Relatório

Projeto do Curso

Antonny Victor da Silva, DRE: 120031917

7 de janeiro de 2023

Resumo

Este relatório tem como objetivo analisar um conjunto de dados aplicando a teoria aprendida em classe. É muito importante que seja realizada uma análise crítica dos resultados encontrados.
O projeto será baseado em dados reais fornecidos por um provedor de Internet de médio porte. Os dados representam a taxa de dados enviados em bps (taxa de upload) e a taxa de dados recebidos em bps (taxa de download) de/por um dispositivo na casa de um usuário do provedor. Dois tipos de dispositivos devem ser analisados: Smart-TV e Chromecast.

As Seções aqui encontradas são todas referentes às Seções do arquivo .ipynb disponibilizado no link do projeto abaixo.

https://github.com/antonnyvictor18/Probest

Sumário

1 Introdução

1.1 Importação de Bibliotecas Necessárias

Primeiramente, para poder trabalhar com esses dados, é necessário que algumas bibliotecas sejam importadas e é aqui nesta seção que realizo essas importações.

1.2 Declaração da Função do Método de Sturges

Como, posteriormente, será necessário estimar o tamanho do bin de forma que o histograma represente, adequadamente, os dados estudados. Declarei a função do método de Sturges apresentado em aula, obtido da seguinte forma:

$$nc = 1 + 3,3 \times \log_{10} N$$

Sendo nc o número de classes e N o número de pontos de dados.

1.3 Carregamento das Bases de Dados

Como o ambiente de trabalho está configurado, nesta seção do arquivo eu carrego as bases de dados fornecidas (dataset_chromecast.csv e dataset_smart-tv.csv) utilizando as funções da biblioteca pandas.

1.4 Familiarizando-se com as Tabelas

Neste momento, eu apenas tento visualizar, superficialmente, ambas as tabelas fornecidas pelo projeto a fim de familiarizar-me com elas.

1.5 Tratamento de Dados

Com as análises feitas acima, podemos concluir que nossas tabelas ainda não estão da forma que gostaríamos. Portanto, neste bloco, fiz um reescalonamento de dados para $\log_{10}$ nas colunas "bytes_up" e "bytes_down" de ambas as tabelas, somando 1 dentro do log de acordo com a relação abaixo:

$$nova\_coluna = \log_{10}(1 + dados\_smartTv[\text{"}bytes\_down\text{"}])$$

Em seguida, criei, também para ambas as tabelas, uma nova coluna chmada "hour" que recebe apenas o valor da hora (sem considerar minutos e segundos) da coluna "date_hour". a fim de facilitar as futuras extrações de dados.

2 Estatísticas Gerais

O objetivo deste estudo é avaliar os dados sem considerar o horário em que foram gerados, ou seja, considerar todos os dados de cada um dos arquivos para obter as estatísticas descritas a seguir. Para cada tipo de dispositivo, Smart-TV e Chromecast, calcular: Histograma, Função Distribuição Empírica, Box Plot, Média, Variância e Desvio Padrão, para a taxa de upload e taxa de download.

2.1 Histograma

- O primeiro passo é coletar os dados a serem analisados e organizá-los, a fim de facilitar a interpretação dos mesmos.
- Calcular a amplitude (diferença entre o maior e o menor valor dos dados utilizados).
- Definir os números de bins (as barras verticais). Para ordenar a quantidade de classes, utilizamos o método de Sturges.
- Calcular o intervalo das classes (dividindo-se a amplitude pelo número de classes).
- Determinar os limites das classes. Aqui, você seleciona o menor valor da amostra (se for mais viável, ele pode ser arredondado para baixo). Para calcular o limite superior da primeira classe, soma-se o valor do intervalo de classe.
- Montar o histograma.

