Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

Departamento de Estatística

Relatório Trabalho 1 - Grupo 2
Análise de Regressão

Grupo: Andrielle Couto - 770295
Crystiane Souza - 760955
Douglas Nestlehner - 752728
Eric Sato - 729739

Outubro, 2021

Sumário

Capítulo 1

Introdução

Este trabalho tem como objetivo analisar uma possível relação entre o desempenho relativo de uma CPU (Central Processing Unit) e 6 outros fatores, sendo eles: o tempo de ciclo da máquina (em nanossegundos), a memória principal mínima (em kilobytes), a memória principal máxima (em kilobytes), a memória cache (em kilobytes), os canais mínimos (em unidades) e os canais máximos (em unidades).

Para isso, utilizamos a base de dados "Relative CPU Performance" retirada da plataforma UCI (`https://archive.ics.uci.edu/ml/datasets/Computer+Hardware`), a qual possui 209 observações.

Nesse contexto, tomamos:

- Y: desempenho relativo da CPU;
- x_1 : tempo de ciclo da máquina (em nanossegundos);
- x_2 : memória principal mínima (em kilobytes);
- x_3 : memória principal máxima (em kilobytes);
- x_4 : memória cache (em kilobytes);
- x_5 : canais mínimos (em unidades);
- x_6 : canais máximos (em unidades).

Portanto, queremos ajustar o modelo abaixo:

$$Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \beta_4 x_{i4} + \beta_5 x_{i5} + \beta_6 x_{i6} + \epsilon_i, \quad i = 1, .., 209$$

Nesse sentido, no Capítulo 2 apresentamos os resultados obtidos, que compreende a análise descritiva dos dados, o ajuste do modelo, a análise de resíduos, as transformações necessárias para quando o modelo ajustado não é adequado (as suposições iniciais de linearidade, normalidade e homocedasticidade não se confirmaram), a construção da Tabela ANOVA e a apresentação do modelo ajustado final. No Capítulo 3 estão as conclusões e no Apêndice A pode ser encontrado o código feito em linguagem Python e utilizado para a realização desse trabalho.

Capítulo 2

Resultados

Neste capítulo, apresentamos os resultados obtidos.

2.1 Análise Descritiva dos Dados

Como citado anteriormente, a base de dados utilizada possui 209 observações e 6 covariáveis, as quais foram descritas na Introdução. Na Tabela 2.1 abaixo estão apresentadas algumas observações referentes à essas covariáveis.

	x_1	x_2	x_3	x_4	x_5	x_6
0	23	16000	64000	64	16	32
1	23	32000	64000	128	32	64
2	30	8000	64000	96	12	176
3	30	8000	64000	128	12	176
...	...	...	...	...	...	...
205	300	192	768	6	6	24
206	810	512	512	8	1	1
207	480	96	512	0	1	1
208	350	64	64	0	1	4

Tabela 2.1: Base de dados das covariáveis.

Primeiramente, realizamos o cálculo de algumas medidas descritivas de cada covariável, como: mínimo, mediana, média, máximo, entre outros. Os resultados podem ser observamos na Tabela 2.2.

	x_1	x_2	x_3	x_4	x_5	x_6
Mínimo	17.00	64.00	64.00	0.00	0.00	0.00
1º Quartil	50.00	768.00	4000.00	0.00	1.00	5.00
Mediana	110.00	2000.00	8000.00	8.00	2.00	8.00
Média	203.82	2867.98	11796.15	25.21	4.70	18.27
3º Quartil	225.00	4000.00	16000.00	32.00	6.00	24.00
Máximo	1500.00	32000.00	64000.00	256.00	52.00	176.00

Tabela 2.2: Medidas descritivas das covariáveis.

Notamos que ambas as memórias principais (mínima e máxima), representadas pelas covariáveis x_2 e x_3, possuem valor mínimo de 64 kilobytes, mas que o valor máximo da memória principal mínima é de 32000 kilobytes, enquanto que a da memória principal máxima é de 64000 kilobytes. Ademais, percebemos também que a média da memória principal máxima (x_3) é muito maior do que a média das outras covariáveis. Além disso, os valores mínimos de memória cache (x_4), canais mínimos (x_5) e canais máximos (x_6) são iguais a 0.

Para a continuidade da análise descritiva dos dados, plotamos um gráfico Boxplot com todas as covariáveis juntas e um com elas separadas para fins de comparação:

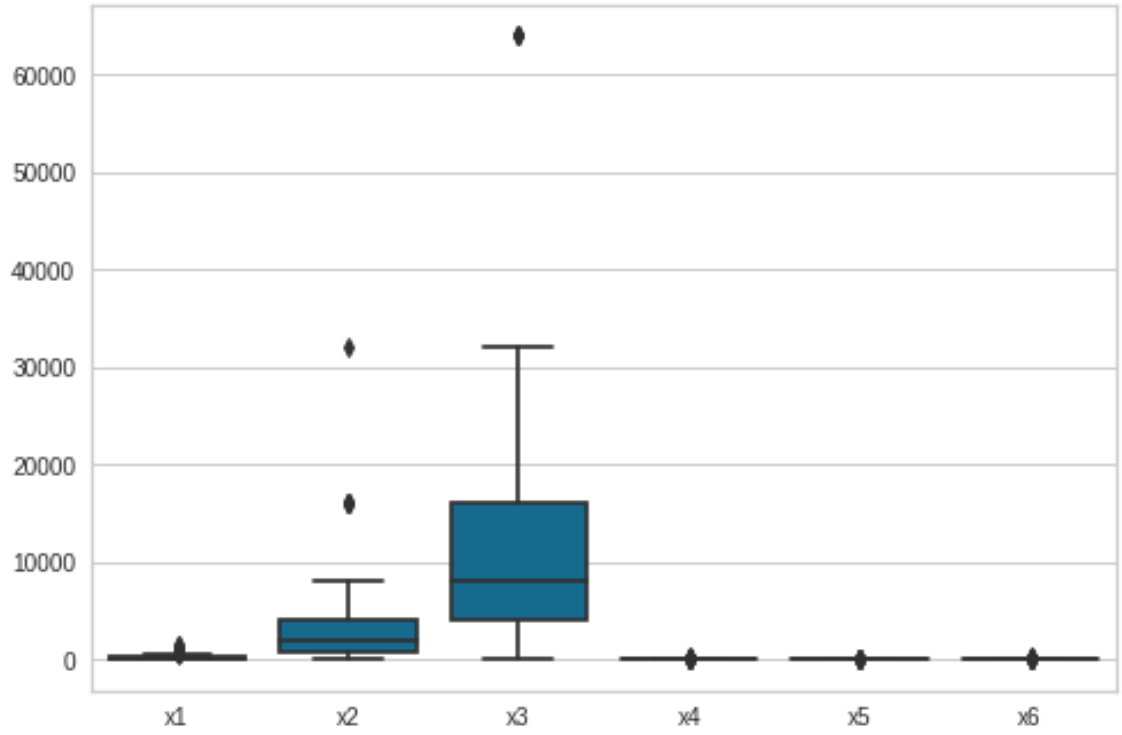

Figura 2.1: Boxplot conjunto das covariáveis.

