectos como preço, desempenho, características técnicas, tendências e comparações entre empresas.

#### **3.3.1. Métricas de Preço por Empresa**

- a) Preço médio, mais alto e mais baixo por empresa: Qual é o smartphone mais caro e mais barato em cada país?

#### **3.3.2. Métricas Técnicas por Empresa**

- a) Média, mínimo e máximo da capacidade da bateria por empresa;
- b) Média, mínimo e máximo do tamanho de tela dos smartphones por empresa;
- c) Média, mínimo e máximo da memória RAM dos smartphones por empresa;

#### **3.3.3. Métricas de Desempenho – TOP TEN**

- a) Smartphones com maior capacidade de bateria;
- b) Smartphones com maior memória RAM;
- c) Smartphones com melhor câmera traseira.

#### **3.3.4. Métricas de Segmentação**

- a) Smartphones por Faixa de Preço, sendo: abaixo de U\$ 200, entre U\$ 200 e U\$ 500 ou acima de U\$ 500.
- b) Smartphones por Tamanho de Tela, sendo: pequena, sendo < 6"; médias, entre 6" e 6.5"; ou grandes, > 6.5"?
- d) Distribuição dos smartphones por quantidade de memória RAM.

## 4. A Pipeline de Dados

A pipeline de dados foi construída e executada a partir do ambiente de cloud Databricks Community Edition. O funcionamento da pipeline segue especificado adiante.

### 4.1 Criação da base de dados

A base de dados da pipeline é dividida em três camadas, sendo elas as camadas bronze, silver e gold, conforme abaixo:

#### 4.1.1. Camada Bronze

Camada de dados brutos (RAW), onde são persistidos os dados da forma como foram coletados na origem. Nessa camada não é aplicada nenhum método de tratamento, transformação, validação ou enriquecimento de dados. Seu objetivo é atender necessidade como reproprocessamento, auditoria e análise histórica. A camada bronze é criada por [este notebook](#)<sup>4</sup> Python no ambiente Databricks.

#### 4.1.1. Camada Silver

Camada de dados validados, tratados e transformados a partir da camada bronze. Seu objetivo é atender trabalhos de analytics, gerando relatórios e painéis executivos. A camada silver é criada por [este notebook](#)<sup>5</sup> Python no ambiente Databricks.

#### 4.1.1. Camada Gold

Camada de dados enriquecidos e otimizados a partir da camada bronze. Seu objetivo é oferecer maior performance e privilegiar o uso em trabalhos de ML. A camada gold é criada por [este notebook](#)<sup>6</sup> Python no ambiente Databricks.

## 4.2. ETLs

A pipeline de dados conta com três ETLs feitos em notebook Python, sendo um para cada camada de dados, conforme detalhamento adiante.

### 4.2.1. ETL de coleta de dados (camada bronze)

O ETL responsável pela coleta de dados está disponível [neste notebook Python](#)<sup>7</sup>. O notebook coleta os dados a partir da base "Mobiles Dataset (2025)", já detalhada neste

---

4 Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/2\\_create\\_db\\_bronze.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/2_create_db_bronze.ipynb)

5 Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/3\\_create\\_db\\_silver.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/3_create_db_silver.ipynb)

6 Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/4\\_create\\_db\\_gold.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/4_create_db_gold.ipynb)

7 Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/5\\_etl\\_db\\_bronze.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/5_etl_db_bronze.ipynb)



documento. Ela oferece informações técnicas sobre aparelhos tipo smartphone, com lançamento realizado no ano de 2025.

Os dados são armazenados na camada bronze em seu estado bruto, preservando características da fonte dos dados original que, neste caso, é a base do Kaagle, site votlado para trabalhos de analytics e data science, bastante conhecido pela comunidade.

#### **4.2.2. ETL de análise e tratamento de dados (camada silver)**

O ETL responsável pelo tratamento e transformação dos dados está disponível [neste notebook Python](#)<sup>8</sup>. O notebook carrega os dados a partir da camada bronze, realiza os tratamentos necessários e persiste os dados transformados na camada silver.

O conjunto de dados obtido na base do Kaggle não apresenta dados faltantes e nulos. Contudo, o conjunto de dados trás, em todas as colunas, a unidade de medida e os símbolos de moeda junto com o valor numérico, que é o dado de interesse. Logo, o primeiro tratamento realizado foi a separação do dado numérico destes símbolos de moeda e unidades de medida indesejáveis.

Em seguida, foi necessário tratamento de dados duplicados na base. Estes dados foram confirmados como linhas em duplicidade e as cópias foram eliminadas. Foram detectados também alguns falsos positivos quanto à duplicidade. Na realidade, tratavam-se de modelos de marcas diferentes, cuja nomenclatura e características eram bastante parecidas. Neste caso, um identificador único foi atribuído a todos os modelos de smartphones, o qual serviu também como chave primária para a tabela de modelos de smartphones.

Por último, foi necessário tratar também os dados de preços, em 5 países diferentes, pois o valor estava fora de padrão. Alguns registros utilizam o “.” como separador de milhar e outros a “,”. O mesmo ocorre para separador de casas decimais. Desta forma, um método Python precisou ser feito para analisar dado por dado e aplicar a regra de correção mais adequada, convencionando todos os dados para nenhum separador de milhar e o uso do “.” como separador de decimais.

Para ter uma visão geral dos dados e identificar anomalias, é verificada a estatística sumarizada para avaliar mínimos, máximos, média e desvio padrão. Após os tratamentos descritos, nenhuma anomalia adicional foi identificada.

---

8 Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/6\\_etl\\_db\\_silver.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/6_etl_db_silver.ipynb)

### 4.2.3. ETL de otimização de dados (métricas – camada gold)

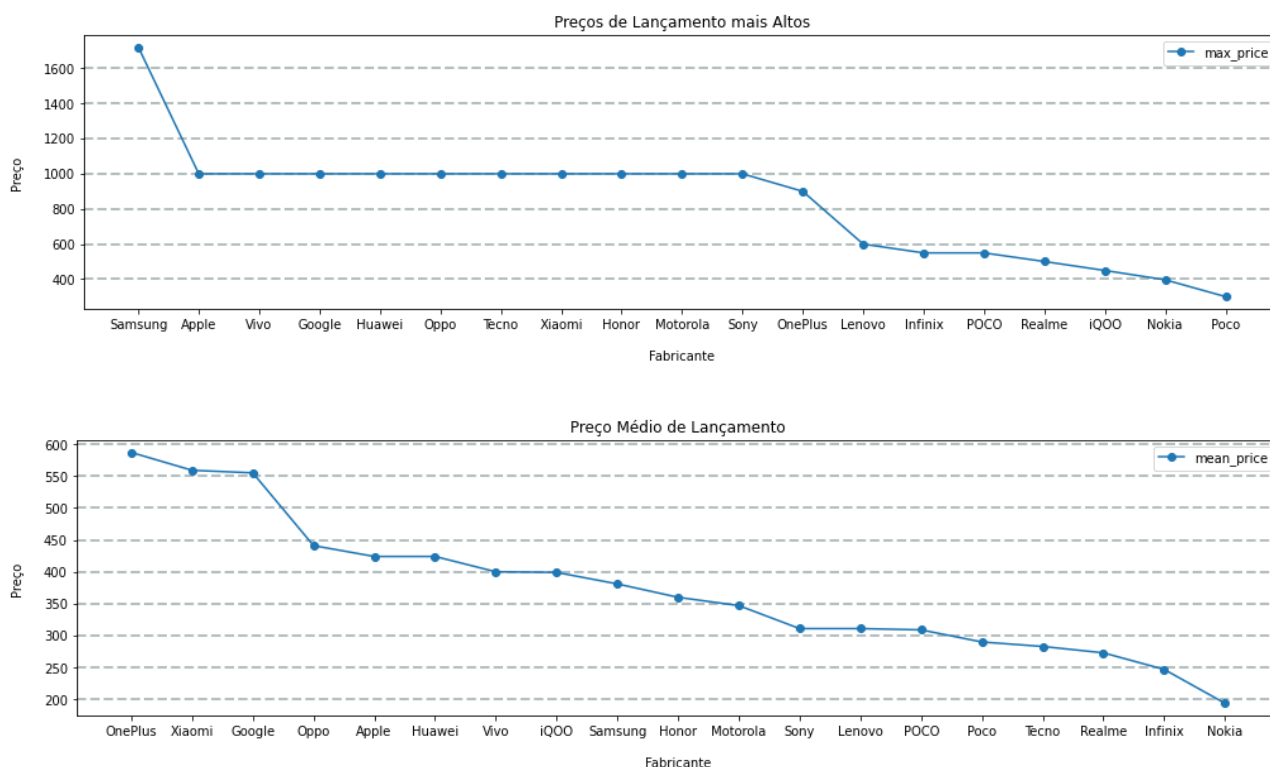
O ETL responsável pelo enriquecimento dos dados e criação de métricas a partir deles está disponível [neste notebook Python](#)<sup>9</sup>. O notebook carrega os dados a partir da camada silver, enriquece os dados e cria métricas de negócio que são persistidos na camada gold.