2.1.1 Frequência x $\log_{10}$(Bytes Up) Chromecast

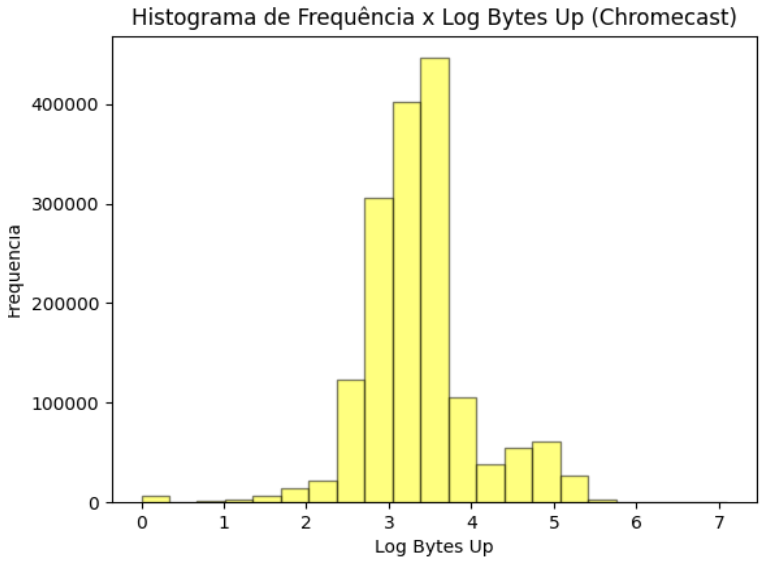


Figura 1

2.1.2 Frequência x $\log_{10}$(Bytes Up) Smart TV

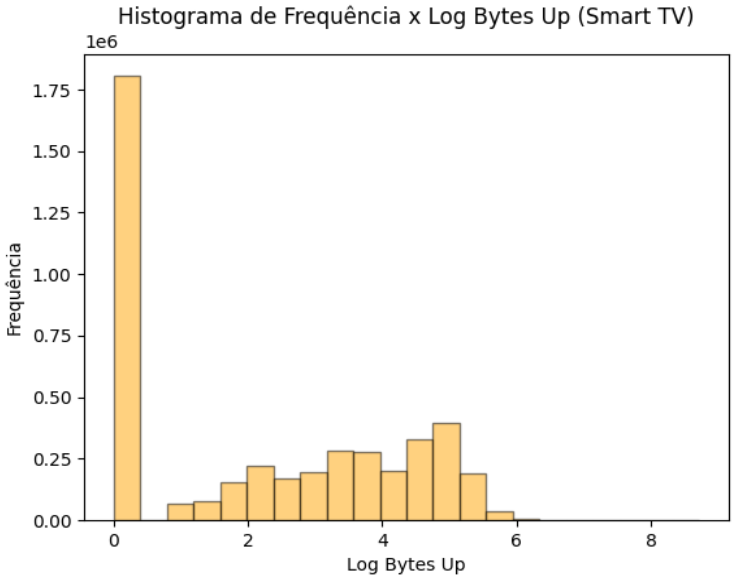


Figura 2

2.1.3 Frequência x $\log_{10}$(Bytes Down) Chromecast

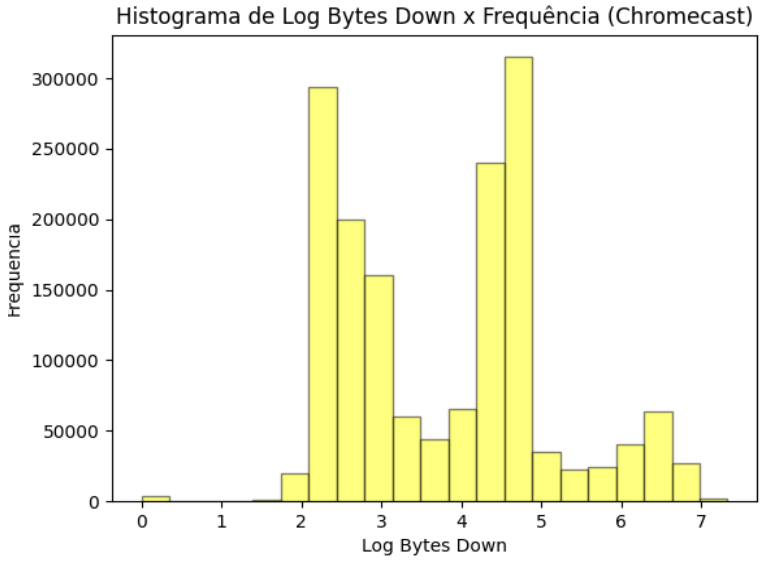


Figura 3

2.1.4 Frequência x $\log_{10}$(Bytes Down) Smart TV

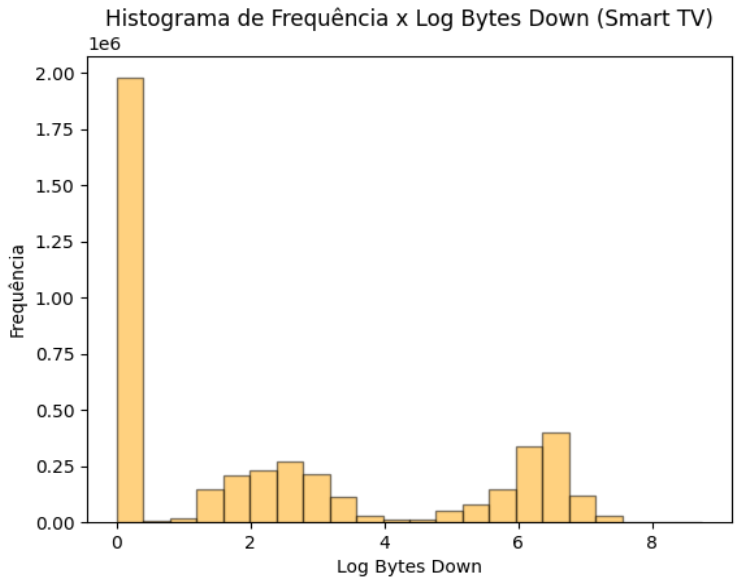


Figura 4

2.2 Função Distribuição Empírica

A Função Distribuição Empírica foi obtida da seguinte forma:

$$F_x(X) = P[X < x]$$

Sendo que o valor de $\forall x F_X(x) \in [0, 1]$ e P é uma Função de Probabilidade.