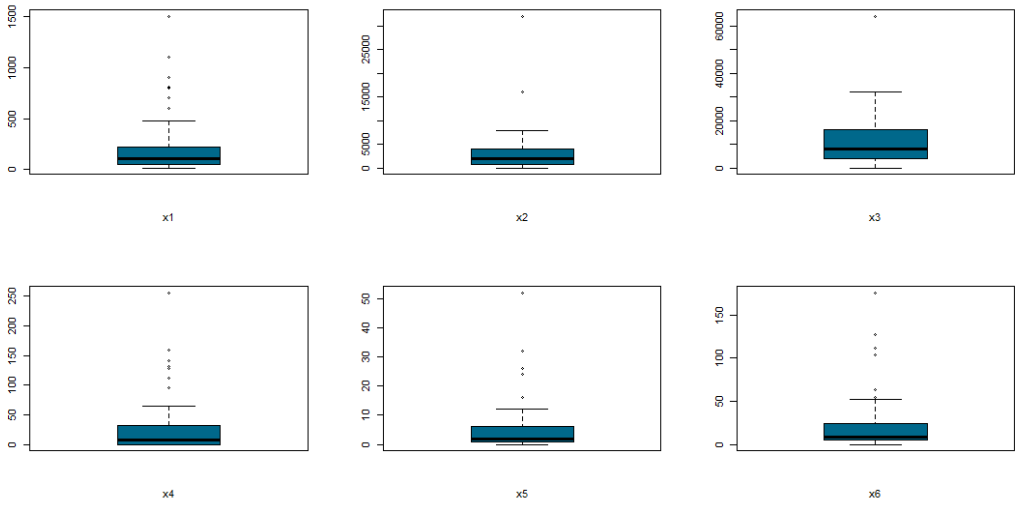

Figura 2.2: Boxplots separados das covariáveis.

Através da Figura 2.1, o fato da covariável x_3 possuir valores muito maiores que as demais fica ainda mais visível, além de percebermos pontos outliers bem aberrantes, em relação ao todo, presentes nas covariáveis x_2 e x_3. Ademais, por meio da Figura 2.2 observamos que todas as covariáveis possuem outliers, assim como um comportamento assimétrico à direita.

Para verificar o grau de dependência entre duas variáveis, plotamos uma matriz de correlação, representada pela Figura 2.3.

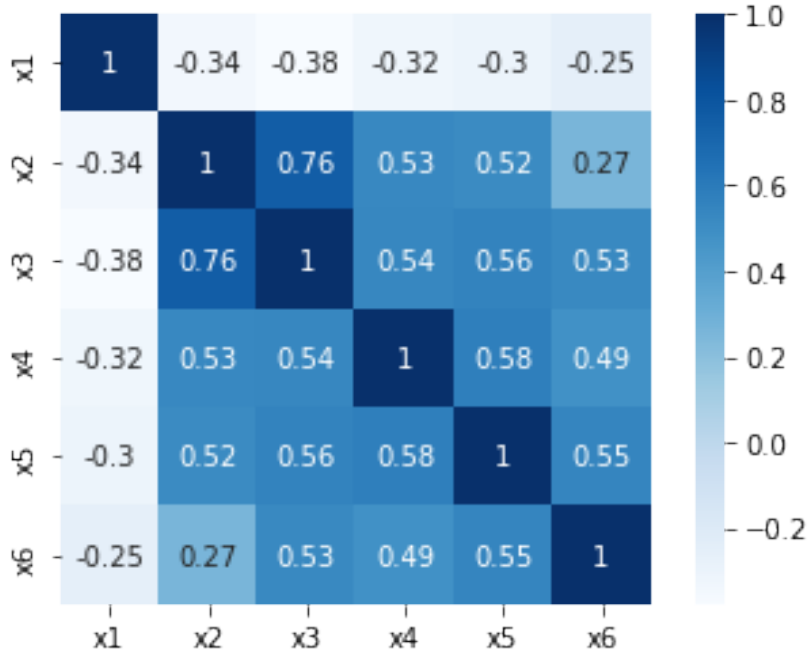

Figura 2.3: Matriz de correlação entre as covariáveis.

Primeiro, ressaltamos que as correlação de cada covariável com ela mesma é igual a 1 e, portanto, a diagonal principal dessa matriz é inteira de uns. Dessa forma, ao analisarmos a

Figura 2.3 notamos que temos uma correlação forte em apenas um par de variáveis, sendo: a variável de memória principal mínima em kilobytes (x_2) que tem 76% de correlação com a variável de memória principal máxima em kilobytes (x_3). Além disso, é perceptível que várias variáveis possuem uma correlação moderada, onde: a variável de memória cache em kilobytes (x_4) que tem 58% de correlação com os canais mínimos em unidades (x_5), a variável de memória principal máxima em kilobytes (x_3) que tem 56% de correlação com os canais mínimos em unidades (x_5), entre outros. Ademais, como nenhuma correlação entre duas covariáveis é 0, podemos concluir que elas não são independentes.

A fim demonstrar se existem correlações lineares entre as covariáveis, fizemos em seguida a matriz de dispersão, a qual está representada abaixo.

