

# main

September 19, 2023

## 1 Importação das bibliotecas utilizadas na Analise

```
[1]: import pandas as pd
import numpy as np
import warnings

# Funções auxiliares
from utils.A_utils import dist_cumulativa, plot_boxplot
from utils.A_utils import teste_shapiro, teste_levene, teste_mannwhitneyu
from utils.A_utils import avaliacao_teste_shapiro, agrupar, \
    ↪remocao_todos_outliers

warnings.filterwarnings('ignore')
```

## 2 Conjunto de dados fornecidos pela empresa ACME

```
[2]: (
    axadefeito, acme,
    participacao,
    desenvolvedores
) = pd.read_excel('../ACME_dataset_1.xlsx', sheet_name=None).values()
```

## 3 Pré-processamento dos dados

### 3.1 Agrupando os dados de projetos que utilizaram a técnica ad-hoc e a técnica axadefeito no mesmo dataframe

```
[3]: projetos = pd.concat([acme, axadefeito]).sort_values(by='Projeto').
    ↪reset_index(drop=True).drop(columns='SUCESSO')
projetos.head()
```

```
[3]:
```

	Projeto	Porte	Técnica	Tempo Inspeção (horas)	Defeitos	\
0	1	Pequeno	ACME	0.0	0	
1	2	Pequeno	ACME	0.0	0	
2	3	Pequeno	AXADEFEITO	0.0	0	
3	4	Pequeno	AXADEFEITO	0.0	0	

4	5	Médio	ACME	4.0	0
	Esforço (h/h)	Tamanho (Kloc)	Produtividade (kloc/mês)		
0	5.25	7	5.333333		
1	7.50	10	5.333333		
2	7.50	10	5.333333		
3	8.00	7	3.500000		
4	39.25	51	5.197452		

### 3.2 Extraindo apenas projetos que realizaram Inspeção

```
[4]: projetos_utilizaveis = projetos.loc[projetos['Tempo Inspeção (horas)'] > 0].
     ↪copy()
```

Como não tem como associar a participação dos desenvolvedores nos projetos que estão usando a técnica axadefeito, então não seria possível comparar grupos de devs similares usando técnicas distintas

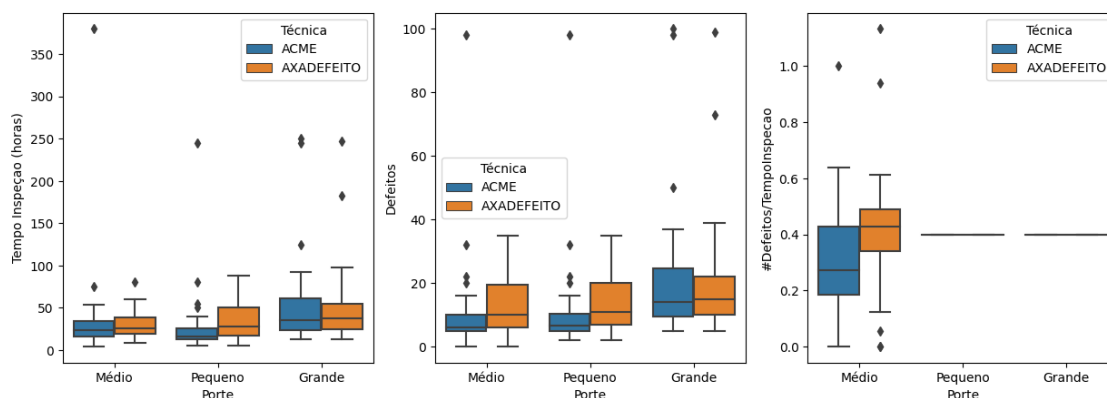
### 3.3 Criação da coluna '#Defeitos/TempoInspecao' e separação das colunas de interesse

```
[5]: projetos_utilizaveis.loc[:, '#Defeitos/TempoInspecao'] =
     ↪projetos_utilizaveis['Defeitos'].div(
        projetos_utilizaveis['Tempo Inspeção (horas)'])

colunas_de_interesse = [
    'Tempo Inspeção (horas)', 'Defeitos',
    '#Defeitos/TempoInspecao'
]
```