2.2.1 $\log_{10}$(Bytes Up) Chromecast

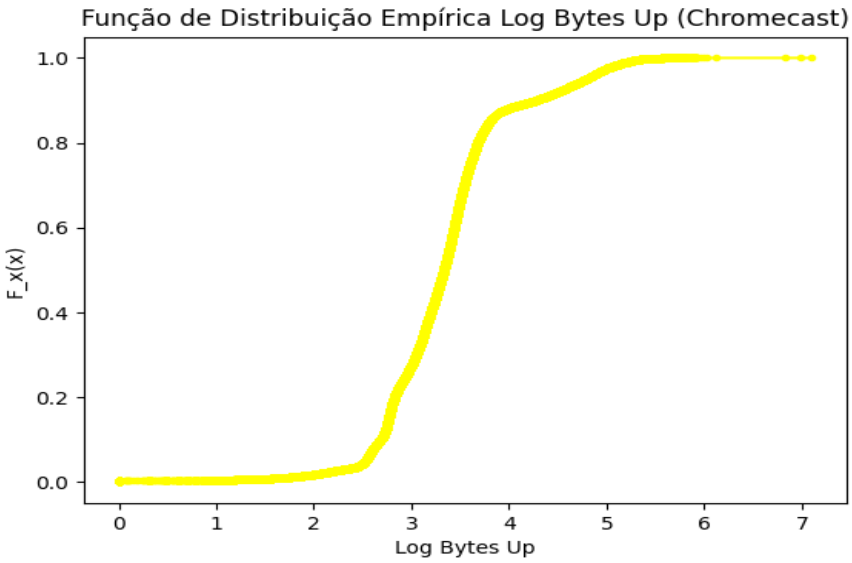


Figura 5

2.2.2 $\log_{10}$(Bytes Up) Smart TV

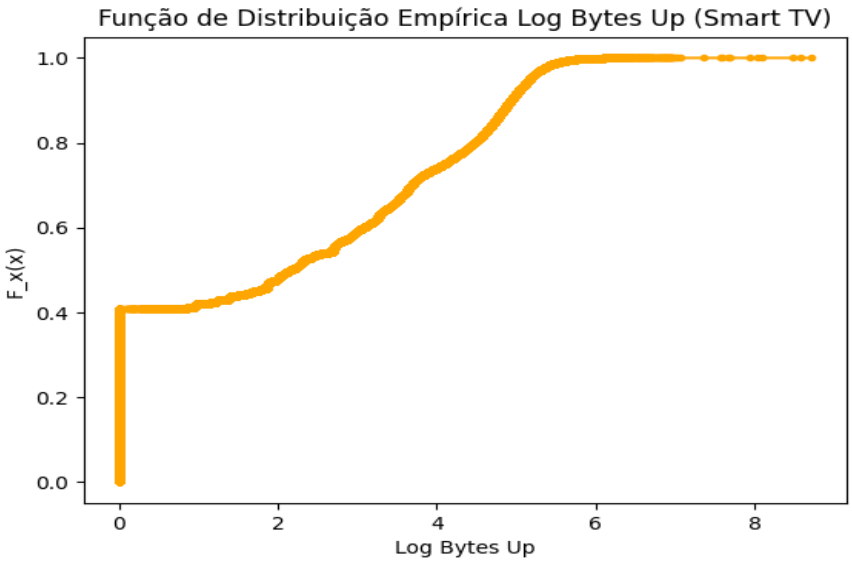


Figura 6

2.2.3 $\log_{10}$(Bytes Down) Chromecast

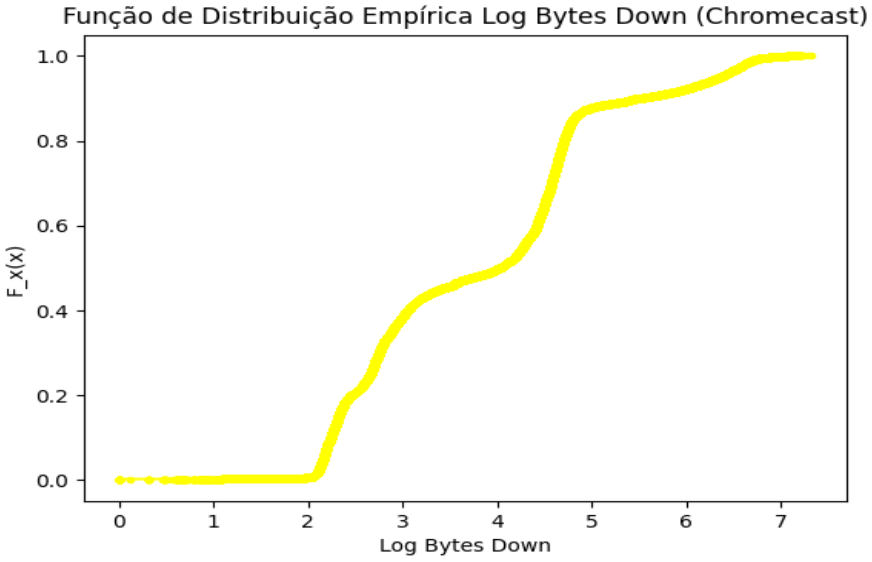


Figura 7

2.2.4 $\log_{10}$(Bytes Down) Smart TV

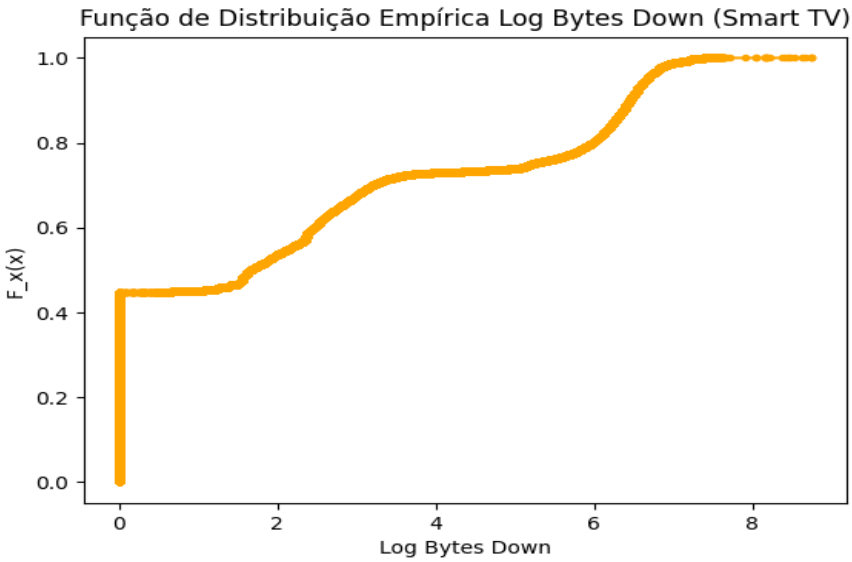


Figura 8

2.3 Box Plot

Box Plot são úteis para identificar outliers e para comparar distribuições. Para construí-lo, há várias maneiras, mas o início dá-se pelo cálculo do primeiro quartil, a mediana e o terceiro quartil. Sendo assim, percentil populacional de proporção p (ou de percentagem $100p\%$) é o valor P_p tal que:

$$P[X \leq P_p] \geq p$$
e
$$P[X \geq P_p] \geq 1 - p.$$

O primeiro quartil é pois o percentil de proporção 0.25 (25%), isto é, é o valor $P_{0.25}$ tal que a probabilidade da variável X tomar um valor não superior a $P_{0.25}$ é pelo menos 0.25 e, simultaneamente, a probabilidade de a variável X tomar um valor não inferior a $P_{0.25}$ é pelo menos 0.75. Analogamente, o terceiro quartil é o percentil de proporção 0.75 (75%) e o segundo quartil (mediana) é o percentil de proporção 0.5 (50%). Abaixo, vemos uma representação de com um Box Plot é formado:

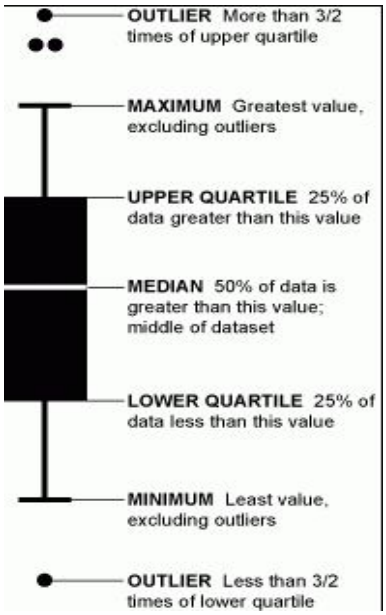


Figura 9

2.3.1 $\log_{10}$(Bytes Up) e $\log_{10}$(Bytes Down) Chromecast

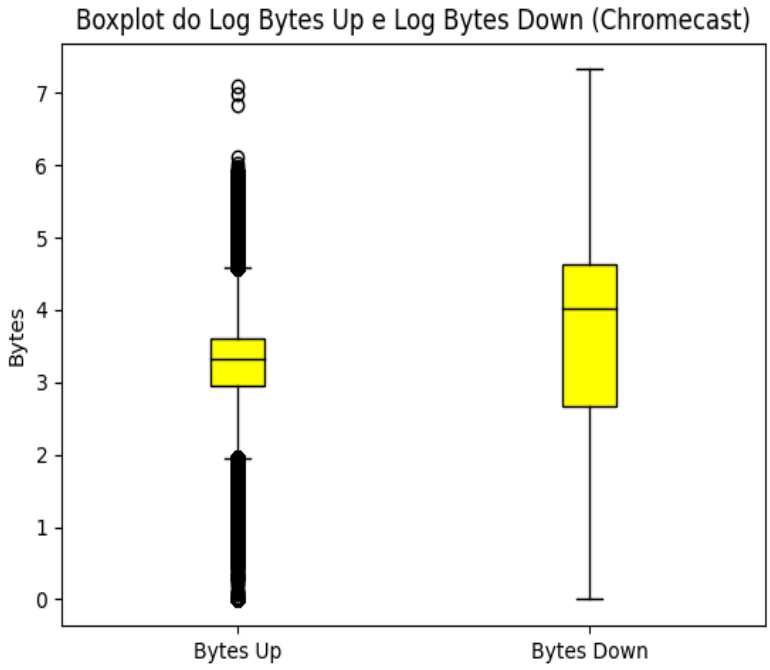


Figura 10

2.3.2 $\log_{10}$(Bytes Up) e $\log_{10}$(Bytes Down) Smart TV

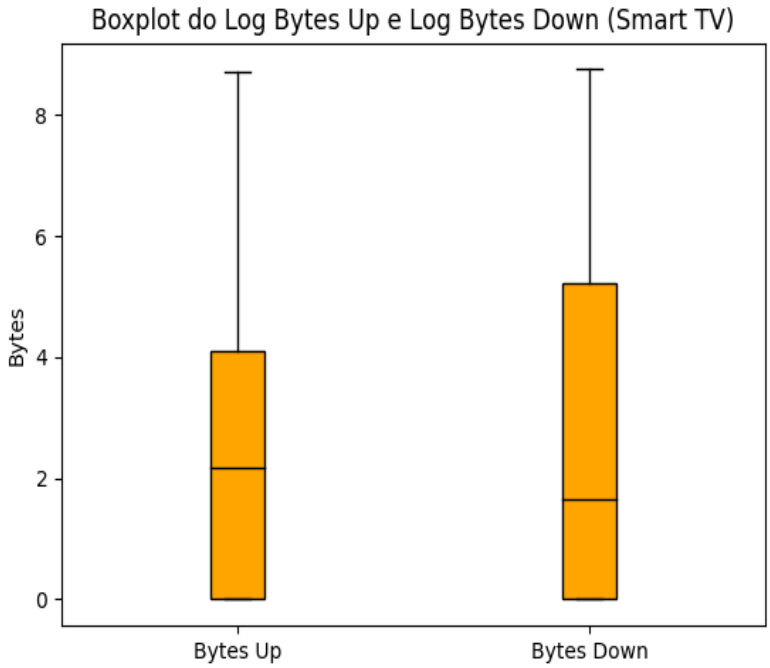


Figura 11

2.4 Média, Variância e Desvio Padrão

A média de um determinado conjunto de dados X = $[x_1, x_2, ..., x_n]$ com n elementos é dado pela seguinte forma:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

Para o cálculo da variância, temos a seguinte definição:

$$var(x) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}$$

Para o cálculo do desvio padrão, temos que:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n}}$$

Vale ressaltar que, para uma melhor vizualização dos dados, arredondei todos os resultados com apenas 3 casas decimais.

2.4.1 Chromecast

Tabela 1: Valores Estatísticos (Chromecast)

Estatística	Log Bytes Up	Log Bytes Down
Média	3.350	3.800
Variância	0.460	1.664
Desvio Padrão	0.678	1.290

2.4.2 Smart TV

Tabela 2: Valores Estatísticos (Smart TV)

Estatística	Log Bytes Up	Log Bytes Down
Média	2.158	2.352
Variância	4.110	6.721
Desvio Padrão	2.027	2.593

2.5 Análise dos dados

À primeira vista, nota-se que há muitos dados com frequência de upload e download 0 em ambos os dispositivos, ainda que isso ocorra com menos regularidade no Chromecast. A partir disso, desconfia-se de que os dispositivos funcionam de forma diferente ao baixar ou enviar arquivos.

Ademais, os histogramas do Chromecast evidenciam a diferença entre o funcionamento do Download e do Upload. Haja vista que o pico de upload acontece na faixa de 1000 bytes/s, enquanto a Smart TV tem seu pico na faixa de 0 bytes/s. Possivelmente, esse fenômeno é observado devido a capacidade de internet dos dispositivos.

