

GABRIEL SILVA CASE

DESAFI0 1

1. Analyze the data provided and present your conclusions .
2. In addition to the spreadsheet data, make a query in SQL and make a graphic of it and try to explain the anomaly behavior you found.
3. In [this csv](#) you have the number of sales of POS by hour comparing the same sales per hour from today, yesterday and the average of other days. So with this we can see the behavior from today and compare to other days

time	today	yesterday	same_day_last_week	avg_last_week	avg_last_month
00h	9	12	11	6.42	4.85
01h	3	5	1	1.85	1.92
02h	1	0	0	0.28	0.82
03h	1	0	0	0.42	0.46
04h	0	0	1	0.42	0.21
05h	1	1	2	1.28	0.75
06h	1	1	5	2.85	2.28
07h	2	3	9	5.57	5.21
08h	0	1	18	8.71	10.42
09h	2	9	30	20.0	19.07
10h	55	51	45	29.42	28.35
11h	36	44	38	33.71	28.5
12h	51	39	39	27.57	25.42
13h	36	41	43	25.85	24.21
14h	32	35	36	26.14	25.21
15h	51	35	49	28.14	27.71



CÓDIGO DESAFIO 1

Disponível em: [case_google_colab](#)

SQLite

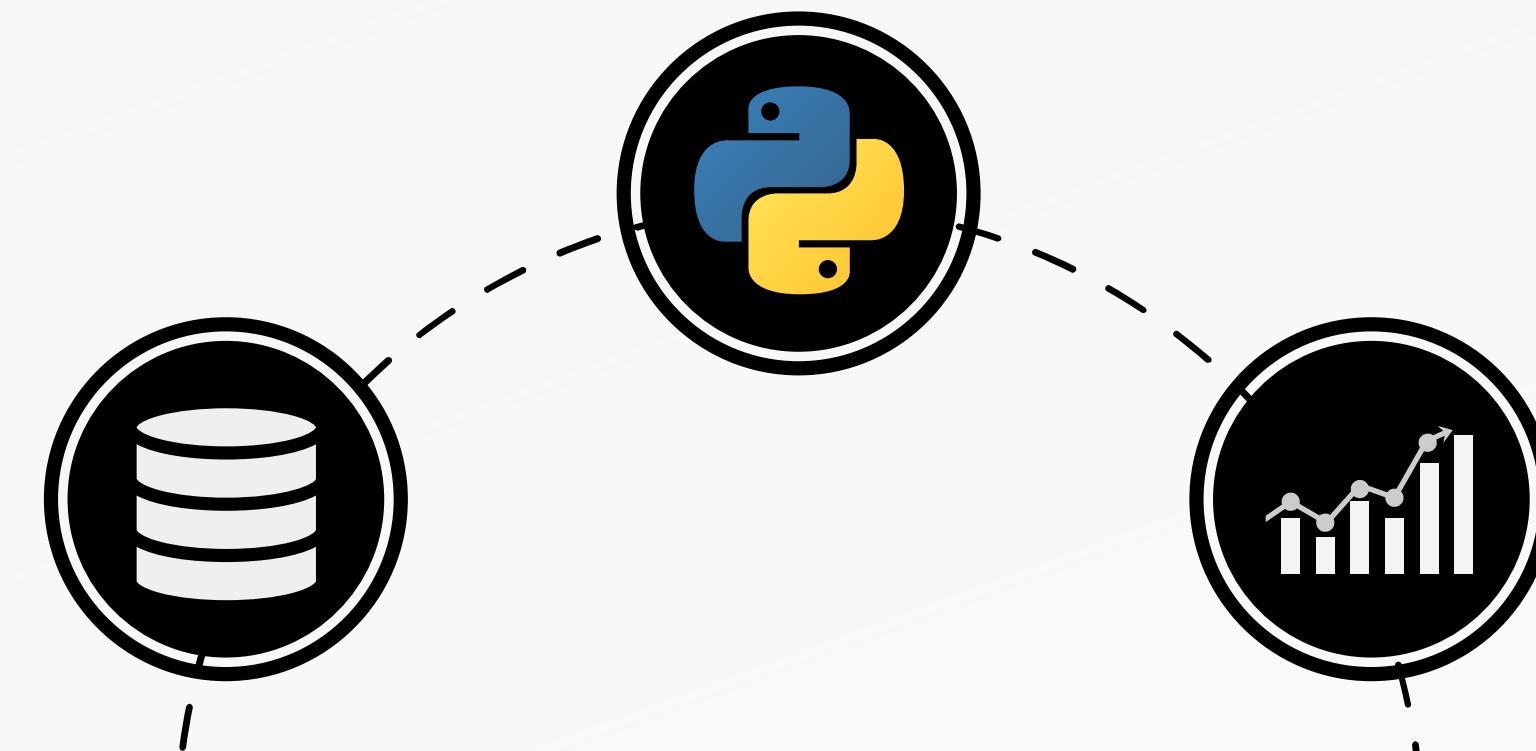
Setup de um ambiente SQLite para consultar os dados usando SQL direto no google colab. Query: diferença absoluta entre as vendas de hoje e das outras colunas. A ideia é identificar se em algum horário hoje as vendas foram muito maiores ou muito menores que nos outros dias/médias.

Python

Utilizando pandas para puxar os dados do github e transformar o csv em dataframe, foi possível criar gráficos para ilustrar os dados de hoje e dos outros dias/médias, bem como os dados de diferenças da query feita em SQL.

Visualização

Com a biblioteca matplotlib, os gráficos da query e do csv foram gerados para análise visual dos eventos. A análise visual facilita a identificação de anomalias por contraste.