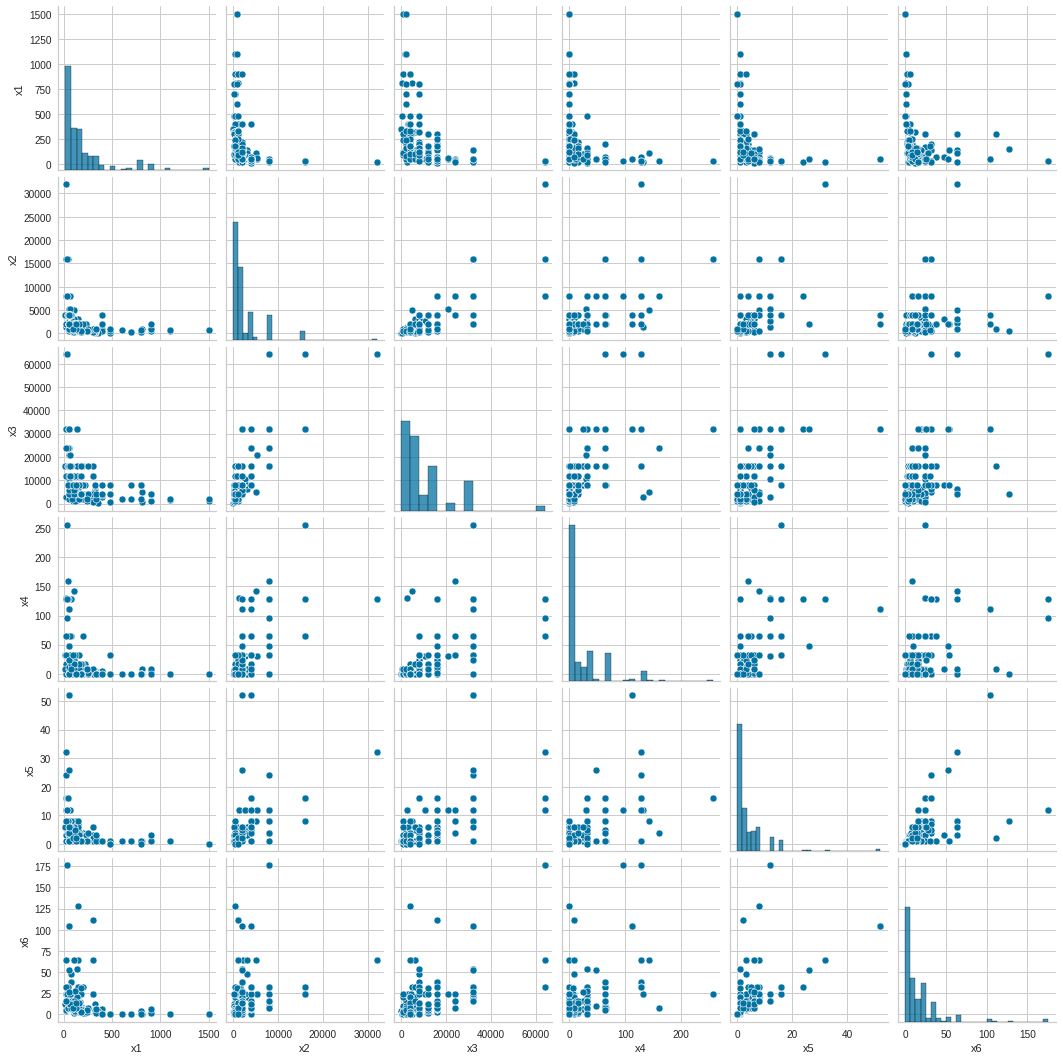

Figura 2.4: Matriz de dispersão.

A diagonal principal da matriz da Figura 2.4 acima contém os histogramas de cada covariável, e os demais gráficos são gráficos de dispersão entre duas diferentes covariáveis. Analisando os gráficos fora da diagonal, concluímos que não parece haver correlações lineares entre as covariáveis, já que nenhum dos gráficos presentes na matriz acima aparentam um comportamento linear.

Após isso, analisamos a variável resposta Y, a qual possui algumas de suas observações apresentadas abaixo:

	Y
0	636
1	1144
2	915
3	1150
...	...
205	36
206	18
207	6
208	10

Tabela 2.3: Base de dados da variável resposta.

Assim como para as covariáveis, algumas de suas medidas descritivas foram calculadas e são apresentadas abaixo:

	Y
Mínimo	6.00
1º Quartil	27.00
Mediana	50.00
Média	105.62
3º Quartil	113.00
Máximo	1150.00

Tabela 2.4: Medidas descritivas da variável resposta.

Percebemos através da Tabela 2.4 que Y parece ter outliers, já que o valor máximo observado é bem superior ao 3º quartil. Para melhor análise, fizemos o histograma da variável resposta.

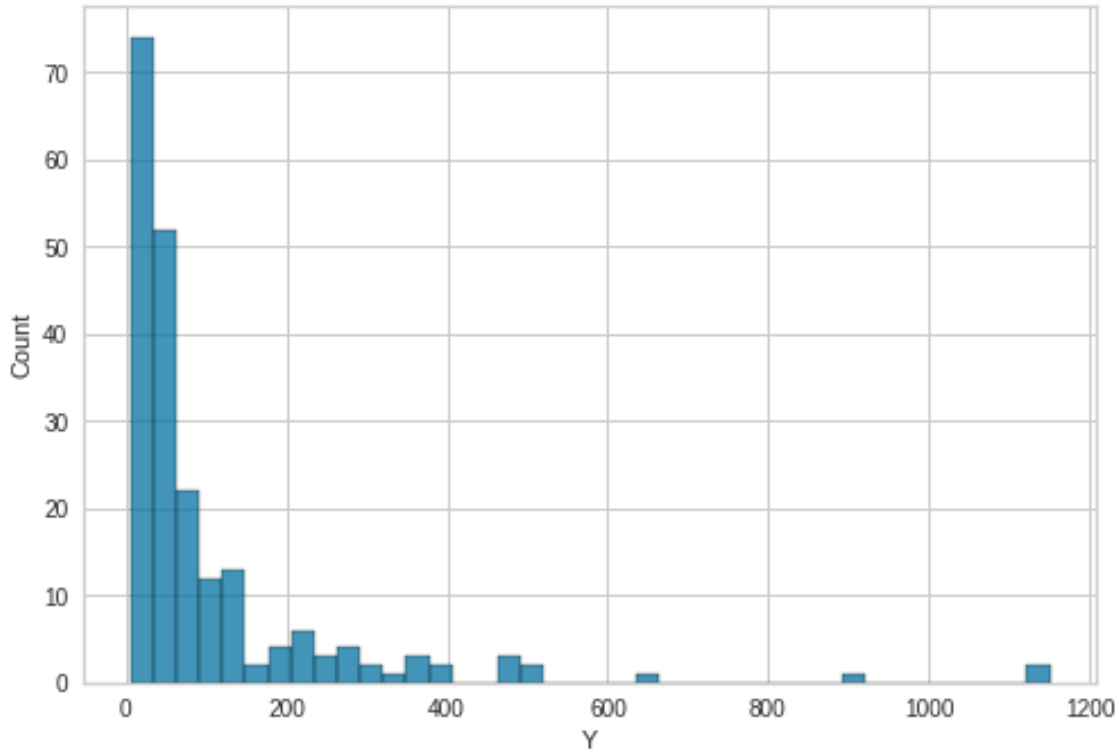

Figura 2.5: Histograma da variável resposta.

A análise do gráfico da Figura 2.5 nos permite perceber uma assimetria à direita, causada provavelmente por algum(ns) outlier(s) com valores muito grandes. Além disso, percebemos que os valores estão bem concentrados entre 0 e 100.

2.2 Ajuste do Modelo

Nessa seção ajustamos um modelo de regressão linear múltipla, com todas as covariáveis x_1, x_2, x_3, x_4, x_5 e x_6 e com todas as 209 observações.