Além disso, é possível observar que os comportamentos momentâneos são complementares entre si (Download e Upload), uma vez que eles não mantêm as duas ações em pico. Em respeito ao box plot de download e upload entre os dispositivos, nota-se que as dispersões para a taxa de upload acabam sendo bem diferentes entre si, a concentração dos dados estão em locais distintos também devido às posições do primeiro e terceiro quartil. Ainda nesse sentido, o Boxplot do chromecast tem a presença de outliers bem nítidos. Já para o boxplot de download, ambos possuem uma região de terceiro quartil próxima. Contudo, os outros quartis bem distante entre-si e também possui outliers. Por fim, com relação ao comportamento da Função Distribuição Empírica entre os dispositivos, podemos dizer que são bem diferentes. O Chromecast apresenta um aumento mais tímido enquanto a Smart TV tem um aumento bem visível no 0 que ocorre porque esse aparelho passa muito tempo sem receber download ou realizar upload.

3 Estatísticas por horário

3.1 Boxplot

Para esta tarefa, precisamos, para ambas as bases de dados, fazer uma filtragem de dados que selecione somente as linhas que contenham um determinado horário e, em seguida, poderemos construir nossos gráficos.
Como o dia tem 24 horas, um for loop do Python de range igual a 24 (começa no 0 e termina no 23, incluindo-o).
A forma como obtive o Boxplot segue os mesmos fundamentos explicados na **Seção 2.3.**

3.1.1 Resultados

Os resultados encontrados foram ilustrados abaixo da seguinte forma: As figuras da esquerda (retângulos amarelos) representam o Chromecast e as figuras da direita (retângulos laranjas) representam a Smart TV.