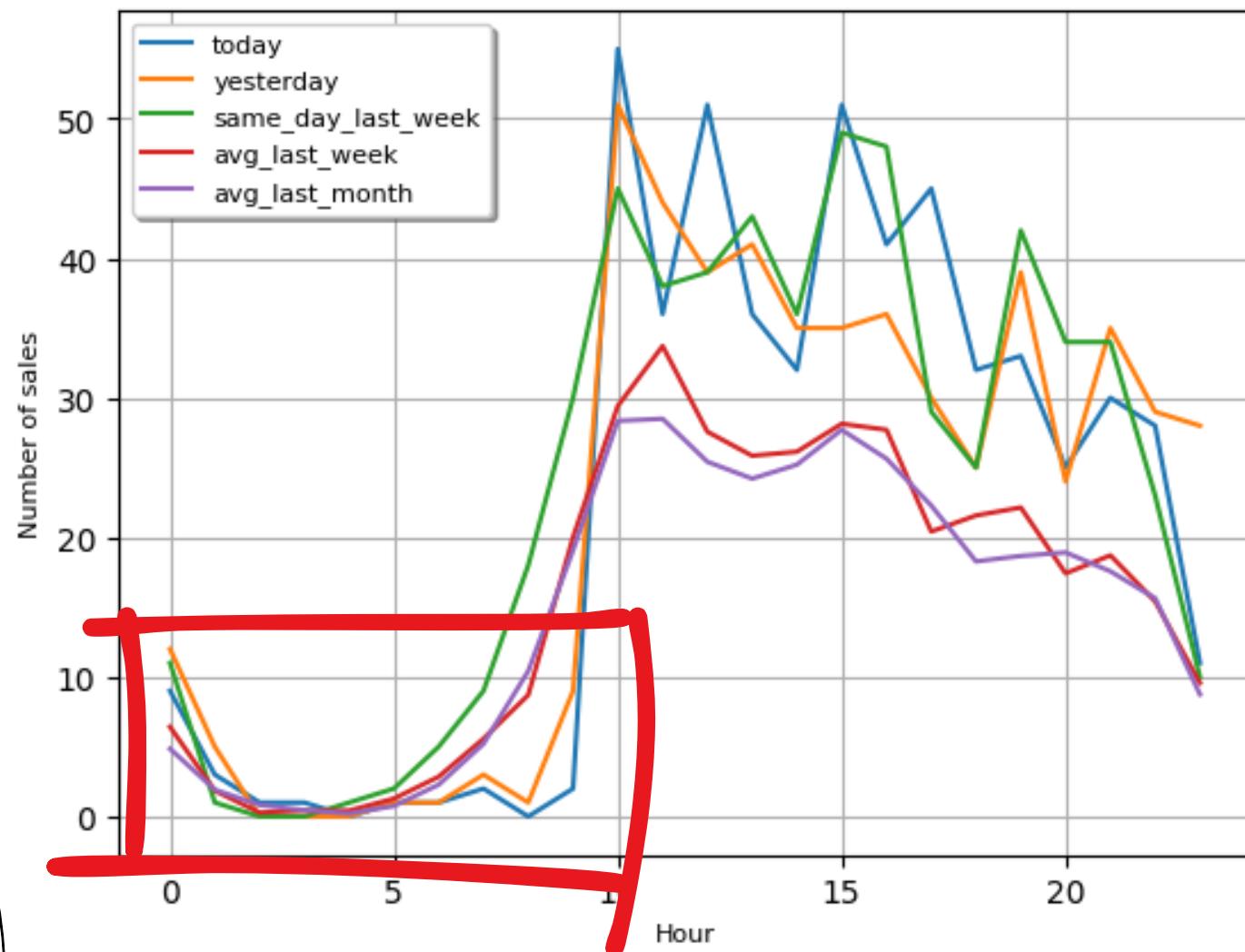
ANÁLISE DESAFIO 1 - CHECKOUT 1

Disponível em: [case_google_colab](#)

Visualização



Comparison between today and the other data points [checkout 1]



O gráfico ao lado ilustra os dados presentes no csv checkout_1. É possível ver que:

- Não há muita movimentação antes das 5h
- Especialmente hoje, não há muita movimentação até próximo das 9h. Às 9h hoje possui o menor número de vendas de todas as curvas, já às 10h, possui o maior. Isso gera uma suspeita.
- Em comparação com ontem, o comportamento de baixas vendas até às 8h é compatível