Assim sendo, encontramos os seguintes valores:

$$\hat{\beta}_0 = -55.89393360702436$$
$$\hat{\beta}_1 = 0.04885490012531934$$
$$\hat{\beta}_2 = 0.015292571902433987$$
$$\hat{\beta}_3 = 0.005571389725107793$$
$$\hat{\beta}_4 = 0.6414014269981577$$
$$\hat{\beta}_5 = -0.27035754831755177$$
$$\hat{\beta}_6 = 1.4824721704649522$$

Dessa forma, o modelo ajustado é dado por:

$$\hat{Y}_i = -55.8939 + 0.0488x_{i1} + 0.0153x_{i2} + 0.0056x_{i3} + 0.6414x_{i4} - 0.2703x_{i5} + 1.4825x_{i6},$$

com $i = 1, ..., 209$.

2.3 Análise de Resíduos

Quando ajustamos um modelo de regressão impomos uma séries de condições (linearidade, independência, normalidade, homocedasticidade, entre outros). A fim de verificar se essas condições iniciais foram de fato satisfeitas, fizemos a análise de resíduos, a qual estuda o comportamento dos dados observados através dos resíduos (erros) do modelo ajustado.

Esclarecemos que como não possuímos a ordem de coleta dos dados, não conseguimos verificar a suposição de independência dos erros. Logo, admitimos que ela é satisfeita.

Calculamos inicialmente os valores ajustados de Y_i, os quais são dados abaixo:

```
[ 6.30642902e+02  9.59487132e+02  7.43726579e+02 ... 7.04209886e-01
-2.69108285e+01  -3.17998939e+01]
```

Figura 2.6: Valores de $\hat{Y}_i$, i = 1, ..., 209.

Tendo os valores de $\hat{Y}_i$, calculamos os resíduos ϵ_i através da fórmula abaixo

$$\epsilon_i = Y_i - \hat{Y}_i, \quad i = 1, ..., 209$$

obtendo:

```
[ 5.35709805e+00  1.84512868e+02  1.71273421e+02 ... 1.72957901e+01
 3.29108285e+01  4.17998939e+01]
```

Figura 2.7: Valores dos resíduos ϵ_i, i = 1, ..., 209.

Desse modo, através desses resultados plotamos o gráfico de Resíduos vs Valores Ajustados, o qual é útil para detectar a presença de outliers, heterocedasticidade, independência, não linearidade e não normalidade, e o gráfico Normal Q-Q, útil para detectar se os erros são identicamente distribuídos com distribuição $N(0, \sigma^2)$.

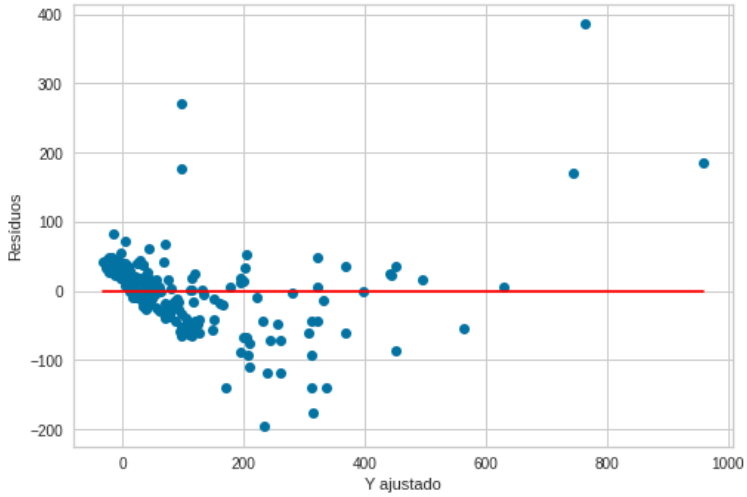

Figura 2.8: Gráfico de Resíduos vs Valores Ajustados.

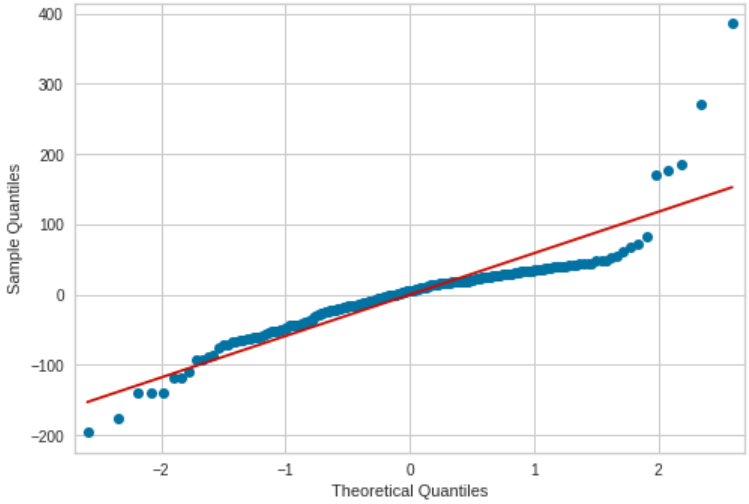

Figura 2.9: Normal Q-Q.

O gráfico da Figura 2.8 indica a presença de outliers, assim como uma não linearidade. Além disso, percebemos uma dispersão dos pontos à medida que aumenta-se os valores ajustados, o que demonstra heterocedasticidade. Ademais, o gráfico da Figura 2.9 nos mostra que não há normalidade, uma vez que os pontos não estão tão próximos da reta.

Para ajudar a confirmar nossas análises gráficas, conduzimos testes para normalidade e homocedasticidade considerando um nível de significância de 5%. Para testar a normalidade, utilizamos os seguintes testes: Shapiro e Anderson-Darling, e obtivemos como resultado:

Teste Shapiro-Wilk

$$\text{valor-p} = 2.5973060846702124e - 14$$

Teste Anderson-Darling

estatística teste = 7.174982887852195

valor crítico = 0.773

Como no teste de Shapiro-Wilk o valor-p = 2.5973060846702124e-14 $< \alpha = 0.05$ e no teste de Anderson-Darling a estatística teste = 7.174982887852195 > valor crítico = 0.773, rejeitamos a hipótese nula em ambos, ou seja, ao nível de significância de 5% concluímos que os resíduos realmente não seguem uma distribuição normal.

Em seguida, para testar a homocedasticidade, utilizamos o teste de Bartlett e obtivemos como resultado:

Teste Bartlett

valor-p = 0.013248243984240488

Assim, fundamentado no teste acima, como valor-p = 0.013248243984240488 $< \alpha =$ 0.05, rejeitamos a hipótese nula, ou seja, ao nível de significância de 5% concluímos que os resíduos são heterocedásticos.