## 5. Autoavaliação:

A avaliação dos resultados é feita através das métricas e dos gráficos gerados a partir delas. Este trabalho é realizado [neste notebook Python](#)<sup>10</sup>.

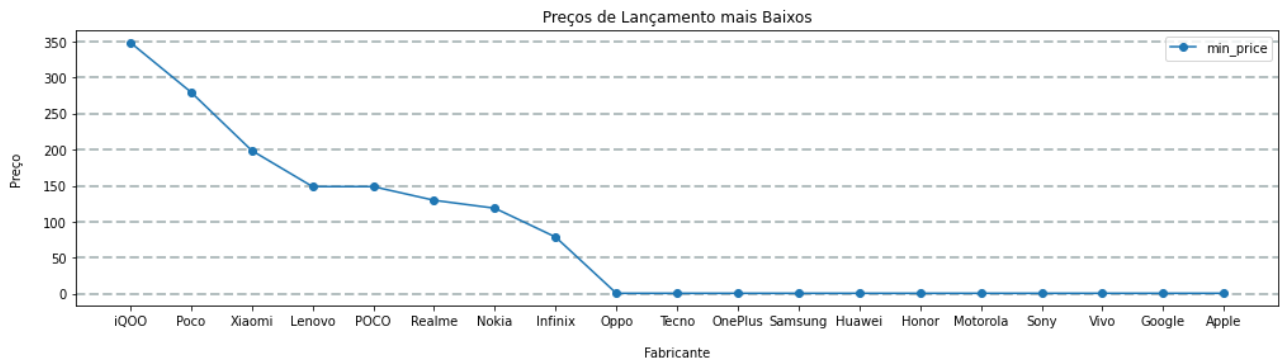
### 5.1. Métricas de Preço por Empresa

a) Preço médio, mais alto e mais baixo por empresa: Qual é o smartphone mais caro e mais barato em cada país?



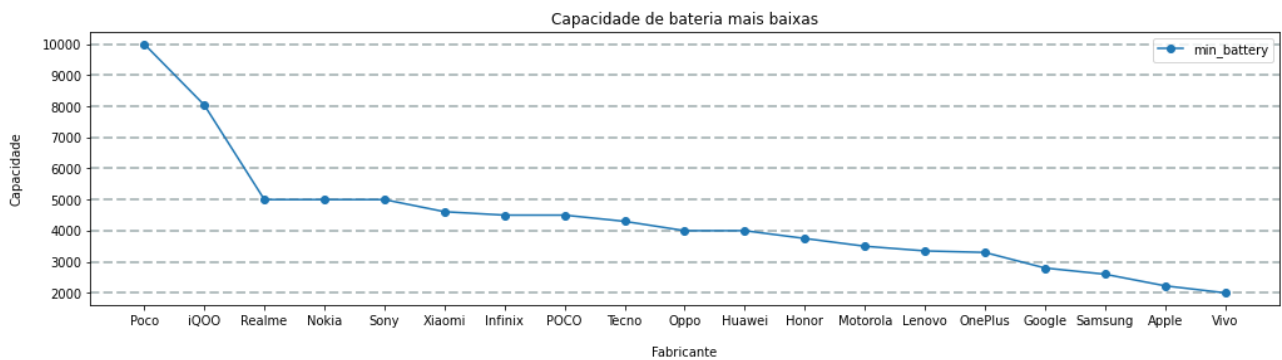
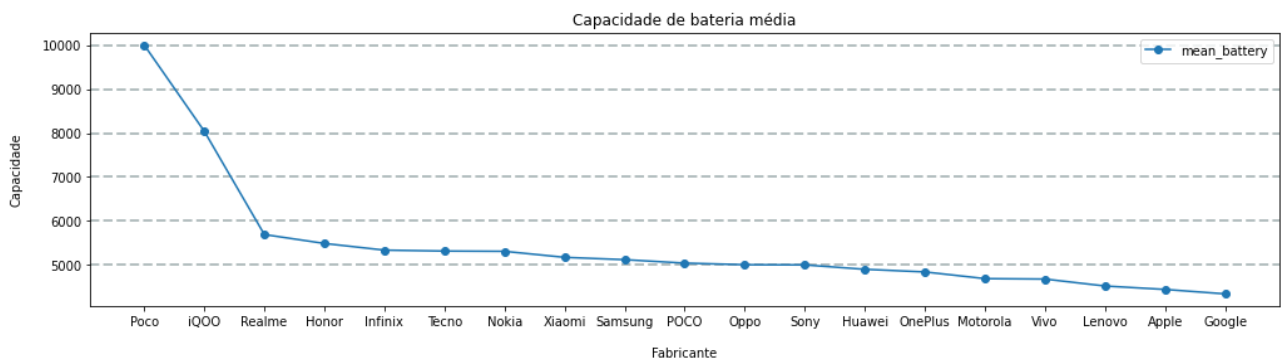
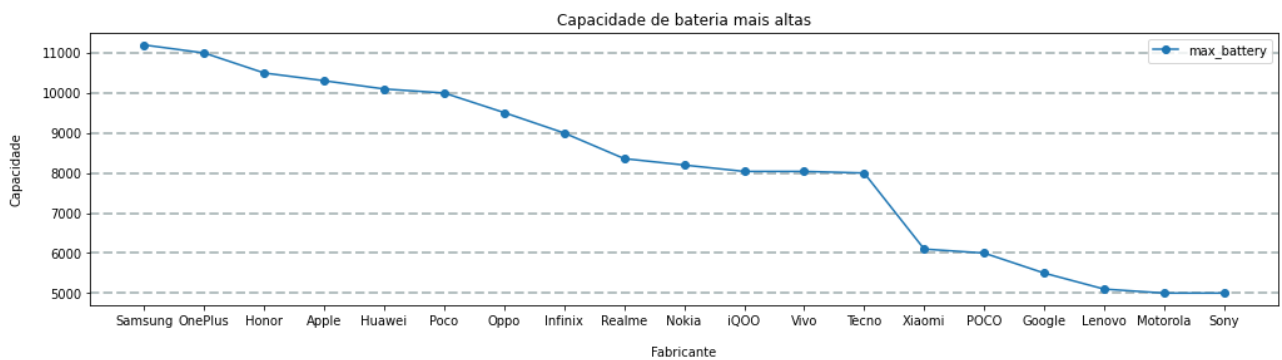
<sup>9</sup> Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/7\\_etl\\_db\\_gold.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/7_etl_db_gold.ipynb)