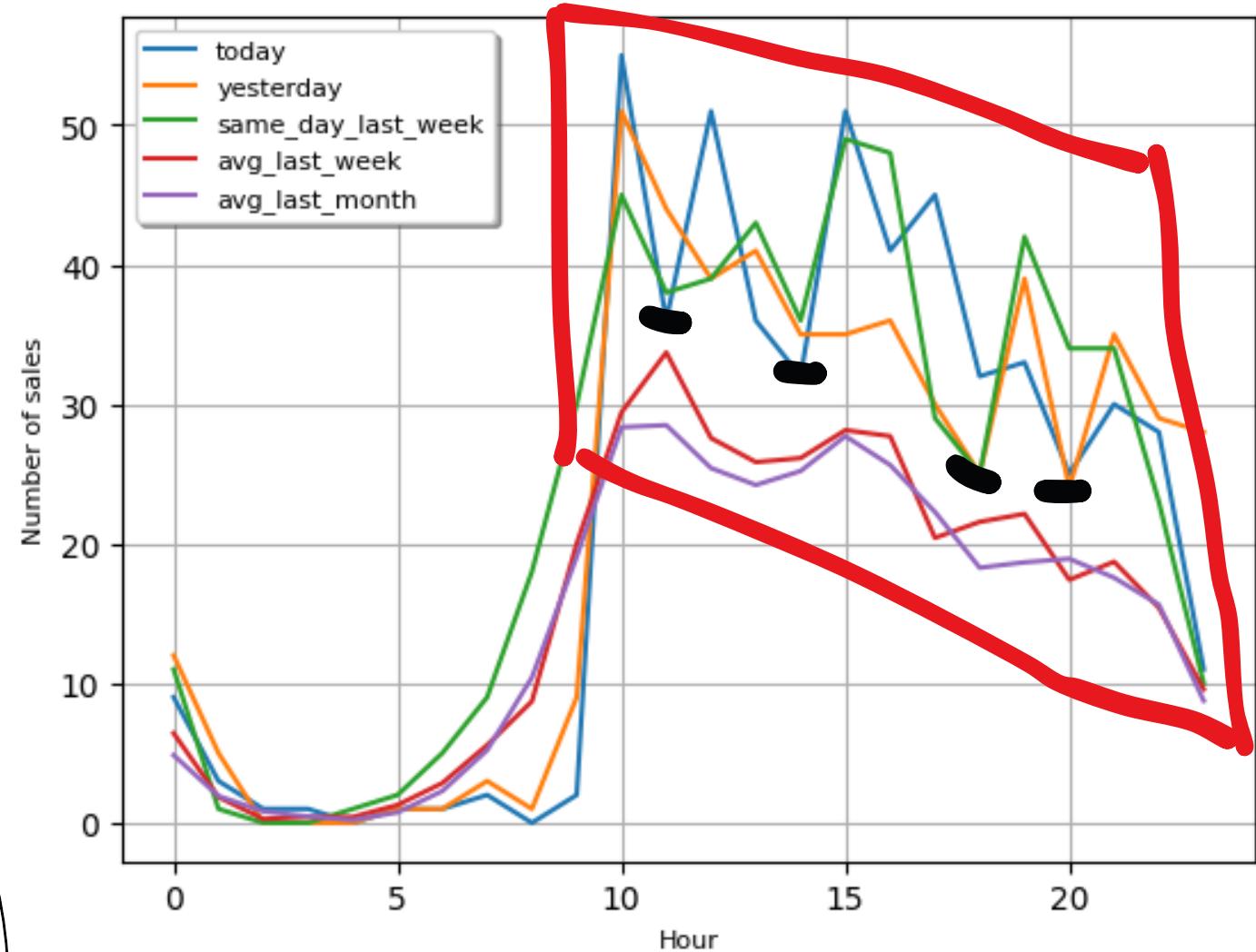
ANÁLISE DESAFIO 1 - CHECKOUT 1

Disponível em: [case_google_colab](#)

Visualização



Comparison between today and the other data points [checkout 1]



O gráfico ao lado ilustra os dados presentes no csv checkout_1. É possível ver que:

- Após as 10h todas as curvas tem perfil parecido, em relação à tendência, porém, as curvas de hoje, ontem e o mesmo dia da semana passada estão significativamente acima das demais (média semana passada e mês passado). Isso pode indicar que o negócio está em uma tendência de crescimento de vendas nestes horários, já que até mesmo os pontos mais baixos dessas curvas estão acima da média das curvas de dados mais antigos.

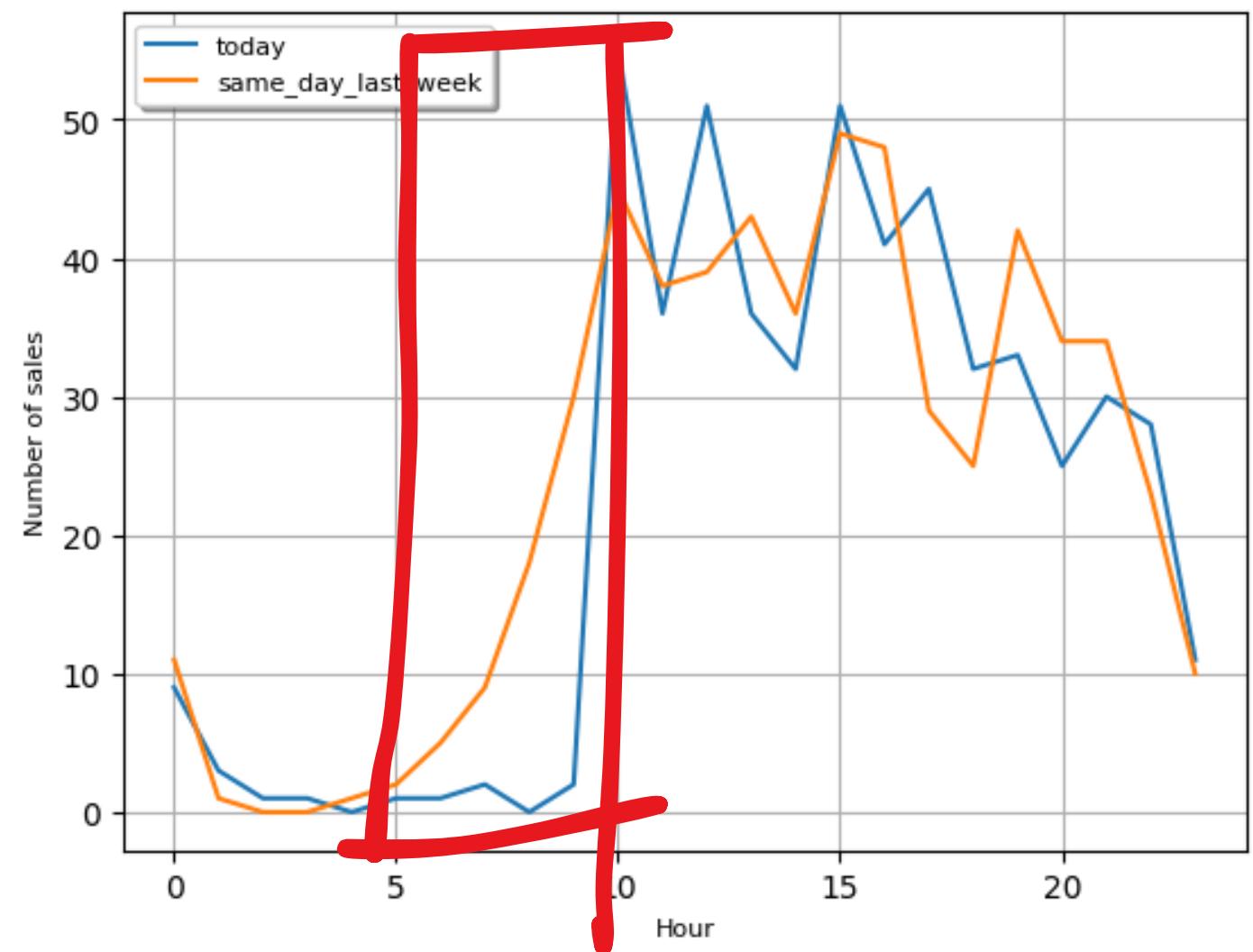
ANÁLISE DESAFIO 1 - CHECKOUT 1

Disponível em: [case_google_colab](#)