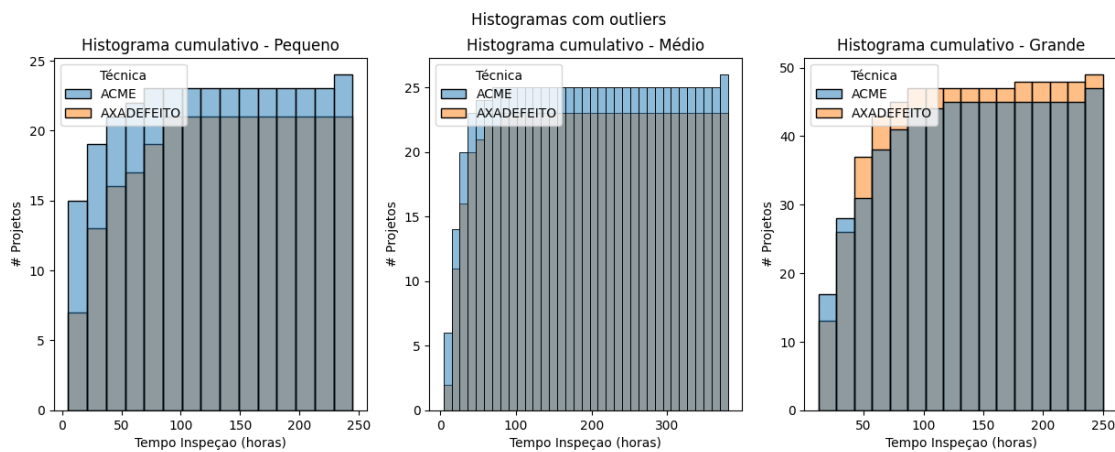
## 4 Análise descritiva em função do porte e das técnicas

```
[6]: plot_boxplot(projetos_utilizaveis, colunas_de_interesse)
```



Na coluna ‘#Defeitos/TempoInspecao’ temos que projetos de Pequeno e Grande porte obtiveram desempenho iguais ao alternar apenas a técnica de inspeção. Mas, em projetos de médio há indícios de que a técnica Axadefeito tenha desempenhado melhor que a inspeção ad-hoc da empresa ACME

```
[7]: dist_cumulativa(projetos_utilizaveis)
```



Através do histograma cumulativo temos que apesar de ambas as técnicas possuírem um desempenho similar, temos que o tempo de inspeção é menor em projetos de pequeno e médio porte ao utilizar a técnica Axadefeito

## 5 Testes de Hipóteses

### 5.1 Identificação do tamanho das amostras

```
[8]: porte_tecnica_group = projetos_utilizaveis.groupby(by=['Porte', 'Técnica'])
```

```
[9]: porte_tecnica_group['Projeto'].count().to_frame()
```

```
[9]:
```

Porte	Técnica	Projeto
Grande	ACME	47
	AXADEFEITO	49
Médio	ACME	26
	AXADEFEITO	23
Pequeno	ACME	24
	AXADEFEITO	21

## 5.2 Teste de Shapiro-Wilk para identificar se as amostras possuem distribuição normal

```
[10]: idx_list, dict_tests = teste_shapiro(porte_tecnica_group,colunas_de_interesse)
```

### 5.2.1 Utilização de $\alpha = 0.05$ para determinar se será possível rejeitar a hipótese nula do teste de shapiro-wilk

```
[11]: results = avaliacao_teste_shapiro(idx_list,dict_tests,alfa=0.05)
results
```

```
[11]:
```

		Tempo Inspeção (horas)-Shapiro \	
Grande	ACME	(0.6524451971054077, 2.7145226066949135e-09)	
	AXADEFEITO	(0.660088300704956, 2.1773671754488078e-09)	
Médio	ACME	(0.38274580240249634, 2.0139641065952674e-09)	
	AXADEFEITO	(0.8943251967430115, 0.01934642158448696)	
Pequeno	ACME	(0.48917537927627563, 4.433870515185845e-08)	
	AXADEFEITO	(0.8885175585746765, 0.02108931913971901)	

		Defeitos-Shapiro \	
Grande	ACME	(0.6524451971054077, 2.7145226066949135e-09)	
	AXADEFEITO	(0.660088300704956, 2.1773671754488078e-09)	
Médio	ACME	(0.49988752603530884, 2.4935355114052982e-08)	
	AXADEFEITO	(0.9019625186920166, 0.02775290608406067)	
Pequeno	ACME	(0.48917537927627563, 4.433870515185845e-08)	
	AXADEFEITO	(0.8885175585746765, 0.02108931913971901)	