Desse modo, como observamos que algumas das suposições não estão atendidas, ou seja, como constatamos não linearidade, heterocedasticidade e não normalidade, foi preciso fazer transformações simultaneamente tanto na variável resposta quanto nas covariáveis.

2.4 Transformações

Como detectamos na análise de resíduos um problema de não linearidade, heterocedasticidade e não normalidade, foi necessário realizar uma transformação simultânea em Y e nas covariáveis.

Primeiramente, aplicamos uma transformação em Y a fim de obter homocedasticidade e normalidade. Para isso, testamos várias transformações possíveis, entre elas: $\sqrt{Y}, Y^2, \frac{1}{Y}$ e $log(Y)$. A transformação escolhida que melhor se ajustou foi:

$$Y' = log(Y + 0.001)$$

Destacamos que somamos 0.001 para os valores de Y pois através da Tabela 2.4 percebemos que algumas observações de Y assumem valor 0, e o logarítimo natural não está definido em 0.

Desse modo, ajustamos então um novo modelo de regressão considerando $Y' = log(Y + 0.001)$, e obtivemos:

$$\hat{Y'}_i = 3.3966 - 0.0008x_{i1} + 2.7077 \cdot 10^{-5}x_{i2} + 4.2689 \cdot 10^{-5}x_{i3} + 0.0077x_{i4} + 0.0067x_{i5} + 0.0002x_{i6},$$

com $i = 1, ..., 209$.

Fizemos a análise de resíduos para esse novo modelo plotando o gráfico de Resíduos vs Valores Ajustados e o Normal Q-Q:

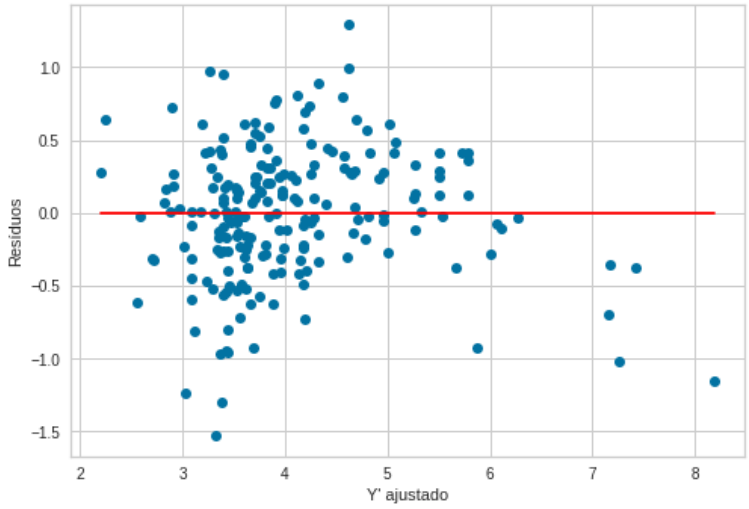

Figura 2.10: Gráfico de Resíduos vs Valores Ajustados para o novo modelo.

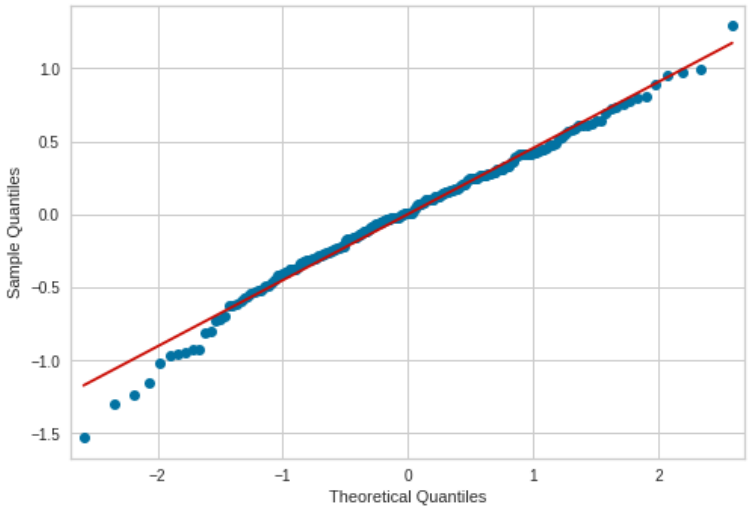

Figura 2.11: Normal Q-Q para o novo modelo.

O gráfico da Figura 2.10 indica a presença de outliers ainda, assim como uma não linearidade. Além disso, percebemos uma menor concentração dos pontos em comparação com a análise do modelo anterior, o que demonstra a possibilidade de termos obtido homocedasticidade. Ademais, o gráfico da Figura 2.11 nos mostra que agora há normalidade, uma vez que os pontos estão bem mais próximos da reta.

Para ajudar a confirmar nossas análises gráficas, conduzimos novamente testes para normalidade e homocedasticidade considerando um nível de significância de 5%. Para testar a normalidade, utilizamos os mesmos testes: Shapiro-Wilk e Anderson-Darling, e obtivemos como resultado:

Teste Shapiro-Wilk

valor-p = 0.11035391688346863

Teste Anderson-Darling

estatística teste = 0.5898591186378326

valor crítico = 0.773

Como no teste de Shapiro-Wilk o valor-p = $0.11035391688346863 > \alpha = 0.05$ e no teste de Anderson-Darling a estatística teste = $0.5898591186378326 <$ valor crítico = 0.773, não rejeitamos a hipótese nula em ambos, ou seja, ao nível de significância de 5% concluímos que os resíduos agora seguem uma distribuição normal.

Em seguida, para testar a homocedasticidade, utilizamos o teste de Bartlett e obtivemos como resultado:

Teste Barlett

valor-p = 0.018160357043744087

Assim, fundamentado no teste acima, como o valor-p = $0.018160357043744087 < \alpha = 0.05$, rejeitamos a hipótese nula, ou seja, ao nível de significância de 5% concluímos que os resíduos continuam heterocedásticos. Porém, como a análise gráfica apontava para a possibilidade de homocedasticidade, conduzimos dessa vez também o teste de Levene para homocedasticidade para fins de comparação.

Teste Levene

valor-p = 0.16537242164007893

Observamos pelo teste acima que o valor-p $= 0.16537242164007893 > \alpha = 0.05$ e, portanto, não rejeitamos a hipótese nula, ou seja, ao nível de significância de 5% podemos concluir que os resíduos são homocedásticos.

Como obtivemos dois testes com diferentes conclusões, nos apoiamos na análise gráfica e consideramos provisoriamente a suposição de homocedasticidade comprovada para fazer transformações nas covariáveis. Após as transformações nas covariáveis, voltamos a fazer a análise de resíduos e verificar essa suposição.