Visualização



Comparison between today and the same day last week [checkout 1]



O gráfico ao lado ilustra os dados presentes no csv checkout_1. É possível ver que:

- Comparar hoje com ontem pode levar a um falso alerta, já que muitos negócios tem comportamento distinto de vendas ao longo dos dias da semana
- Sem mais detalhes sobre o perfil do cliente e o modelo de negócio, vamos adotar um padrão semanal, e o ponto de contraste a ser considerado será entre hoje e o mesmo dia da semana passada
- Já observamos que parece haver um crescimento nas vendas WoW a MoM, através das médias. Assumindo isso, o período entre 5h e 9h de hoje e da semana passada chama atenção.

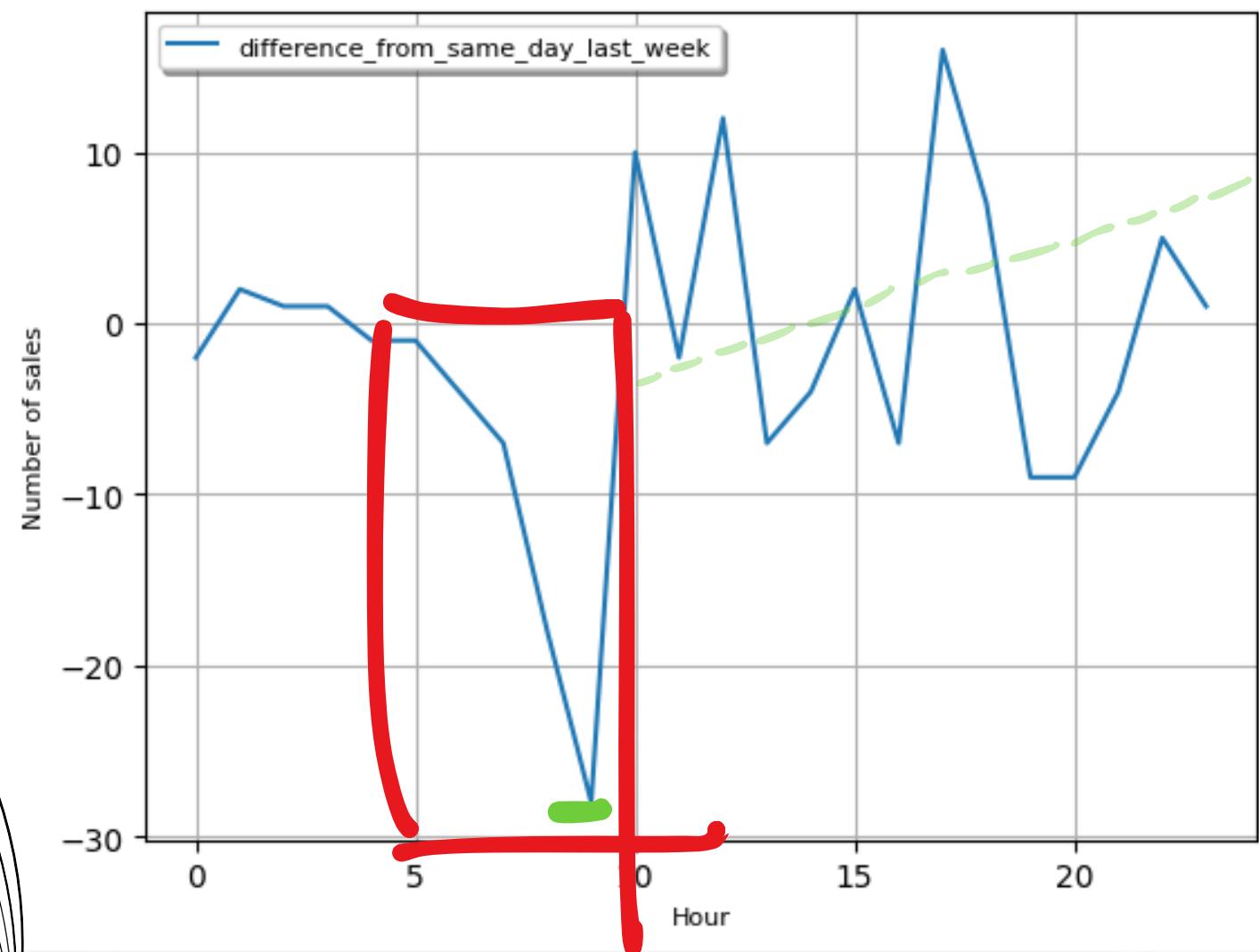
ANÁLISE DESAFIO 1 - CHECKOUT 1

Disponível em: [case_google_colab](#)

Visualização



Difference between today and the same day last week [checkout 1]



O gráfico ao lado ilustra a diferença entre hoje e o mesmo dia semana passada. Se < 0 , indica que o registro de hoje é menor pelo fator indicado no gráfico. Se > 0 , indica que o registro de hoje é maior pelo fator indicado no gráfico.