<sup>10</sup> Acessível em [https://github.com/edprata/pucrj\\_cellphones/blob/main/8\\_metrics\\_and\\_graphics.ipynb](https://github.com/edprata/pucrj_cellphones/blob/main/8_metrics_and_graphics.ipynb)

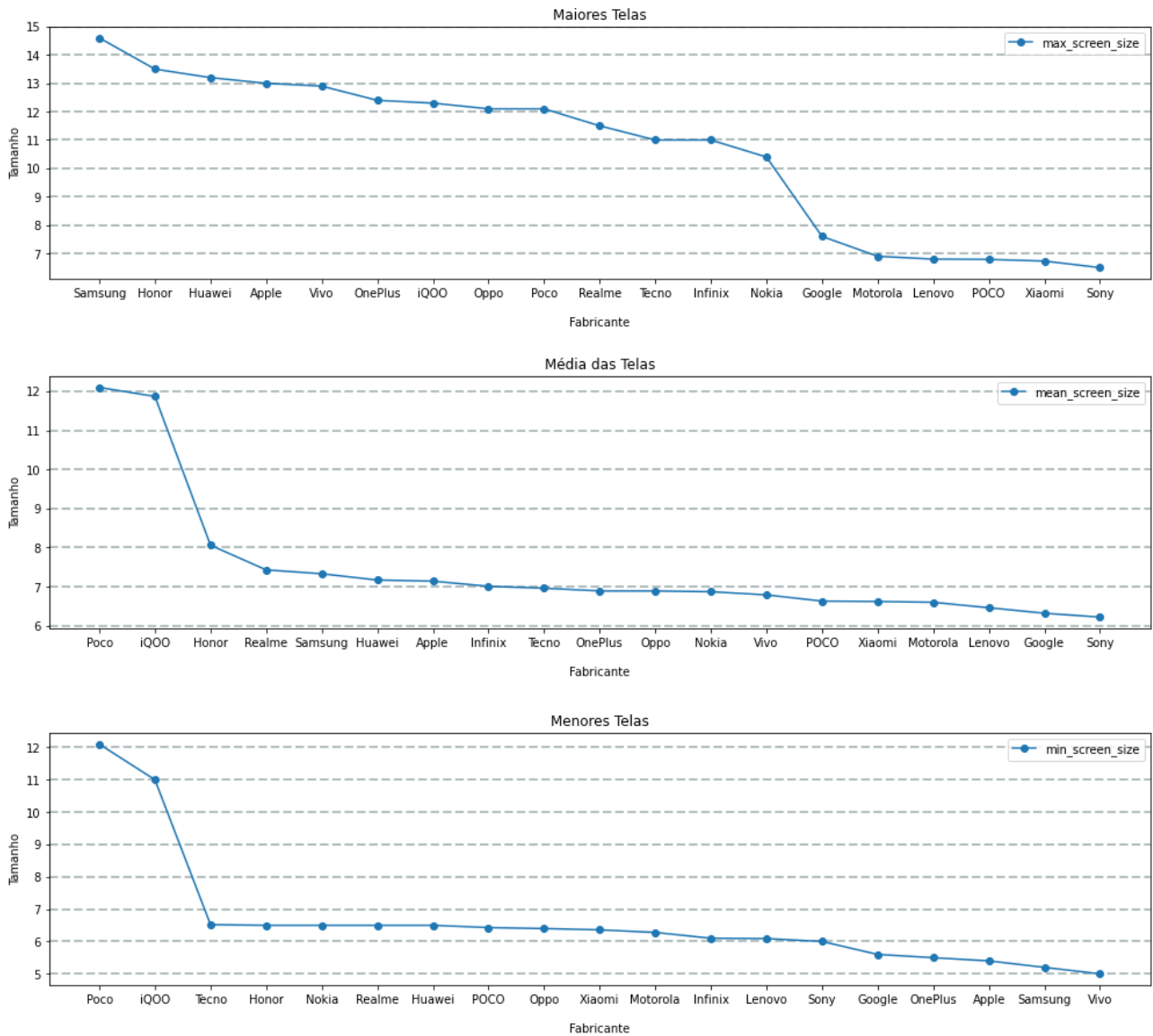


## 5.2. Métricas Técnicas por Empresa

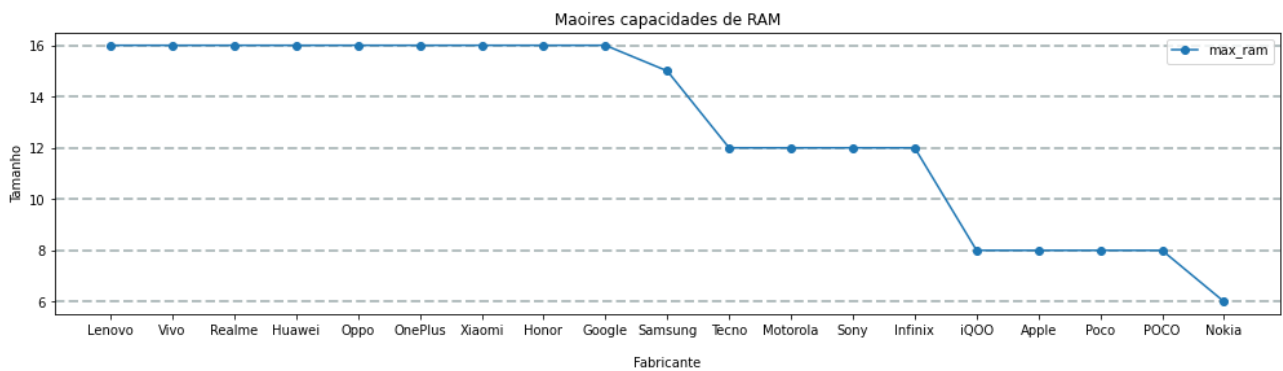
a) Média, mínimo e máximo da capacidade da bateria por empresa:

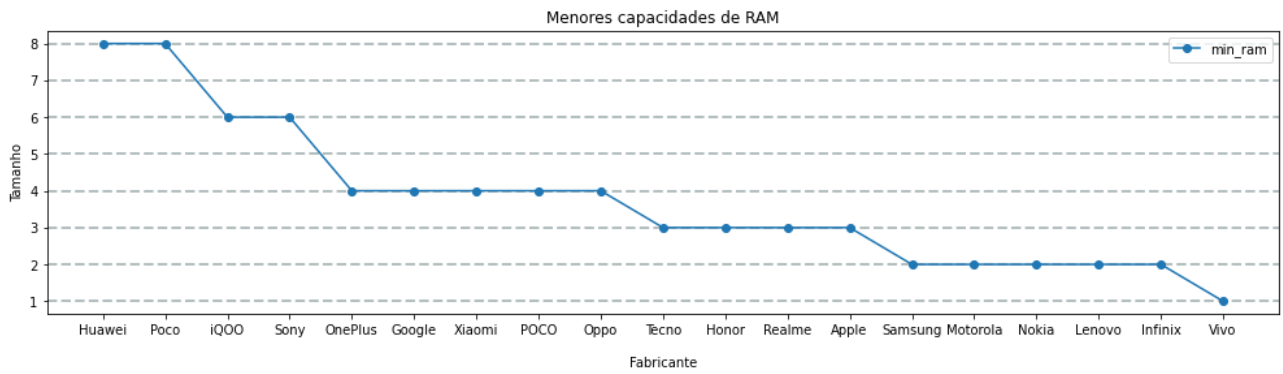
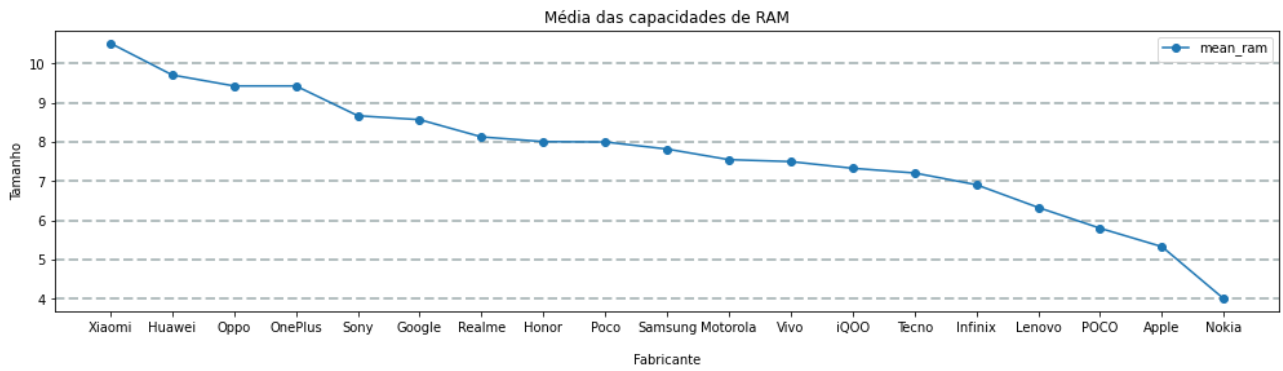


b) Média, mínimo e máximo do tamanho de tela dos smartphones por empresa:



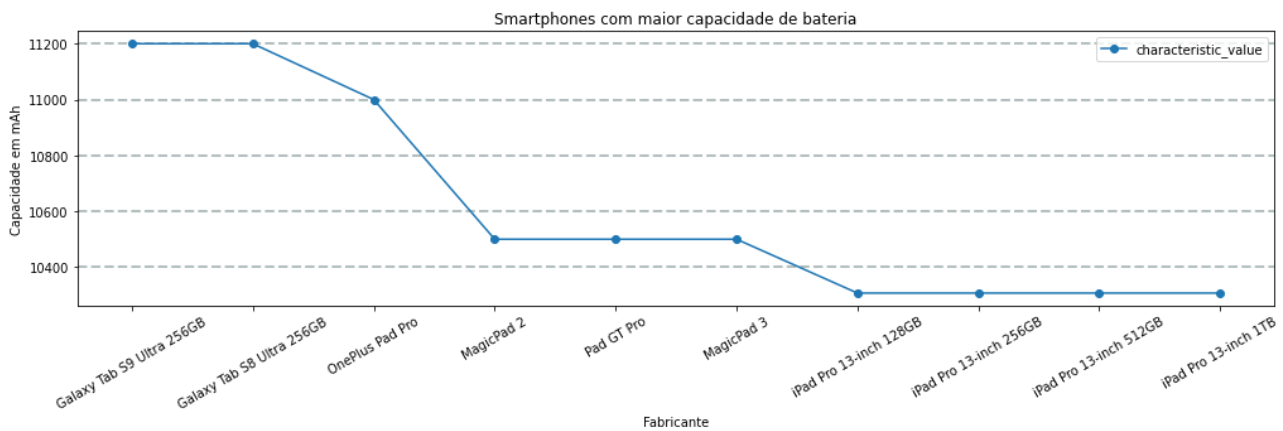
c) Média, mínimo e máximo da memória RAM dos smartphones por empresa:



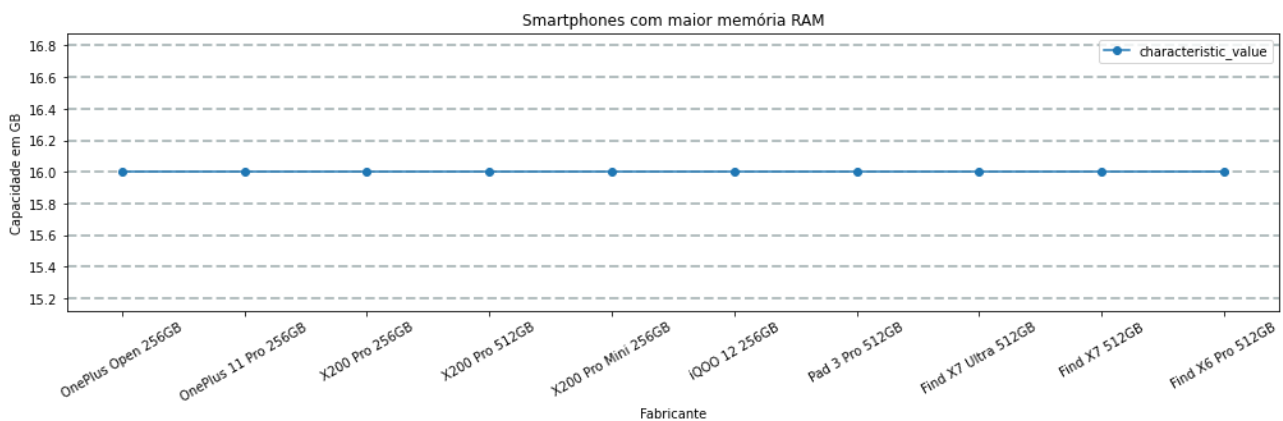


### 5.3. Métricas de Desempenho – TOP TEN

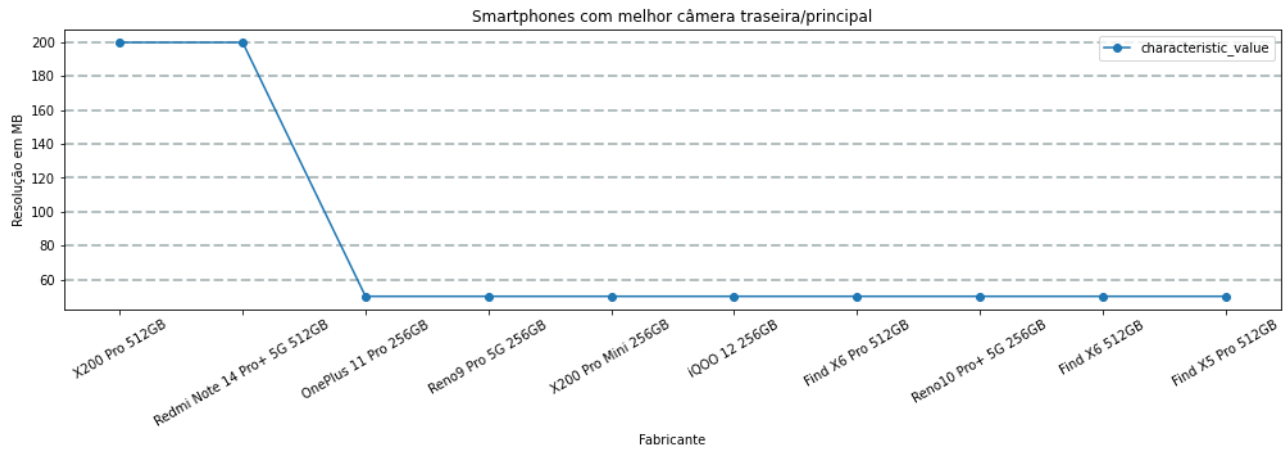
a) Smartphones com maior capacidade de bateria:



b) Smartphones com maior memória RAM:

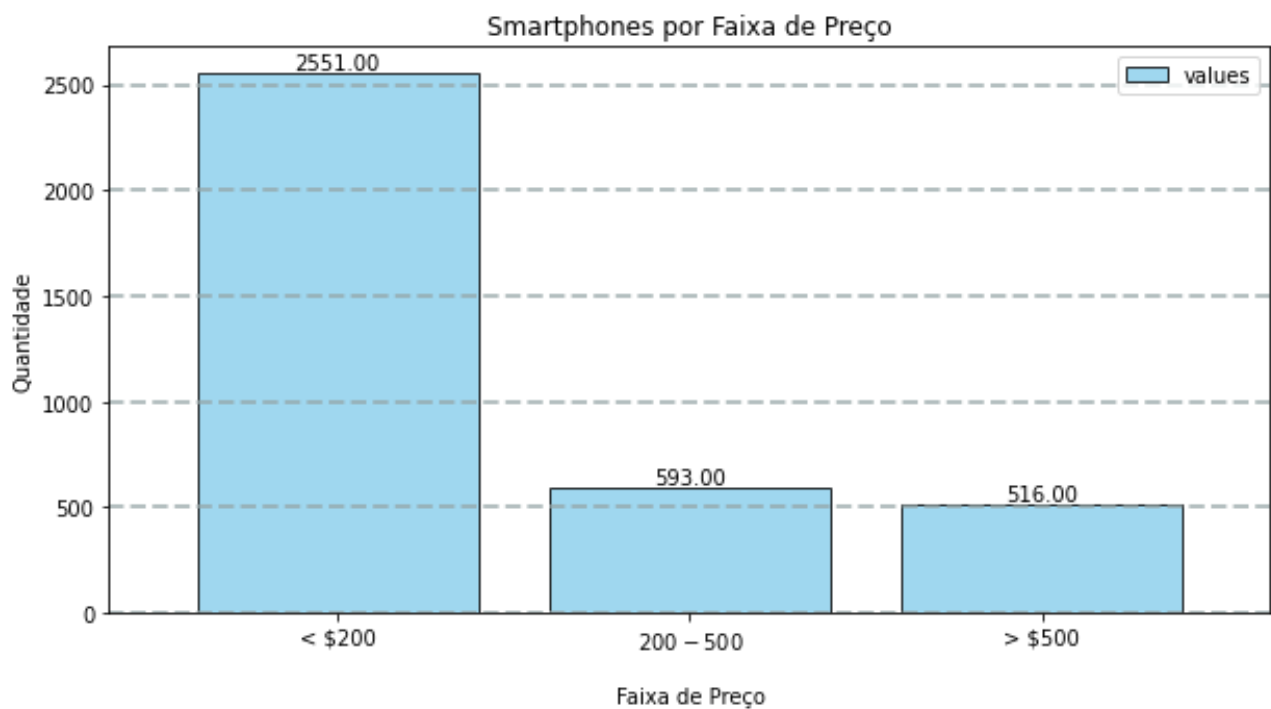


c) Smartphones com melhor câmera traseira:

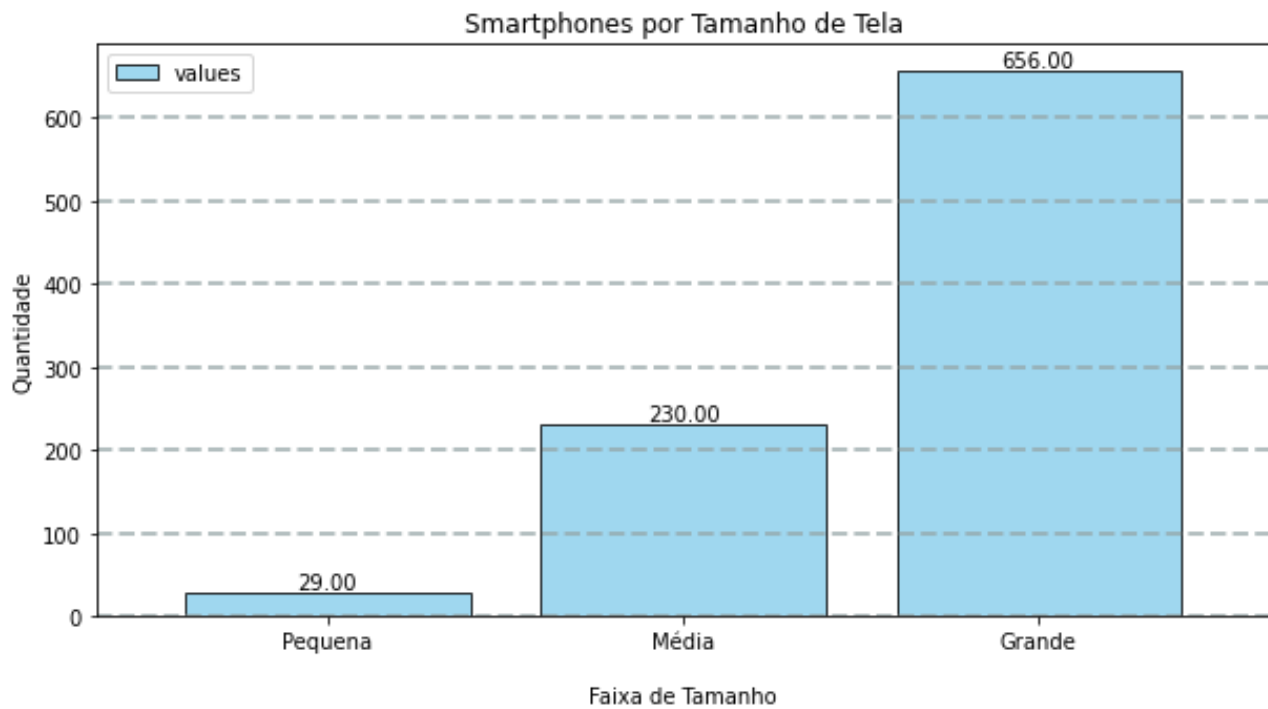


## 5.4. Métricas de Segmentação

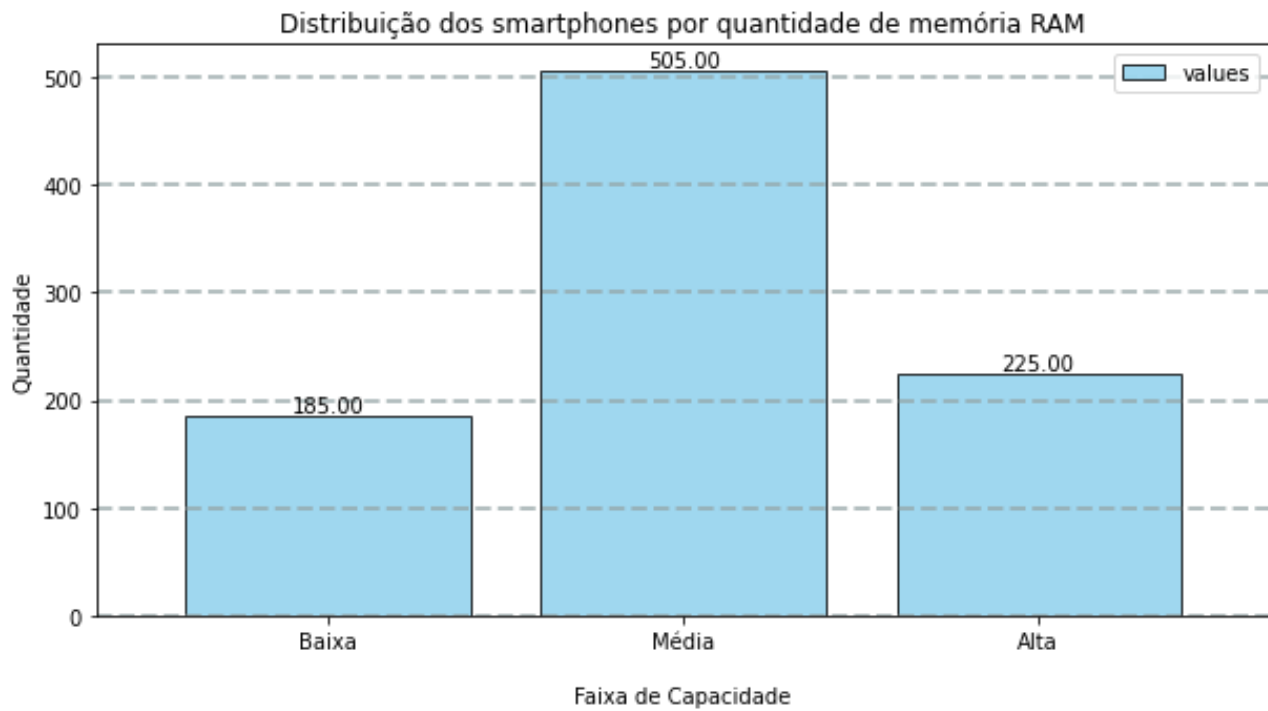
a) Smartphones por Faixa de Preço, sendo: abaixo de U\$ 200, entre U\$ 200 e U\$ 500 ou acima de U\$ 500:



b) Smartphones por Tamanho de Tela, sendo: pequena, sendo < 6"; médias, entre 6" e 6.5"; ou grandes, > 6.5":



d) Distribuição dos smartphones por quantidade de memória RAM:



## 5.5. Conclusões

Pelas **métricas de preço** observamos uma forte tendência ao lançamento de modelos mais sofisticados na faixa dos USD 1.000. Muitos fabricantes adotam esta estratégia, de forma que é interessante lançar um modelo com mais recursos nessa faixa. A média é bastante variada e fica muito prejudicada devido ao mínimo de muitos fabricantes ser zero, o que indica ausência do preço de lançamento para alguns modelos, sendo interessante uma nova análise sem estes modelos para os quais falta o preço. Há também modelos da Samsung que saem muito do padrão, alcançando preços de lançamento perto de USD 2.000 - seria interessante verificar que modelo é esse para saber se é interessante considerá-lo na análise, mas a tendência maior é descartar, pois o objetivo de negócio é estabelecer características técnicas e de preço mais comuns para o lançamento de um produto dentro dos padrões de mercado. O valor para um modelo de entrada fica de difícil definição devido valores ausentes e média prejudicada. É preciso tratar este problema e refazer a análise.

Sobre as características técnicas, alguns elementos chamam mais a atenção, como bateria, memória RAM, tamanho da tela e resolução da câmera principal ou câmera traseira.

A **capacidade das baterias** têm uma faixa dinâmica de distribuição bastante ampla, indo de 2.000 mAh a 11.000 mAh. Os modelos mais sofisticados partem da faixa dos 5.000 mAh, demonstrando um salto significativo com relação a modelos mais antigos. Vemos uma forte tendência a baterias mais robustas para dar suporte as demandas dos usuários, cada vez mais conectados e utilizando os smartphones por períodos prolongados. A Apple é um fabricante que já sofreu duras críticas a capacidade das baterias de seus modelos, mas hoje aparece com modelos bastante potentes nesse quesito, apresentando baterias com até mais de 10.000 mAh - apesar de a média dos modelos ser ainda bastante baixa. Pelos dados, vê-se uma média geral bastante estável, por volta dos 5.000 mAh. O que dá indícios de ser uma faixa interessante para a construção de novos modelos.

O **tamanho das tela** é bastante variável. Provavelmente há modelos de tablet misturados com modelos de smartphones na base, pois as máximas mostram modelos de até 15 polegadas. Além disso, cerca de 2/3 do gráfico de máximas mostra modelos com telas acima de 11 polegadas, o que é muito para um smartphone. Isso prejudica a média geral, levando-a para próximo de 7 polegadas, bastante perto do gráfico de valores mínimos, o que demonstra um forte desvio padrão na amostra. É preciso aprofundar a análise sobre as telas para chegar a uma conclusão mais assertiva.



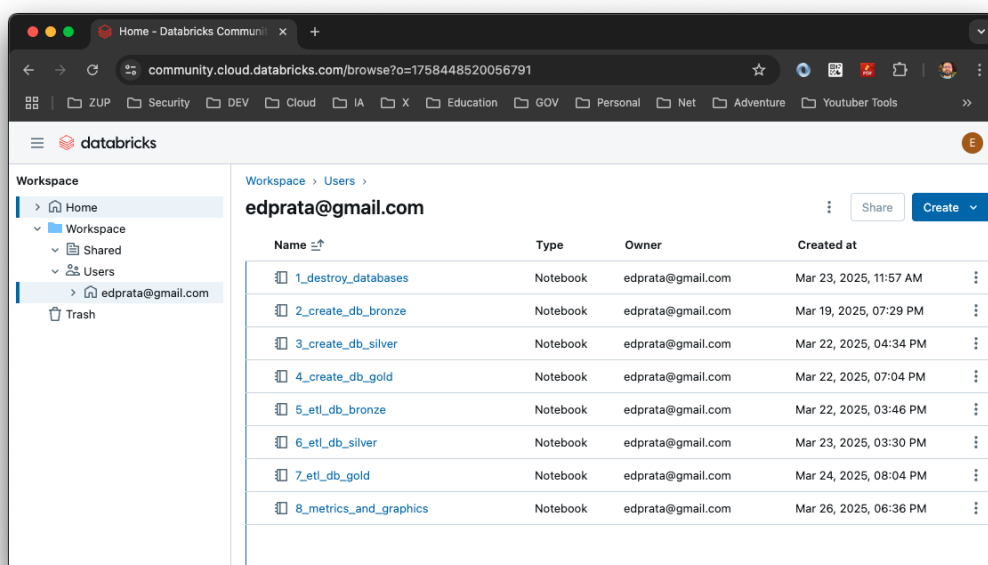
A análise da capacidade de **memória RAM** mostra uma faixa de distribuição dinâmica curta, variando entre 2 GB e 16 GB, uma variação de 14 GB. Apesar disso, sabemos que cada 1 GB a mais de memória RAM é algo bastante significativo em termos práticos. Percebe-se que poucos fabricantes ainda trabalham com memórias de 2 GB, estando a grande maioria partindo de modelos com 3 GB ou superior, o que podemos entender como um padrão mínimo de mercado, mesmo para modelos de entrada nos dias atuais. Metade do gráfico apresenta modelos com 4 GB ou mais. Para lançar modelos de entrada, em competição com o mercado, talvez fosse 4 GB o mínimo recomendado. Já as máximas chegam a 16 GB, sendo que metade dos fabricantes possuem modelos com essa capacidade e mais de 2/3 oferecem modelos mais sofisticados a partir de 12 GB. Logo, o lançamento de um modelo nessa categoria mostra-se de grande importância.

Algumas características necessitam aprofundamento, mais tratamentos ou mais dados, porém, já podemos indicar, a partir desta pesquisa, alguns insights interessantes:

1. Proposta de modelo de entrada (menor custo): Pelo menos 5.000 mAh de bateria e 4 GB de memória RAM.
2. Proposta de modelo mais sofisticado (maior custo): Preço de lançamento em torno de USD 1.000, pelo menos 12 GB de memória RAM e bateria acima de 8.000 mAh.