Após isso, aplicamos uma transformações nas covariáveis a fim de obter linearidade. Para tal, primeiro observamos os seguintes diagramas de dispersão:

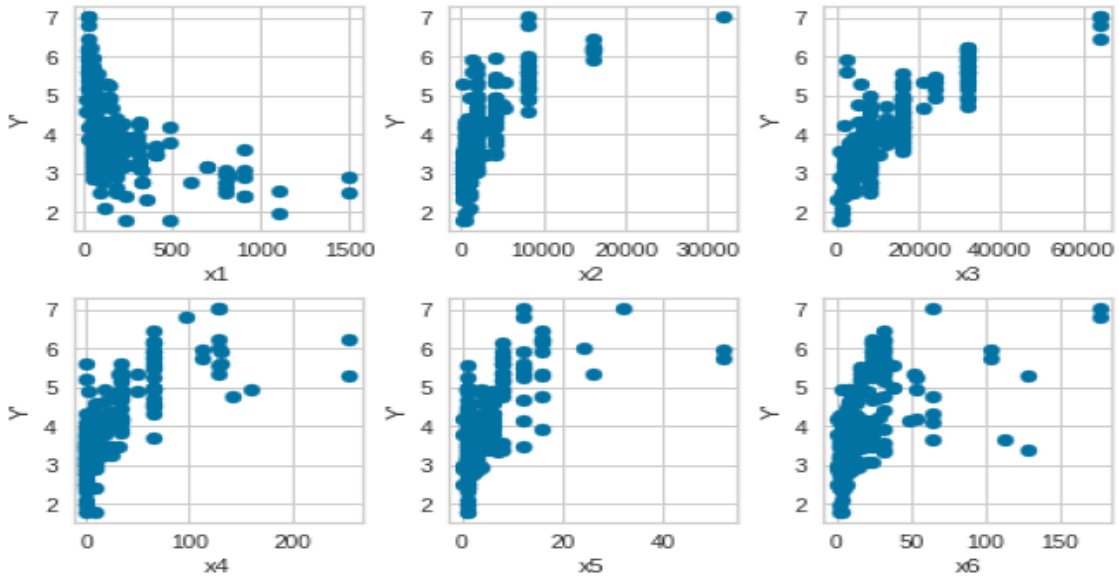

Figura 2.12: Diagramas de dispersão.

Através da análise dos gráficos da Figura 2.12 acima, percebemos que as transformações que seriam adequadas para solucionar o problema de não linearidade detectado na análise de resíduos seriam:

$$x'_1 = \frac{1}{x_1} \quad ; \quad x'_2 = \sqrt{x_2} \quad ; \quad x'_3 = \sqrt{x_3} \quad ; \quad x'_4 = \sqrt{x_4} \quad ; \quad x'_5 = \sqrt{x_5} \quad ; \quad x'_6 = \sqrt{x_6}$$

Desse modo, ajustamos um novo modelo de regressão considerando as transformações acima e obtivemos:

$$\hat{Y'}_i = 2.2433 + 7.8224\frac{1}{x_{i1}} + 0.0065\sqrt{x_{i2}} + 0.0080\sqrt{x_{i3}} + 0.0959\sqrt{x_{i4}} + 0.0192\sqrt{x_{i5}} + 0.0622\sqrt{x_{i6}},$$

com $i = 1, ..., 209$.

Fizemos novamente a análise de resíduos para esse último modelo plotando o gráfico de Resíduos vs Valores Ajustados e o Normal Q-Q:

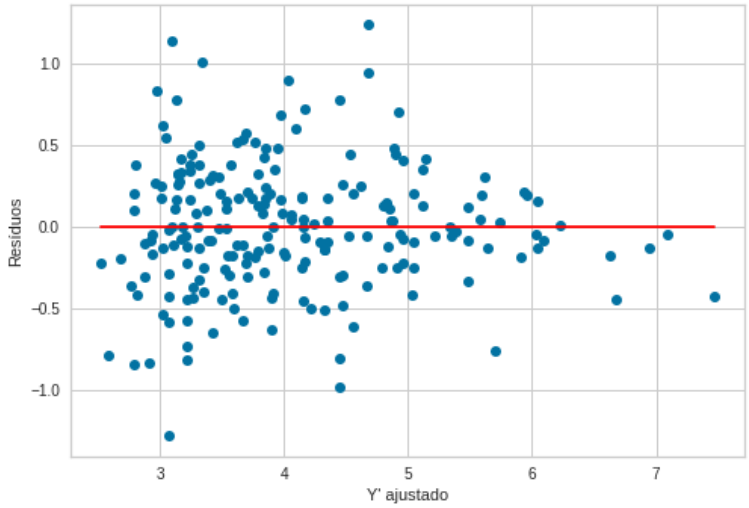

Figura 2.13: Gráfico de Resíduos vs Valores Ajustados para o novo modelo.

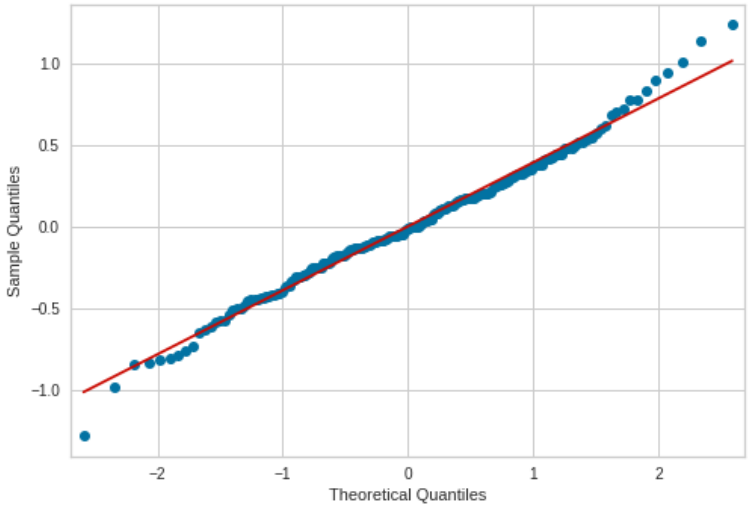

Figura 2.14: Normal Q-Q para o novo modelo.

O gráfico da Figura 2.13 indica a presença de alguns outliers. Porém, agora a linearidade parece ter sido satisfeita. Além disso, o gráfico não possui dispersão grande ou irregular de resíduos entre valores ajustados e, portanto, a homocedasticidade está verificada. Ademais, o gráfico da Figura 2.14 nos mostra que os pontos estão ainda mais próximos da reta, o que comprova a normalidade.