- Confirmando a suspeita, entre 5h e 9h os registros de hoje são bem menores que os do mesmo dia da semana passada. E de 10h pra frente os registros de hoje acabam ficando ligeiramente maiores na média em relação ao mesmo dia da semana passada. A segunda constatação reforça a hipótese de crescimento de vendas WoW. Já a primeira, demonstra uma anomalia nos registros.
- Chega a quase 30 a diferença da quantidade de vendas da semana passada pra essa semana próximo das 9h

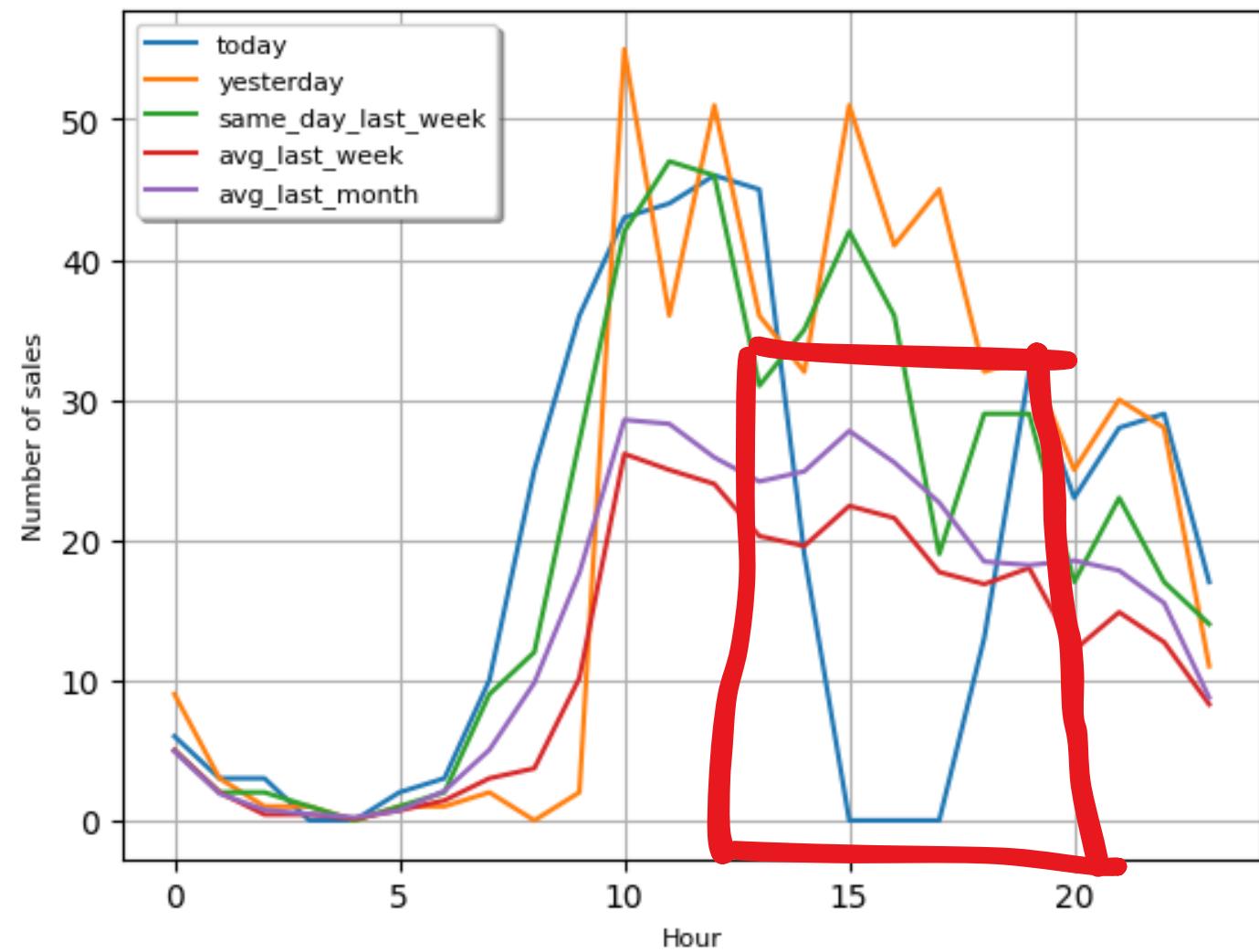
ANÁLISE DESAFIO 1 - CHECKOUT 2

Disponível em: [case_google_colab](#)

Visualização



Comparison between today and the other data points [checkout 2]



O gráfico ao lado ilustra os dados presentes no csv checkout_2. É possível ver que:

- Utilizando o mesmo racional da análise do arquivo anterior, fica mais simples enxergar que a anomalia está entre 14h e 18h.
- Mais especificamente, entre 15h e 17h os registros zeram