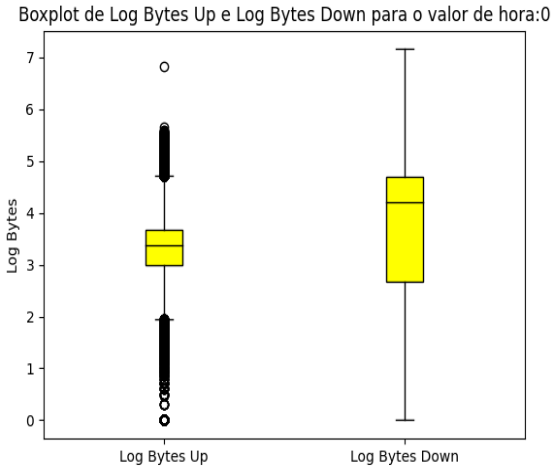


Figura 12

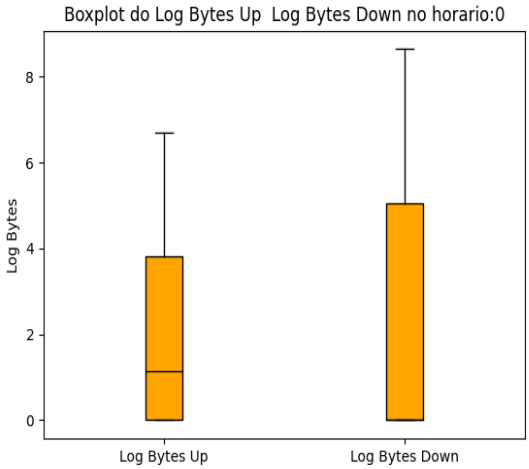


Figura 13

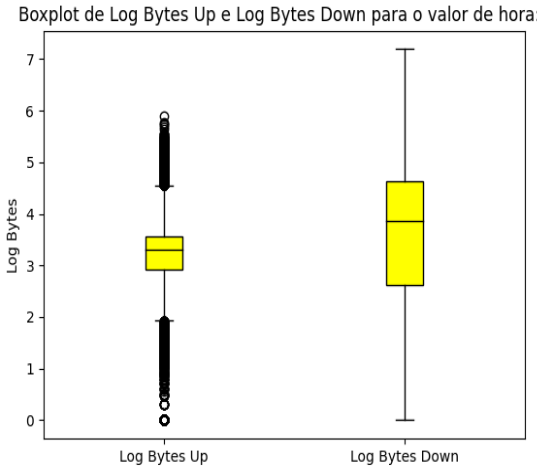


Figura 14

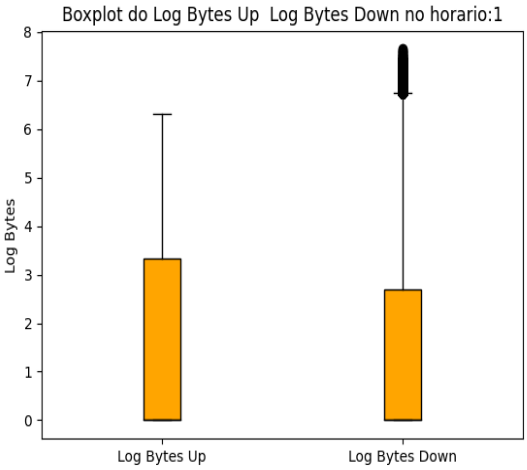


Figura 15

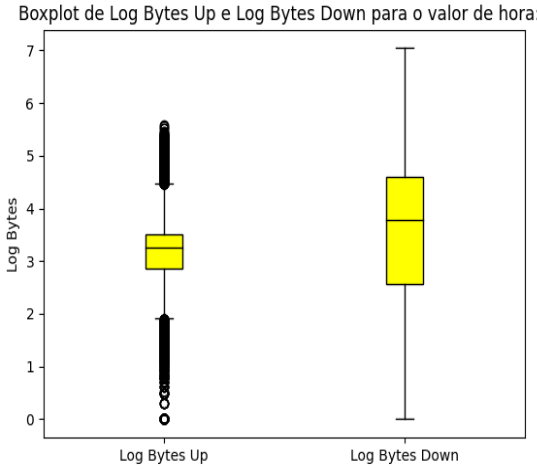


Figura 16

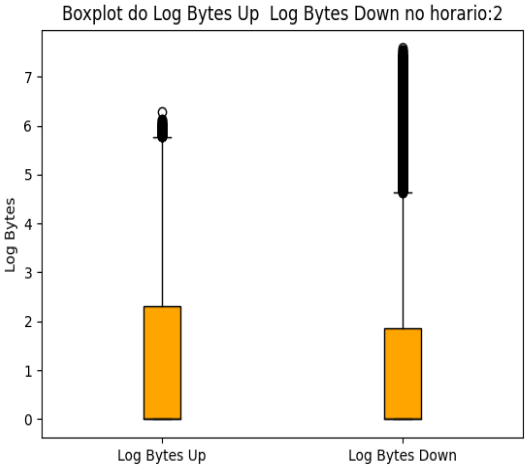
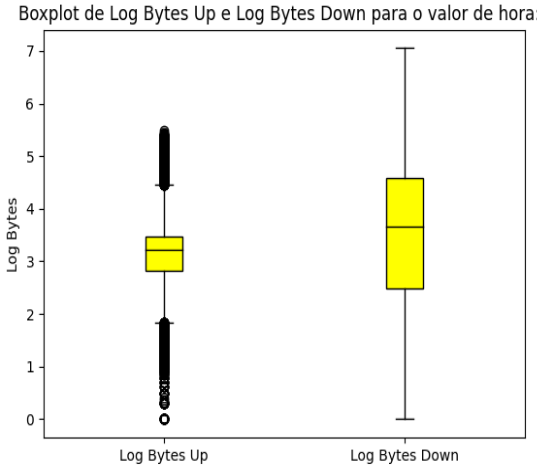