Para ajudar a confirmar nossas análises gráficas, conduzimos novamente testes para normalidade e homocedasticidade considerando um nível de significância de 5%. Para testar a normalidade, utilizamos os mesmos testes: Shapiro-Wilk e Anderson-Darling, e obtivemos como resultado:

Teste Shapiro-Wilk

valor-p = 0.30084821581840515

Teste Anderson-Darling

estatística teste = 0.5256174646598311

valor crítico = 0.773

Como no teste de Shapiro-Wilk o valor-p = 0.30084821581840515 $> \alpha = 0.05$ e no teste de Anderson-Darling a estatística teste = 0.5256174646598311 $<$ valor crítico = 0.773, não rejeitamos a hipótese nula em ambos, ou seja, ao nível de significância de 5% concluímos que os resíduos continuam a seguir uma distribuição normal.

Em seguida, para testar a homocedasticidade, utilizamos os testes de Bartlett e Levene, e obtivemos como resultado:

Teste Bartlett

valor-p = 0.482022473018619

Teste Levene

valor-p = 0.7455870818080153

Assim, fundamentado no teste acima, como no teste de Bartlett o valor-p = 0.482022473018619 $> \alpha = 0.05$ e no teste de Levene o valor-p = 0.7455870818080153 $> \alpha = 0.05$, não rejeitamos a hipótese nula em ambos, ou seja, ao nível de significância de 5% concluímos que os resíduos são homocedásticos.

Portanto, encontramos um modelo que segue as suposições iniciais de independência, linearidade, normalidade e homocedasticidade dos resíduos.

2.5 Tabela ANOVA

Nessa seção, a fim de verificar se todas as covariáveis do modelo são significativas, à um nível de significância de 5%, geramos a Tabela ANOVA.

	Graus de liberdade	**Soma dos quadrados**	**Quadrados médios**	**F**	**Pr(>F)**
x'_1	1.0	115.365456	115.365456	726.559167	$7.818561 \cdot 10^{-69}$
x'_2	1.0	32.565741	32.565741	205.095517	$1.434448 \cdot 10^{-32}$
x'_3	1.0	28.040172	28.040172	176.593972	$2.274409 \cdot 10^{-29}$
x'_4	1.0	17.373063	17.373063	109.413672	$9.669343 \cdot 10^{-21}$
x'_5	1.0	1.247805	1.247805	7.858538	$5.551135 \cdot 10^{-03}$
x'_6	1.0	1.930096	1.930096	12.155537	$6.003785 \cdot 10^{-04}$
Resíduos	202.0	32.074225	0.158783		

Tabela 2.5: Tabela ANOVA.

Através da observação da Tabela 2.5 acima, percebemos que todas as covariáveis possuem um valor-p muito menor do que o nível de significância ($\alpha = 0.05$). Portanto, podemos concluir que todas essas covariáveis são significativas para o modelo.

2.6 Modelo Ajustado Final

Como encontramos um modelo (através de transformações simultaneamente em Y e nas covariáveis) que segue as suposições iniciais de independência, linearidade, normalidade e homocedasticidade dos resíduos e verificamos que todas as covariáveis presentes eram significativas, este será nosso modelo ajustado final, o qual é dado por:

$$\hat{Y'}_i = 2.2433 + 7.8224\frac{1}{x_{i1}} + 0.0065\sqrt{x_{i2}} + 0.0080\sqrt{x_{i3}} + 0.0959\sqrt{x_{i4}} + 0.0192\sqrt{x_{i5}} + 0.0622\sqrt{x_{i6}},$$

com $i = 1, ..., 209$ e $Y'_i = log(Y_i + 0.001)$.

Capítulo 3

Conclusão

Com base nos resultados obtidos no Capítulo anterior, podemos agora obter conclusões sobre o trabalho, o qual tinha como objetivo analisar uma possível relação entre o desempenho relativo de uma CPU e 6 outros fatores, sendo eles: o tempo de ciclo da máquina (em nanossegundos), a memória principal mínima (em kilobytes), a memória principal máxima (em kilobytes), a memória cache (em kilobytes), os canais mínimos (em unidades) e os canais máximos (em unidades).

Ao ajustar pela primeira vez o modelo e fazer a análise de resíduos, percebemos que algumas condições iniciais impostas sobre os resíduos (normalidade, homocedasticidade e linearidade) não estavam satisfeitas. Assim, a fim de buscar um modelo apropriado, fizemos transformações em Y e nas covariáveis, nessa ordem, e com a análise de resíduos concluímos que o modelo ajustado final ficou de acordo com o que era necessário.

Após isso, geramos a Tabela ANOVA com a finalidade de verificar se todas as covariáveis do modelo eram significativas, e constatamos que todas as 6 a princípio usadas eram.

Ademais, como o modelo ajustado final é dado por

$$\hat{Y'}_i = 2.2433 + 7.8224\frac{1}{x_{i1}} + 0.0065\sqrt{x_{i2}} + 0.0080\sqrt{x_{i3}} + 0.0959\sqrt{x_{i4}} + 0.0192\sqrt{x_{i5}} + 0.0622\sqrt{x_{i6}},$$

com $i = 1, ..., 209$ e $Y'_i = log(Y_i + 0.001)$, temos que o log do desempenho relativo de uma CPU (somado de 0.001) é previsto através do inverso do tempo de ciclo da máquina e da raiz de 5 outros fatores: memória principal mínima, memória principal máxima, memória cache, canais mínimos e canais máximos.

Apêndice A

Código

Para esse trabalho, o seguinte código em linguagem Python foi utilizado:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy import stats
from yellowbrick.regressor import ResidualsPlot
import random
from sklearn.linear_model import LinearRegression
import seaborn as sns
from statsmodels.formula.api import ols
import statsmodels.api as sm
from numpy.random import seed
from numpy.random import randn
from statsmodels.graphics.gofplots import qqplot
from matplotlib import pyplot
import math
import scipy.stats as stats
from scipy.stats import shapiro
from scipy.stats import anderson
from scipy.stats import levene

# Importando a base de dados
dados = pd.read_excel('base_trabalho1_regressao.xlsx')
dados

# Obtendo as medidas descritivas
round(dados.describe(),2)

# Separando os dados de Y e das covariaveis
Y = dados['Y']
Y = Y.values
covariaveis_df = dados.drop(columns=["Y"])
covariaveis = covariaveis_df.iloc[:,list(range(6))].values

# Graficos para analise descritiva
```

```
sns.heatmap(covariaveis_df.corr(), annot=True, square = True, cmap = "
    Blues")
sns.pairplot(covariaveis_df)
sns.boxplot(data=covariaveis_df, color = "b")
plt.show()
sns.histplot(data=dados, x = "Y", color = "b", )
plt.show()

# Ajustando o primeiro modelo
modelo = ols(formula='Y~x1+x2+x3+x4+x5+x6', data=dados).fit()
modelo_intercept = modelo.params["Intercept"]
modelo_coefs = [modelo.params["x1"],modelo.params["x2"],modelo.params["
    x3"],modelo.params["x4"],modelo.params["x5"],modelo.params["x6"]]