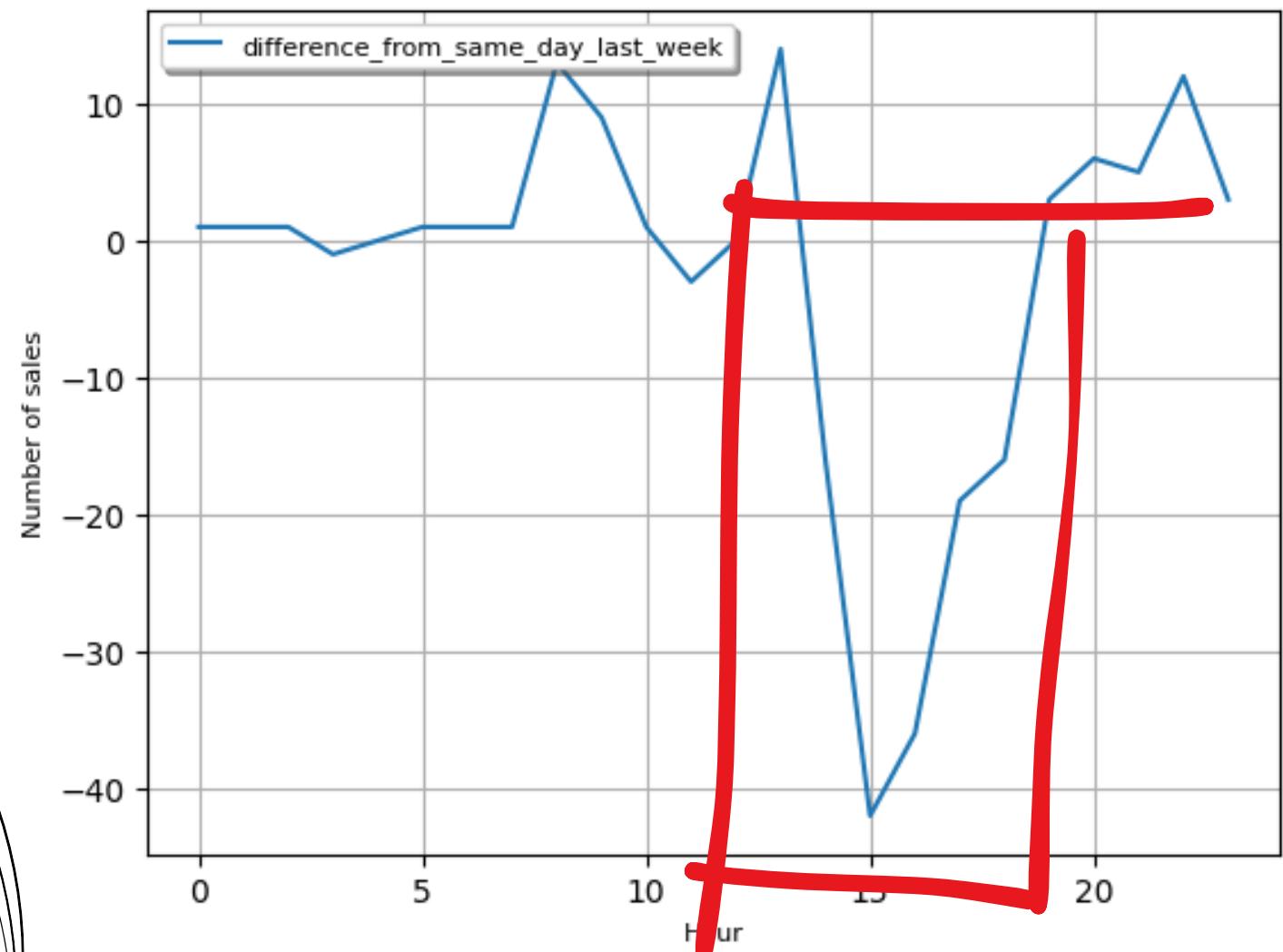
ANÁLISE DESAFIO 1 - CHECKOUT 2

Disponível em: [case_google_colab](#)

Visualização



Difference between today and the same day last week [checkout 2]



Trazendo também o gráfico da diferença entre hoje e o mesmo dia da semana passada, só para reforçar a análise. É possível ver que:

- De fato às 14h e 18h o volume é menor que na semana anterior, mas não tão agressivo quanto entre 15h e 17h.