		#Defeitos/TempoInspecao-Shapiro \	
Grande	ACME	(1.0, 1.0)	
	AXADEFEITO	(1.0, 1.0)	
Médio	ACME	(0.9260732531547546, 0.06251519173383713)	
	AXADEFEITO	(0.892756998538971, 0.017979398369789124)	
Pequeno	ACME	(1.0, 1.0)	
	AXADEFEITO	(1.0, 1.0)	

		Tempo Inspeção (horas)-ShapiroResultado \	
Grande	ACME	Normal	
	AXADEFEITO	Normal	
Médio	ACME	Normal	
	AXADEFEITO	Normal	
Pequeno	ACME	Normal	
	AXADEFEITO	Normal	

		Defeitos-ShapiroResultado \	
Grande	ACME	Normal	
	AXADEFEITO	Normal	
Médio	ACME	Normal	

	AXADEFEITO	Normal
Pequeno	ACME	Normal
	AXADEFEITO	Normal
	#Defeitos/TempoInspecao-ShapiroResultado	
Grande	ACME	Não-normal
	AXADEFEITO	Não-normal
Médio	ACME	Não-normal
	AXADEFEITO	Normal
Pequeno	ACME	Não-normal
	AXADEFEITO	Não-normal

```
[12]: results[['#Defeitos/TempoInspecao-ShapiroResultado',
'Defeitos-ShapiroResultado',
'Tempo Inspeção (horas)-ShapiroResultado']]
```

```
[12]: #Defeitos/TempoInspecao-ShapiroResultado \
Grande ACME Não-normal
      AXADEFEITO Não-normal
Médio ACME Não-normal
      AXADEFEITO Normal
Pequeno ACME Não-normal
       AXADEFEITO Não-normal

      Defeitos-ShapiroResultado \
Grande ACME Normal
      AXADEFEITO Normal
Médio ACME Normal
      AXADEFEITO Normal
Pequeno ACME Normal
       AXADEFEITO Normal

      Tempo Inspeção (horas)-ShapiroResultado
Grande ACME Normal
      AXADEFEITO Normal
Médio ACME Normal
      AXADEFEITO Normal
Pequeno ACME Normal
       AXADEFEITO Normal
```

Temos então as colunas ‘Defeitos’ e ‘Tempo Inspeção (horas)’ em todos os agrupamentos feitos possuem distribuição normal. E na coluna ‘#Defeitos/TempoInsepeção’ apenas a amostra que apresenta dados de projetos de porte Médio com a técnica Axadefeito possui distribuição normal

### 5.2.2 Utilização do teste de levene para identificar a homocedasticidade dos dados de tempo de Inspeção

Foram realizado 2 testes, um teste para identificar homocedasticidade entre amostras que possuem o mesmo porte e outro teste para identificar homocedasticidade entre amostras que possuem mesma técnica

```
[13]: # Não consigo rejeitar a hipótese nula de que a variancia populacionais são
      ↪ iguais das amostras
      # de uma mesma técnica e portes distintos e de um mesmo porte e técnicas
      ↪ distintas
      teste_levene(projetos_utilizaveis)
```

```
[13]: {'ACME': LeveneResult(statistic=0.1598406430857783, pvalue=0.8525107503294878),
      'AXADEFEITO': LeveneResult(statistic=1.1698168439269472,
      pvalue=0.31509846911496975),
      'Grande': LeveneResult(statistic=0.42505093706928343,
      pvalue=0.5160184105552446),
      'Médio': LeveneResult(statistic=0.7759874568209109,
      pvalue=0.38285370280367303),
      'Pequeno': LeveneResult(statistic=0.003000457046364036,
      pvalue=0.9565702982310833)}
```

Em nenhum dos agrupamentos foi possível rejeitar a hipótese nula do teste de levene, então para realizar os testes de hipótese desse estudo precisaremos recorrer a métodos não-paramétricos

### 5.2.3 Teste de Mann-Whitney U

```
[14]: teste_mannwhitneyu(porte_tecnica_group,colunas_de_interesse)
```

```
[14]:
```