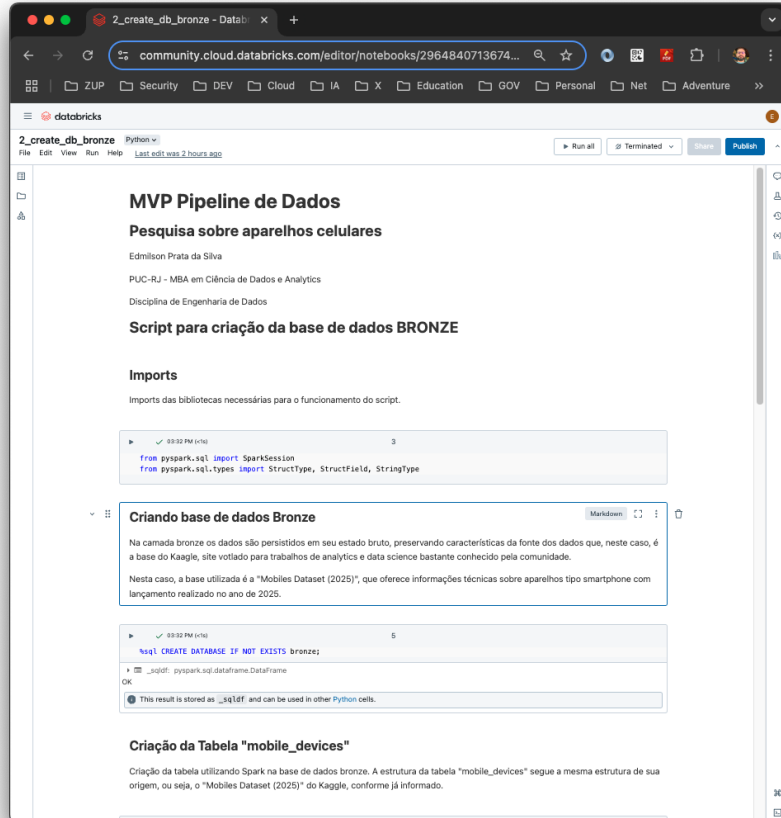
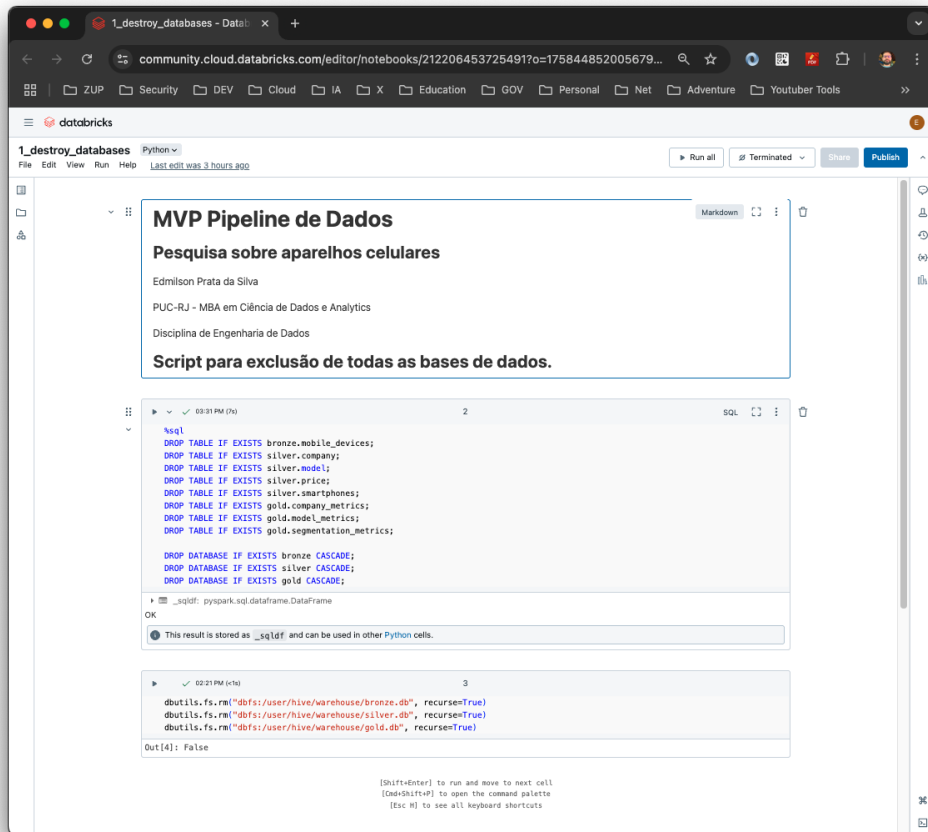
## 6. Evidências do Trabalho

A seguir são apresentadas as evidências de execução da pipeline de dados no ambiente de Cloud Databricks:



The screenshot shows the Databricks web interface. On the left, the 'Workspace' sidebar is visible with a tree view containing 'Home', 'Workspace', 'Shared', 'Users', and 'edprata@gmail.com'. The main area displays the 'Workspace > Users > edprata@gmail.com' page. It features a table of notebooks with columns for Name, Type, Owner, and Created at. The table lists eight notebooks, all of type 'Notebook' and owned by 'edprata@gmail.com'. The notebooks are named sequentially from 1 to 8, representing steps in a data pipeline. The 'Created at' column shows the timestamps for each notebook's creation.

Name	Type	Owner	Created at
1_destroy_databases	Notebook	edprata@gmail.com	Mar 23, 2025, 11:57 AM
2_create_db_bronze	Notebook	edprata@gmail.com	Mar 19, 2025, 07:29 PM
3_create_db_silver	Notebook	edprata@gmail.com	Mar 22, 2025, 04:34 PM
4_create_db_gold	Notebook	edprata@gmail.com	Mar 22, 2025, 07:04 PM
5_etl_db_bronze	Notebook	edprata@gmail.com	Mar 22, 2025, 03:46 PM
6_etl_db_silver	Notebook	edprata@gmail.com	Mar 23, 2025, 03:30 PM
7_etl_db_gold	Notebook	edprata@gmail.com	Mar 24, 2025, 08:04 PM
8_metrics_and_graphics	Notebook	edprata@gmail.com	Mar 26, 2025, 06:36 PM



The screenshot shows a Databricks notebook interface with the title "3\_create\_db\_silver". The notebook content includes:

- MVP Pipeline de Dados**  
**Pesquisa sobre aparelhos celulares**  
Edmilson Prata da Silva  
PUC-RJ - MBA em Ciência de Dados e Analytics  
Disciplina de Engenharia de Dados
- Script para criação da base de dados SILVER**
- Imports**  
Imports das bibliotecas necessárias para o funcionamento do script.
- Code Cell 3:** Python code importing SparkSession and SQL types.

```
from pyspark.sql import SparkSession
from pyspark.sql.types import StructType, StructField, StringType, DecimalType
```
- Criando base de dados SILVER**  
Na camada silver os dados são persistidos após terem sido tratados, transformados e enriquecidos para o trabalho de analytics. Nesta caso, a base original é a "Mobiles Dataset (2025)", do Kaggle. Ela foi persistida, mantendo suas características originais na camada bronze. Após, os dados sofrerão os tratamentos necessários para armazenamento nesta camada.
- Code Cell 5:** SQL command to create the database.

```
%%sql
CREATE DATABASE IF NOT EXISTS silver;
```
- Criando as tabelas**  
A seguir o script para criação das tabelas da camada silver.
- Code Cell 7:** Python code starting the Silver layer creation.

```
print("Iniciando criação da camada silver...")
```

The screenshot shows a Databricks notebook interface with the title "4\_create\_db\_gold". The notebook content includes:

- MVP Pipeline de Dados**  
**Pesquisa sobre aparelhos celulares**  
Edmilson Prata da Silva  
PUC-RJ - MBA em Ciência de Dados e Analytics  
Disciplina de Engenharia de Dados
- Script para criação da base de dados GOLD**
- Imports**  
Imports das bibliotecas necessárias para o funcionamento do script.
- Code Cell 3:** Python code importing SparkSession and SQL types.

```
from pyspark.sql import SparkSession
from pyspark.sql.types import StructType, StructField, StringType, DecimalType
```
- Criando base de dados GOLD**  
A camada Gold contém dados altamente refinados e otimizados para consumo final. Esses dados são agregados, enriquecidos e organizados para atender necessidades específicas de negócio.
- Code Cell 5:** SQL command to create the database.

```
%%sql
CREATE DATABASE IF NOT EXISTS gold;
```
- Criando as tabelas**  
A seguir o script para criação das tabelas da camada gold.
- Code Cell 7:** Python code starting the Gold layer creation.

```
print("Iniciando criação da camada gold...")
# Iniciar uma sessão Spark
```

5\_etl\_db\_bronze - Databricks

community.cloud.databricks.com/editor/notebooks/1066694302692...

5\_etl\_db\_bronze Python

File Edit View Run Help Last edit was 4 days ago

Run all Terminated Share Publish

## MVP Pipeline de Dados

### Pesquisa sobre aparelhos celulares

Edmilson Prata da Silva

PUC-RJ - MBA em Ciência de Dados e Analytics

Disciplina de Engenharia de Dados

### Script para carga da camada BRONZE

#### Imports

Imports das bibliotecas necessárias para o funcionamento do script.

```
import pandas as pd
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
```

#### Carga de Dados

Os dados serão recuperados do GitHub, repositório público. O arquivo foi copiado do Kaggle para o GitHub devido o acesso ao Kaggle apresentando instabilidades durante tentativas de baixar diretamente.

Após carregados, as colunas são renomeadas, conforme o padrão da tabela. Em seguida, os dados são persistidos sem alterações, conforme padrão da camada bronze.

```
df_pandas = pd.read_csv(
    'https://github.com/edprata/pucrj_cellphones/raw/refs/heads/main/kaggle_mobile_dataset_2025.csv',
    sep=',', encoding='latin-1', skip_blank_lines=True, on_bad_lines='skip'
)
df_pandas.shape
Out[12]: (930, 15)
```

```
# Converte o DataFrame do Pandas em um DataFrame do Spark
df_spark = spark.createDataFrame(df_pandas)

# Exibe o schema do DataFrame do Spark
df_spark.printSchema()
```

6\_etl\_db\_silver - Databricks

community.cloud.databricks.com/editor/notebooks/3760860936721...

6\_etl\_db\_silver Python

File Edit View Run Help Last edit was 3 hours ago

Run all Terminated Share Publish

## MVP Pipeline de Dados

### Pesquisa sobre aparelhos celulares

Edmilson Prata da Silva

PUC-RJ - MBA em Ciência de Dados e Analytics

Disciplina de Engenharia de Dados

### Script ETL para carga na camada SILVER

#### Imports

Imports das bibliotecas necessárias para o funcionamento do script.

```
import re
import os
import warnings
import pandas as pd
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
```

#### Carga de Dados

Os dados serão carregados a partir da camada bronze para os tratamentos necessários.

```
spark = SparkSession.builder.getOrCreate()
df_spark = spark.table("bronze.mobile_devices")
df_spark.printSchema()
```

```
root
 |-- company_name: string (nullable = true)
 |-- model_name: string (nullable = true)
 |-- mobile_weight: string (nullable = true)
 |-- ram: string (nullable = true)
 |-- front_camera: string (nullable = true)
 |-- back_camera: string (nullable = true)
 |-- processor: string (nullable = true)
 |-- battery_capacity: string (nullable = true)
 |-- screen_size: string (nullable = true)
 |-- launched_price_pakistan: string (nullable = true)
```

7\_etl\_db\_gold - Databricks

community.cloud.databricks.com/editor/notebooks/2555020261803...

ZUPSecurityDEVCloudIAXEducationGOVPersonalNetAdventure

databricks

7\_etl\_db\_goldPython

FileEditViewRunHelpLast edit was 2 days ago

Run allTerminatedSharePublish

MVP Pipeline de Dados

Pesquisa sobre aparelhos celulares

Edmilson Prata da Silva

PUC-RJ - MBA em Ciência de Dados e Analytics

Disciplina de Engenharia de Dados

Script ETL para carga na camada GOLD

Imports

Imports das bibliotecas necessárias para o funcionamento do script.

03:33 PM (UTC)

3

import re  
import uuid  
import warnings  
import pandas as pd  
from pyspark.sql import SparkSession  
from pyspark.sql.functions import col

Carga de Dados

Os dados serão carregados a partir da camada silver gerando as métricas a serem persistidas na camada gold.

03:33 PM (UTC)

5

spark\_company\_metrics = spark.sql("""  
SELECT c.company\_id  
 , c.company\_name  
 , INT(AVG(s.launched\_price)) AS mean\_price  
 , INT(MAX(s.launched\_price)) AS max\_price  
 , INT(MIN(s.launched\_price)) AS min\_price  
 , INT(AVG(s.battery\_capacity)) AS mean\_battery  
 , INT(MIN(s.battery\_capacity)) AS min\_battery  
 , INT(MAX(s.battery\_capacity)) AS max\_battery  
 , AVG(s.screen\_size) AS mean\_screen\_size  
 , MIN(s.screen\_size) AS min\_screen\_size  
 , MAX(s.screen\_size) AS max\_screen\_size  
 , AVG(s.ram) AS mean\_ram  
 , MIN(s.ram) AS min\_ram  
 , MAX(s.ram) AS max\_ram  
FROM silver.smartphones s  
JOIN silver.company c ON c.company\_id = s.company\_id

8\_metrics\_and\_graphics - D

community.cloud.databricks.com/editor/notebooks/4052410975436...

ZUPSecurityDEVCloudIAXEducationGOVPersonalNetAdventure

databricks

8\_metrics\_and\_graphicsPython

FileEditViewRunHelpLast edit was 2 hours ago

Run allTerminatedSharePublish

MVP Pipeline de Dados

Pesquisa sobre aparelhos celulares

Edmilson Prata da Silva

PUC-RJ - MBA em Ciência de Dados e Analytics

Disciplina de Engenharia de Dados

Métricas e Gráficos

Imports

Imports das bibliotecas necessárias para o funcionamento do script.

03:34 PM (UTC)

3

import re  
import uuid  
import warnings  
import pandas as pd  
import seaborn as sb  
import matplotlib.pyplot as plt  
from pyspark.sql import SparkSession  
from pyspark.sql.functions import col

Carga de Dados

Os dados serão carregados a partir da camada gold na qual as métricas já estão prontas para consumo.

03:34 PM (UTC)

5

spark = SparkSession.builder.getOrCreate()  
company\_metrics = spark.table("gold.company\_metrics")  
company\_metrics.printSchema()  
  
company\_metrics: pyspark.sql.dataframe.DataFrame = [company\_id: string, company\_name: string, ... 12 more fields]  
root  
|-- company\_id: string (nullable = true)  
|-- company\_name: string (nullable = true)  
|-- mean\_price: integer (nullable = true)  
|-- max\_price: integer (nullable = true)  
|-- min\_price: integer (nullable = true)  
|-- mean\_battery: integer (nullable = true)  
|-- max\_battery: integer (nullable = true)  
|-- min\_battery: integer (nullable = true)  
|-- mean\_screen\_size: decimal(10,2) (nullable = true)