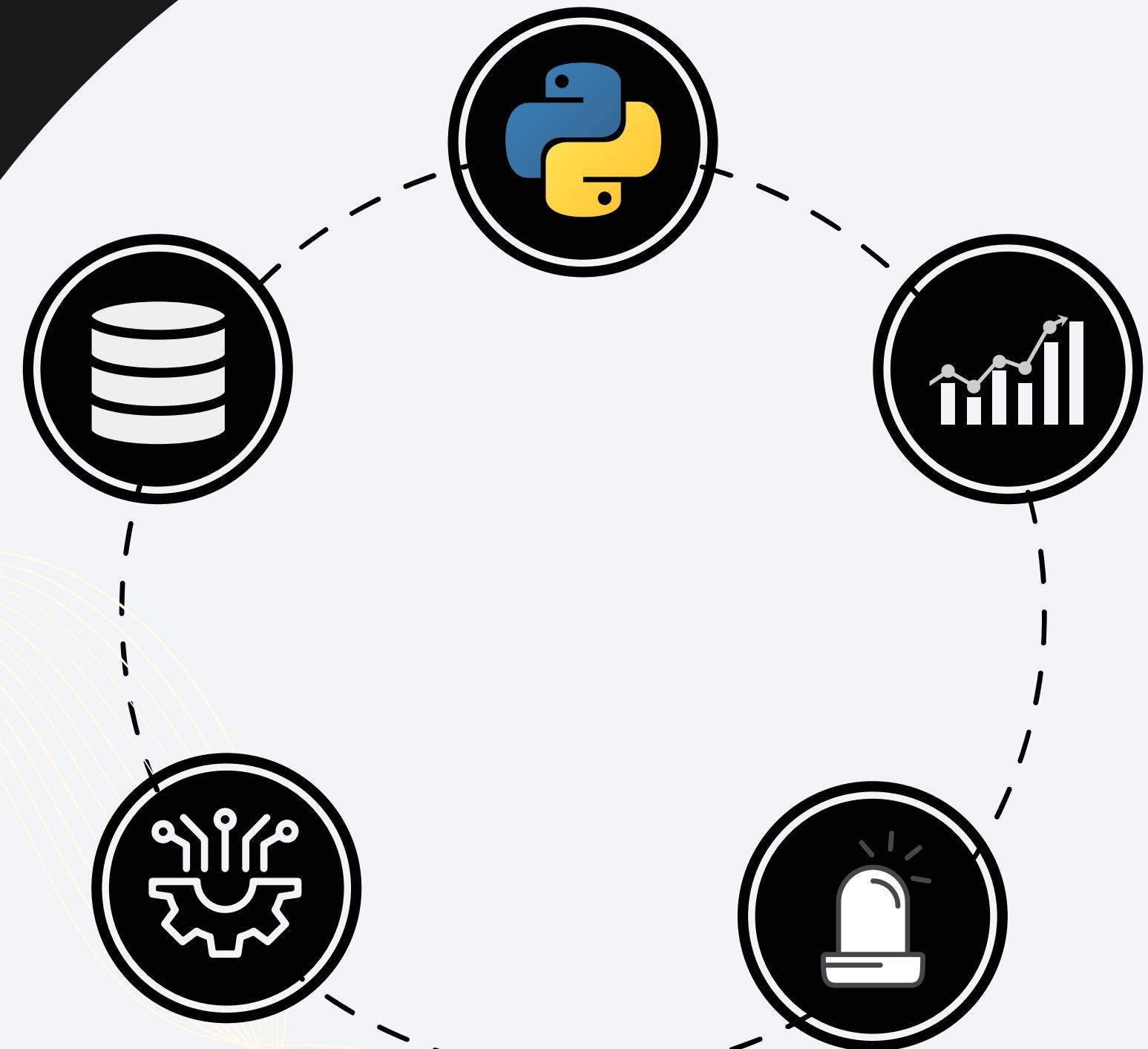
CONCLUSÕES DESAFIO 1

Após cuidadosa análise dos dados tanto em checkout 1 quanto de checkout 2, os seguintes pontos foram concluídos:

- Em checkout 1 foi encontrada uma anomalia 5h e 9h, concluída a partir do baixo registro de vendas nestes horários, seguido de um pico na hora seguinte.
- Já em checkout 2 a anomalia está presente entre 14h e 18h, quando os registros chegaram até mesmo a zerar durante um período desta janela.
- Estas anomalias podem ter sido originadas por diversos fatores, ligados ou não ao produto da CloudWalk (assumindo que os registros se tratam de transações feitas utilizando a InfinitePay).
- Alguns possíveis motivos de responsabilidade do produto/CloudWalk:
 - Falha na conexão com a rede; Falha no leitor de cartão; Falha no processamento do pagamento; Falha de atualização do software; Bloqueios de segurança; Qualquer outra falha de natureza de hardware ou software.
- Alguns possíveis motivos de responsabilidades de terceiros:
 - Falta de energia elétrica no estabelecimento; Bloqueio da conta de pagamento; Estabelecimento fechado; Danos causados ao aparelho.

DESAFIO 2

Sistema de monitoramento





THRESHOLD

O que é acima do normal? Nesta etapa organizamos os dados disponíveis e definimos qual o valor limite para alertar uma transação.

MONITORING

Modelo Random Forest Classifier da lib sklearn foi treinado e testado para, ao receber o horário e quantidade de transações, prever o status da mesma.

ENDPOINT

Endpoint em flask para receber um POST com as informações da transação, e devolver o status (consome funções da etapa de monitoring)

CHAMADA ENDPOINT

Alimenta dados chamando o endpoint, recebe o status das transações, exibe gráficos em tempo real e gerencia alertas.

THRESHOLD

```
#Link do arquivo no git clicando no nome acima
import pandas as pd
import sqlite3
from matplotlib import pyplot as plt

def pd_to_sqlDB(input_df: pd.DataFrame,
                 table_name: str,
                 db_name: str = 'default.db') -> None:
    import logging
    logging.basicConfig(level=logging.INFO,
                        format='%(asctime)s %(levelname)s: %(message)s',
                        datefmt='%Y-%m-%d %H:%M:%S')

    cols = input_df.columns
    cols_string = ', '.join(cols)
    ...
    ...
```

THRESHOLD

- Nesta etapa os dados dos csv do github foram extraídos via `read_csv` (pandas) e tratados em SQL via `sqlite3`.
- Com isso, identificar qual valor utilizar para disparo dos alarmes de anomalias.
- Foi calculada a média de transações por status a cada 10 minutos
- O threshold escolhido para disparar o alarme de anomalia é
 - média + 1*desvio padrão
- Como o desvio padrão médio era pequeno, utilizar a média + 3*desvio padrão parece apropriado (aproximadamente 99,7% dos dados atuais estarão dentro).
- O resultado é uma tabela nomeada *threshold* que conta com as colunas
 - hora: incrementos de 10 minutos para cada um dos status presentes
 - status
 - threshold: valor da média + 1*desvio padrão que será utilizado para setar o alarme.

THRESHOLD

	hora	status	threshold
0	00h 00	approved	28.0
1	00h 00	denied	13.0
2	00h 00	reversed	15.0
3	00h 10	approved	32.0
4	00h 10	denied	9.0
..
445	23h 40	denied	10.0
446	23h 40	reversed	1.0
447	23h 50	approved	64.0
448	23h 50	denied	15.0
449	23h 50	reversed	10.0

MONITORING

- Foi utilizado um RandomForestClassifier, que é um algoritmo ensemble (utiliza vários modelos para gerar o resultado) baseado em decision trees.
- Foi utilizado o csv 1 para treinar e o csv 2 para teste.
- É neste arquivo que a função para prever o status é criada
- Não foi realizado tuning ou ajustes finos, para otimizar o uso do tempo
- A etapa de endpoint aponta para a função predict_status() presente neste arquivo

MONITORING

Acurácia: 0.8135143110061965

Matriz de Confusão:

```
[[1274    31     0     0]
 [ 97    967     0   105]
 [ 36     54     4    73]
 [ 33    112    91   512]]
```

Precisão: 0.7930910416569834

Revocação: 0.8135143110061965

F1-score: 0.8020557220093033

ENDPOINT

- Utilizado framework flask
- Server local com método post para receber o stream de dados para serem classificados e analisados no endpoint /monitorar
- Recebe 2 parâmetros na chamada, o time e o count, e retorna a previsão do status das transações
- Chama a função predict_status() da etapa anterior
- Além disso, possui o endpoint /enviar_email para enviar os alertas por email via smtp do google

ENDPOINT

- * Serving Flask app 'endpoint'
- * Debug mode: on

WARNING: This is a development server.