	Tempo Inspeção (horas)	Defeitos \
Grande	(1131.5, 0.8862362086249319)	(1131.5, 0.8862362086249319)
Médio	(268.0, 0.5408252698113492)	(220.5, 0.11714995579089353)
Pequeno	(170.5, 0.06450566991909974)	(170.5, 0.06450566991909974)

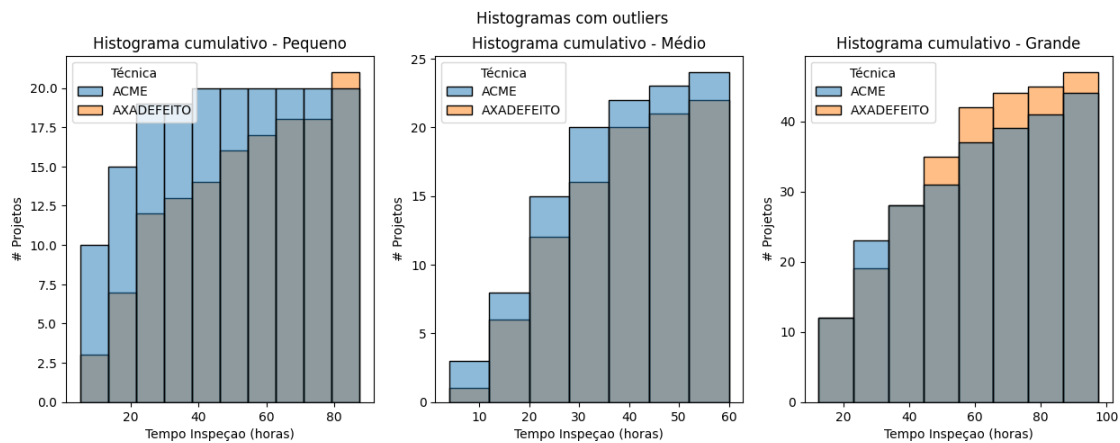
	#Defeitos/TempoInspecao
Grande	(1151.5, 1.0)
Médio	(216.5, 0.10021982428903646)
Pequeno	(252.0, 1.0)

Ao utilizar o teste de Mann-Whitney para testar amostras de mesmo porte e técnicas de diferente, nas colunas de interesse ('Tempo Inspeção (horas)', 'Defeitos' e '#Defeitos/TempoInspecao') para o valor de  $\alpha = 0.05$ , não seria possível rejeitar a hipótese nula do teste de Mann-Whitney, assim não seria possível afirmar que uma técnica é mais eficiente que outra. Contudo, para  $\alpha = 0.10$ , seria possível concluir que o a média de tempo de inspeção e a média da quantidade de defeitos encontrados por projeto de pequeno é distinta, contudo não como avaliar diferença de eficiência entre as técnicas em projetos de pequeno porte

## 6 Remoção dos outliers

```
[23]: df_so_dti = pd.concat([remocao_todos_outliers(j, '#Defeitos/TempoInspecao') for
    ↪ i,j in porte_tecnica_group])
df_so_ti = pd.concat([remocao_todos_outliers(j, 'Tempo Inspeção (horas)') for
    ↪ i,j in porte_tecnica_group])
```

```
[24]: dist_cumulativa(df_so_ti)
```



```
[17]: df_so_dti.groupby(by=['Porte', 'Técnica'])['Projeto'].count().to_frame()
```

```
[17]:
```

		Projeto
Porte	Técnica	
Grande	ACME	47
	AXADEFEITO	49
Médio	ACME	25
	AXADEFEITO	18
Pequeno	ACME	24
	AXADEFEITO	21

```
[25]: df_so_ti.groupby(by=['Porte', 'Técnica'])['Projeto'].count().to_frame()
```

```
[25]:
```