# Obtendo Y ajustado e os residuos
Y_ajustado = modelo.predict()
residuos = Y - Y_ajustado

# Grafico Residuos vs Y ajustado
plt.plot(Y_ajustado, residuos, "bo")
plt.plot([min(Y_ajustado), max(Y_ajustado)],[0,0], color = "red")
plt.xlabel("Y ajustado")
plt.ylabel("Res duos")
plt.show()

# Grafico Normal Q-Q
qqplot(residuos, line='s')
pyplot.show()

# Testes de normalidade
shapiro(residuos)[1]
anderson(residuos)

# Teste de homocedasticidade
res1_1 = np.sort(residuos)[0:104]
res2_1 = np.sort(residuos)[104:209]
stats.bartlett(res1_1, res2_1)

# Transformacao em Y
Y_transf = []
for j in range(0,209):
    Y_transf.append(math.log(Y[j]+0.001))

# Ajuste do segundo modelo
modelo2 = ols(formula='Y_transf~x1+x2+x3+x4+x5+x6', data=dados).fit()
modelo2_intercept = modelo2.params["Intercept"]
modelo2_coefs = [modelo2.params["x1"],modelo2.params["x2"],modelo2.
    params["x3"],modelo2.params["x4"],modelo2.params["x5"],modelo2.params
    ["x6"]]
```

```
80
81 # Obtendo Y ajustado e os residuos
82 Y_ajustado2 = modelo2.predict()
83 residuos2 = Y_transf - Y_ajustado2
84
85 # Grafico Residuos vs Y ajustado
86 plt.plot(Y_ajustado2, residuos2, "bo")
87 plt.plot([min(Y_ajustado2), max(Y_ajustado2)],[0,0], color = "red")
88 plt.xlabel("Y' ajustado")
89 plt.ylabel("Res duos")
90
91 # Grafico Normal Q-Q
92 qqplot(residuos2, line='s')
93 pyplot.show()
94
95 # Testes de normalidade
96 shapiro(residuos2)[1]
97 anderson(residuos2)
98
99 # Testes de homocedasticidade
100 res1_2 = np.sort(residuos2)[0:104]
101 res2_2 = np.sort(residuos2)[104:209]
102 stats.bartlett(res1_2, res2_2)
103 levene(res1_2,res2_2,center='median')
104
105 # Graficos de dispersao de Y' vs as covariaveis
106 plt.subplots_adjust(wspace=0.3,hspace=0.3)
107
108 plt.subplot(2, 3, 1)
109 plt.scatter(covariaveis_df["x1"],Y_transf)
110 plt.xlabel("x1")
111 plt.ylabel("Y'")
112
113 plt.subplot(2, 3, 2)
114 plt.scatter(covariaveis_df["x2"],Y_transf)
115 plt.xlabel("x2")
116 plt.ylabel("Y'")
117
118 plt.subplot(2, 3, 3)
119 plt.scatter(covariaveis_df["x3"],Y_transf)
120 plt.xlabel("x3")
121 plt.ylabel("Y'")
122
123 plt.subplot(2, 3, 4)
124 plt.scatter(covariaveis_df["x4"],Y_transf)
125 plt.xlabel("x4")
126 plt.ylabel("Y'")
127
128 plt.subplot(2, 3, 5)
```

```
plt.scatter(covariaveis_df["x5"],Y_transf)
plt.xlabel("x5")
plt.ylabel("Y'")

plt.subplot(2, 3, 6)
plt.scatter(covariaveis_df["x6"],Y_transf)
plt.xlabel("x6")
plt.ylabel("Y'")

# Tranfomacao nas covariaveis
x1 = list(covariaveis_df["x1"])
x1_transf = []
for k in range(0,209):
    x1_transf.append(1/x1[k])

x2 = list(covariaveis_df["x2"])
x2_transf = []
for k in range(0,209):
    x2_transf.append(math.sqrt(x2[k]))

x3 = list(covariaveis_df["x3"])
x3_transf = []
for k in range(0,209):
    x3_transf.append(math.sqrt(x3[k]))

x4 = list(covariaveis_df["x4"])
x4_transf = []
for k in range(0,209):
    x4_transf.append(math.sqrt(x4[k]))

x5 = list(covariaveis_df["x5"])
x5_transf = []
for k in range(0,209):
    x5_transf.append(math.sqrt(x5[k]))

x6 = list(covariaveis_df["x6"])
x6_transf = []
for k in range(0,209):
    x6_transf.append(math.sqrt(x6[k]))

x1_transf = np.array(x1_transf).reshape(-1,1)
x2_transf = np.array(x2_transf).reshape(-1,1)
x3_transf = np.array(x3_transf).reshape(-1,1)
x4_transf = np.array(x4_transf).reshape(-1,1)
x5_transf = np.array(x5_transf).reshape(-1,1)
x6_transf = np.array(x6_transf).reshape(-1,1)

covariaveis_transf = np.hstack((x1_transf,x2_transf,x3_transf,x4_transf,
   x5_transf,x6_transf))
```

```
covariaveis_transf_df = pd.DataFrame(covariaveis_transf, columns= ["
    x1_transf", "x2_transf", "x3_transf", "x4_transf", "x5_transf", "
    x6_transf"])
covariaveis_transf = covariaveis_transf_df.iloc[:,list(range(6))].values

# Ajuste do terceiro modelo
modelo3 = ols(formula='Y_transf~x1_transf+x2_transf+x3_transf+x4_transf+
    x5_transf+x6_transf', data=covariaveis_transf_df).fit()
modelo3_intercept = modelo3.params["Intercept"]
modelo3_coefs = [modelo3.params["x1_transf"],modelo3.params["x2_transf"
    ],modelo3.params["x3_transf"],modelo3.params["x4_transf"],modelo3.
    params["x5_transf"],modelo3.params["x6_transf"]]

# Obtendo Y ajustado e os residuos
Y_ajustado3 = modelo3.predict()
residuos3 = Y_transf - Y_ajustado3

# Grafico Res duos vs Y ajustado
plt.plot(Y_ajustado3, residuos3, "bo")
plt.plot([min(Y_ajustado3), max(Y_ajustado3)],[0,0], color = "red")
plt.xlabel("Y' ajustado")
plt.ylabel("Res duos")

# Grafico Normal Q-Q
qqplot(residuos3, line='s')
pyplot.show()

# Testes de normalidade
shapiro(residuos3)[1]
anderson(residuos3)

# Testes de homocedasticidade
res1_3 = np.sort(residuos3)[0:104]
res2_3 = np.sort(residuos3)[104:209]
stats.bartlett(res1_3, res2_3)
levene(res1_3,res2_3,center='median')

# Tabela ANOVA
sm.stats.anova_lm(modelo3)
```