- * Running on <http://127.0.0.1:5000>

Press CTRL+C to quit

- * Restarting with stat
- * Debugger is active!
- * Debugger PIN: 995-257-903

CHAMADA ENDPOINT

- Envia dados de horário e quantidade de transações para o endpoint
- O endpoint retorna uma previsão do status da transação
- As transações detratoras (denied, reversed e failed) são monitoradas graficamente em tempo real
- Armazena os dados streamados em uma tabela chamada monitoramento
- Em SQL, verifica se o volume das transações detratoras estão acima do normal estipulado em threshold
- Caso verifique anomalia, envia um email informando o status, o threshold e o valor computado

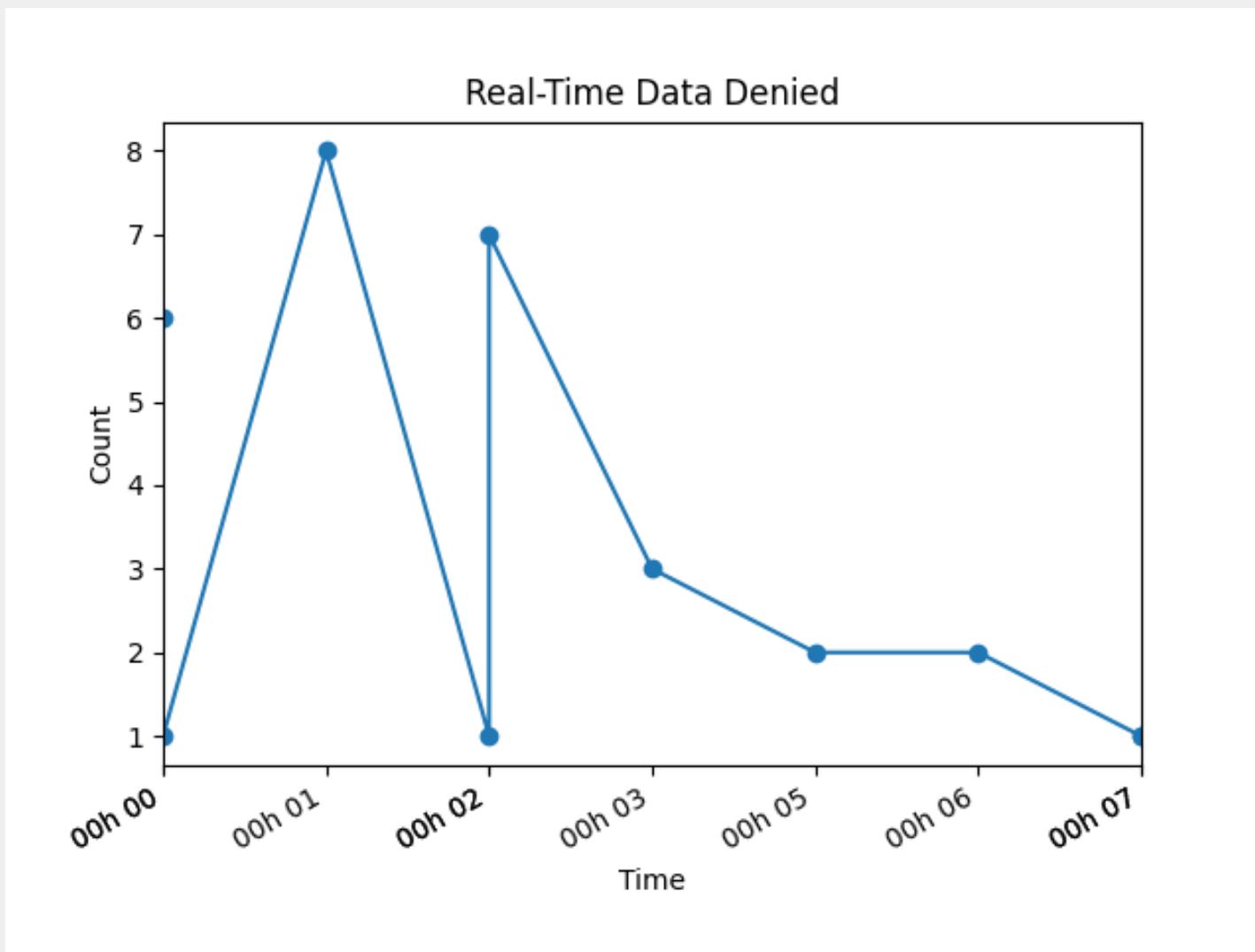
CHAMADA ENDPOINT

- Envia dados de horário e quantidade de transações para o endpoint
- O endpoint retorna uma previsão do status da transação

Resultado da função: approved

CHAMADA ENDPOINT

- As transações detrativas (denied, reversed e failed) são monitoradas graficamente em tempo real



CHAMADA ENDPOINT

- Armazena os dados streamados em uma tabela chamada monitoramento
- Em SQL, verifica se o volume das transações detratoras estão acima do normal estipulado em threshold

hora	status	count	threshold	flag
00h 00	approved	9	5.0	alerta

CHAMADA ENDPOINT

- Caso verifique anomalia, envia um email informando o status, o threshold e o valor computado

santosgabrielsilva83@gmail.com ALERTA DE ANOMALIA!! Status: approved reportou 9 transa... 12:23

X Close | Previous Next

ALERTA DE ANOMALIA!!

s santosgabrielsilva83@gmail.com
To: You

Start reply with: [Ok.](#) [Ok, fico no aguardo.](#) [Ótimo.](#)

Status: approved reportou 9 transações às 00h 00. O máximo considerado normal é 5.0.

[Reply](#) [Forward](#)



OBRIGADO



cloudwalk