		Projeto
Porte	Técnica	
Grande	ACME	44
	AXADEFEITO	47
Médio	ACME	24
	AXADEFEITO	22
Pequeno	ACME	20
	AXADEFEITO	21

```
[26]: porte_tecnica_group_so_dti = df_so_dti.groupby(by=['Porte', 'Técnica'])
porte_tecnica_group_so_ti = df_so_ti.groupby(by=['Porte', 'Técnica'])
```

```
[27]: idx_list_dti, dict_tests_dti = \
    ↳ teste_shapiro(porte_tecnica_group_so_dti, colunas_de_interesse)
idx_list_ti, dict_tests_ti = \
    ↳ teste_shapiro(porte_tecnica_group_so_ti, colunas_de_interesse)
```

```
[20]: avaliacao_teste_shapiro(idx_list_dti, dict_tests_dti, alfa=0.05)
```

```
[20]:                                     Tempo Inspeção (horas)-Shapiro \
Grande ACME (0.6524451971054077, 2.7145226066949135e-09)
        AXADEFEITO (0.660088300704956, 2.1773671754488078e-09)
Médio ACME (0.3897770047187805, 3.5934659692316018e-09)
        AXADEFEITO (0.8613283038139343, 0.012844356708228588)
Pequeno ACME (0.48917537927627563, 4.433870515185845e-08)
        AXADEFEITO (0.8885175585746765, 0.02108931913971901)
```

```
                                     Defeitos-Shapiro \
Grande ACME (0.6524451971054077, 2.7145226066949135e-09)
        AXADEFEITO (0.660088300704956, 2.1773671754488078e-09)
Médio ACME (0.47317975759506226, 2.0549297374827802e-08)
        AXADEFEITO (0.8904672861099243, 0.039311230182647705)
Pequeno ACME (0.48917537927627563, 4.433870515185845e-08)
        AXADEFEITO (0.8885175585746765, 0.02108931913971901)
```

```
                                     #Defeitos/TempoInspecao-Shapiro \
Grande ACME (1.0, 1.0)
        AXADEFEITO (1.0, 1.0)
Médio ACME (0.9806638360023499, 0.8979429006576538)
        AXADEFEITO (0.9235175848007202, 0.1492086797952652)
Pequeno ACME (1.0, 1.0)
        AXADEFEITO (1.0, 1.0)
```

```
                                     Tempo Inspeção (horas)-ShapiroResultado \
Grande ACME Normal
        AXADEFEITO Normal
Médio ACME Normal
        AXADEFEITO Normal
Pequeno ACME Normal
        AXADEFEITO Normal
```

```
                                     Defeitos-ShapiroResultado \
Grande ACME Normal
        AXADEFEITO Normal
Médio ACME Normal
        AXADEFEITO Normal
```



Pequeno	ACME	Normal
	AXADEFEITO	Normal

		#Defeitos/TempoInspecao-ShapiroResultado
Grande	ACME	Não-normal
	AXADEFEITO	Não-normal
Médio	ACME	Não-normal
	AXADEFEITO	Não-normal
Pequeno	ACME	Não-normal
	AXADEFEITO	Não-normal

```
[28]: avaliacao_teste_shapiro(idx_list_ti,dict_tests_ti,alfa=0.05)
```

```
[28]:                                     Tempo Inspeção (horas)-Shapiro \
```

Grande	ACME	(0.8856297731399536, 0.00040518378955312073)
	AXADEFEITO	(0.9379268288612366, 0.014882960356771946)
Médio	ACME	(0.9717548489570618, 0.7103837132453918)
	AXADEFEITO	(0.9514228701591492, 0.3369480073451996)
Pequeno	ACME	(0.9132116436958313, 0.07340069115161896)
	AXADEFEITO	(0.8885175585746765, 0.02108931913971901)

		Defeitos-Shapiro \
Grande	ACME	(0.8856297731399536, 0.00040518378955312073)
	AXADEFEITO	(0.9379268288612366, 0.014882960356771946)
Médio	ACME	(0.8784918785095215, 0.007750254590064287)
	AXADEFEITO	(0.9113417267799377, 0.050425175577402115)
Pequeno	ACME	(0.9132116436958313, 0.07340069115161896)
	AXADEFEITO	(0.8885175585746765, 0.02108931913971901)

		#Defeitos/TempoInspecao-Shapiro \
Grande	ACME	(1.0, 1.0)
	AXADEFEITO	(1.0, 1.0)
Médio	ACME	(0.9257597923278809, 0.07839958369731903)
	AXADEFEITO	(0.9001866579055786, 0.029982300475239754)
Pequeno	ACME	(1.0, 1.0)
	AXADEFEITO	(1.0, 1.0)

		Tempo Inspeção (horas)-ShapiroResultado \
Grande	ACME	Normal
	AXADEFEITO	Normal
Médio	ACME	Não-normal
	AXADEFEITO	Não-normal
Pequeno	ACME	Não-normal
	AXADEFEITO	Normal

		Defeitos-ShapiroResultado \
Grande	ACME	Normal

	AXADEFEITO	Normal
Médio	ACME	Normal
	AXADEFEITO	Não-normal
Pequeno	ACME	Não-normal
	AXADEFEITO	Normal
		#Defeitos/TempoInspecao-ShapiroResultado
Grande	ACME	Não-normal
	AXADEFEITO	Não-normal
Médio	ACME	Não-normal
	AXADEFEITO	Normal
Pequeno	ACME	Não-normal
	AXADEFEITO	Não-normal

```
[21]: teste_levene(df_so_dti)
```

```
[21]: {'ACME': LeveneResult(statistic=0.15707037856770403, pvalue=0.854870149286194),
      'AXADEFEITO': LeveneResult(statistic=0.7757275327469797,
      pvalue=0.4636000472187636),
      'Grande': LeveneResult(statistic=0.42505093706928343,
      pvalue=0.5160184105552446),
      'Médio': LeveneResult(statistic=0.5845306750091859, pvalue=0.4489200921050053),
      'Pequeno': LeveneResult(statistic=0.003000457046364036,
      pvalue=0.9565702982310833)}
```

```
[29]: teste_levene(df_so_ti)
```

```
[29]: {'ACME': LeveneResult(statistic=7.231611499696145, pvalue=0.001257902279199466),
      'AXADEFEITO': LeveneResult(statistic=3.010319974741589,
      pvalue=0.054436091986716825),
      'Grande': LeveneResult(statistic=0.39058826550763565,
      pvalue=0.5335894331631206),
      'Médio': LeveneResult(statistic=0.01509376322597679,
      pvalue=0.9027800554873622),
      'Pequeno': LeveneResult(statistic=9.97373606517994,
      pvalue=0.0030618950185192914)}
```

Olhando os dados de Tempo de Inspeção, temos que as amostras de projetos de pequeno porte tem homocedasticidade, contudo projetos de pequeno porte que usam a técnica ACME não apresentam uma distribuição normal. E apesar dos projetos que usam a Técnica da ACME apresentam homocedasticidade, projetos de pequeno e é médio porte que usam a técnica da ACME não apresentam distribuição normal. Com isso, usaremos apenas testes de hipótese não paramétricos

```
[22]: teste_mannwhitneyu(porte_tecnica_group_so_dti,colunas_de_interesse)
```

```
[22]:
```

	Tempo Inspecao (horas)	Defeitos \
Grande	(1131.5, 0.8862362086249319)	(1131.5, 0.8862362086249319)

Médio	(197.0, 0.4979980256337464)	(130.0, 0.019562203780744775)
Pequeno	(170.5, 0.06450566991909974)	(170.5, 0.06450566991909974)

	#Defeitos/TempoInspecao
Grande	(1151.5, 1.0)
Médio	(123.5, 0.012848128493903628)
Pequeno	(252.0, 1.0)

Ao remover outliers presentes na coluna ‘#Defeitos/TempoInspecao’, temos que em projetos de Médio porte há evidência que é possível rejeitar a hipótese nula de que a média da eficiência de ambas as técnicas de inspeção são iguais

```
[30]: teste_mannwhitneyu(porte_tecnica_group_so_ti,colunas_de_interesse)
```

```
[30]:
```

	Tempo Inspecao (horas)	Defeitos \
Grande	(987.5, 0.714536105014925)	(987.5, 0.714536105014925)
Médio	(223.0, 0.37263058123841597)	(177.0, 0.05634967397818717)
Pequeno	(99.5, 0.003966149721504965)	(99.5, 0.003966149721504965)

	#Defeitos/TempoInspecao
Grande	(1034.0, 1.0)
Médio	(194.5, 0.1289290535989367)
Pequeno	(210.0, 1.0)

Ao remover outlier presentes na coluna ‘Tempo Inspecao (horas)’, temos que em projetos de pequeno porte há evidência que é possível rejeitar a hipótese nula de que a média de tempo gasto com inspeção de ambas as técnicas de inspeção são iguais