Figura 17



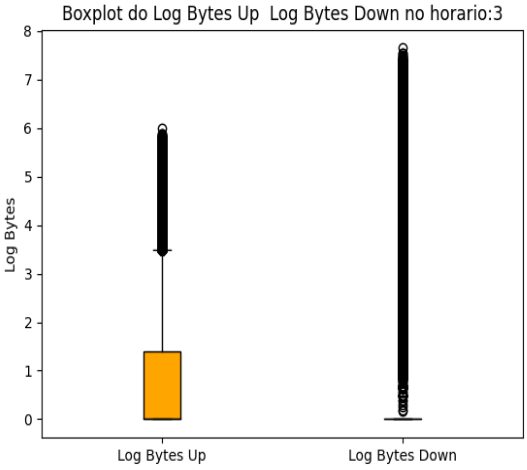


Figura 18

Figura 19

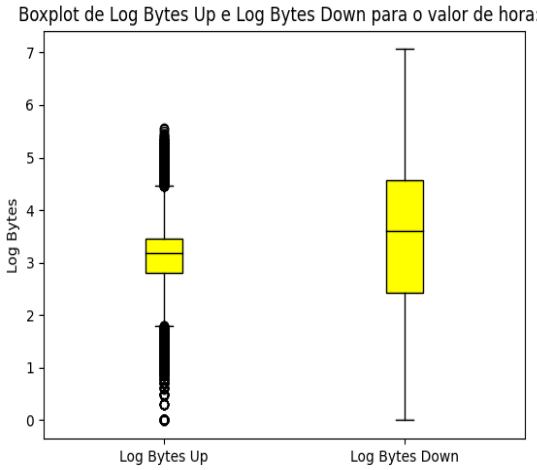


Figura 20

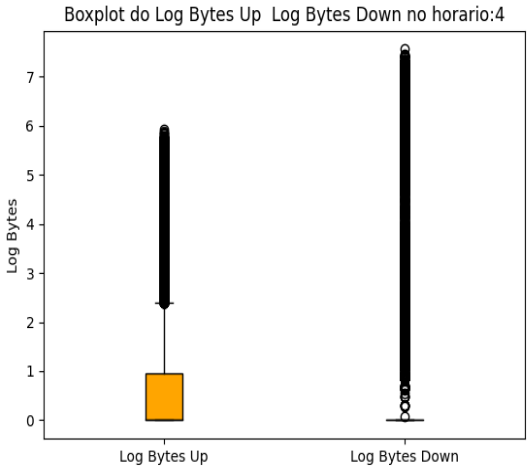


Figura 21

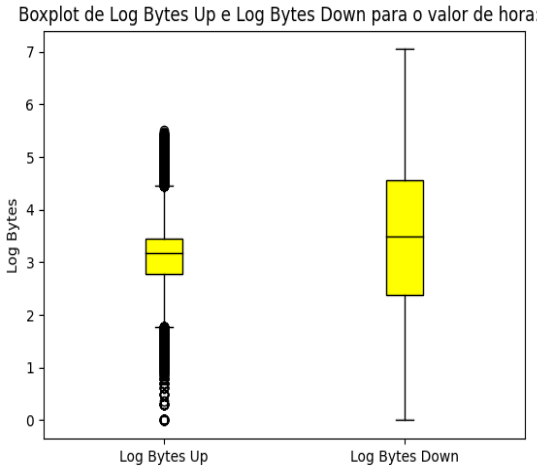


Figura 22

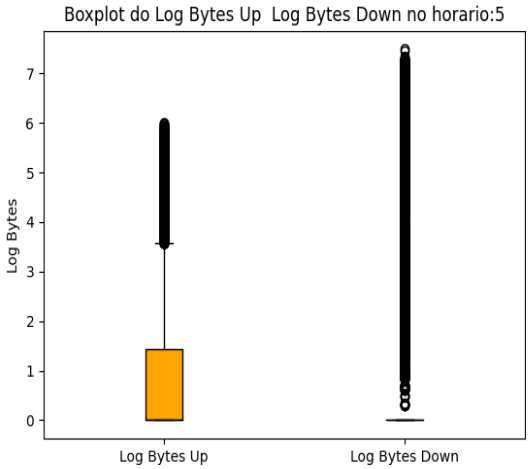


Figura 23

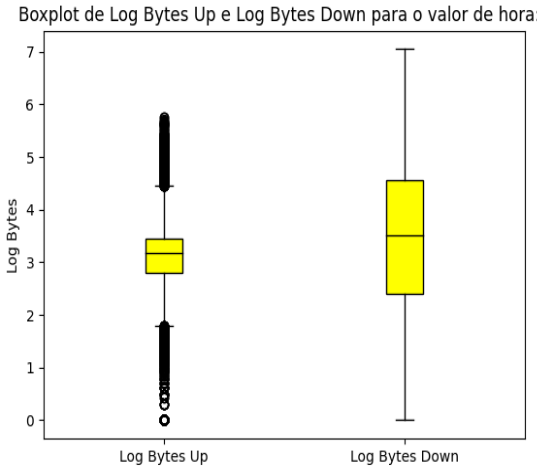


Figura 24

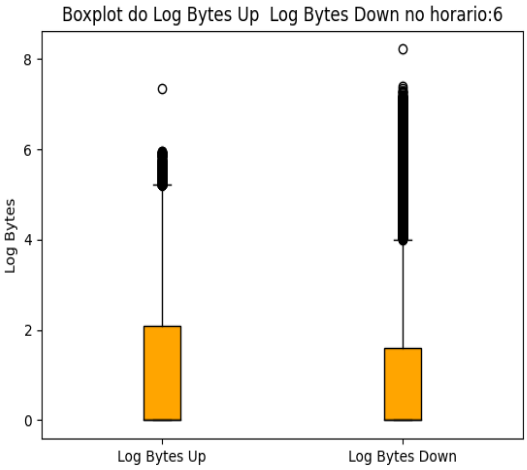


Figura 25

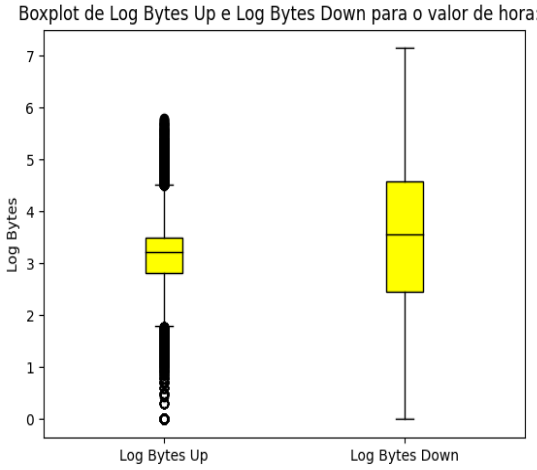


Figura 26

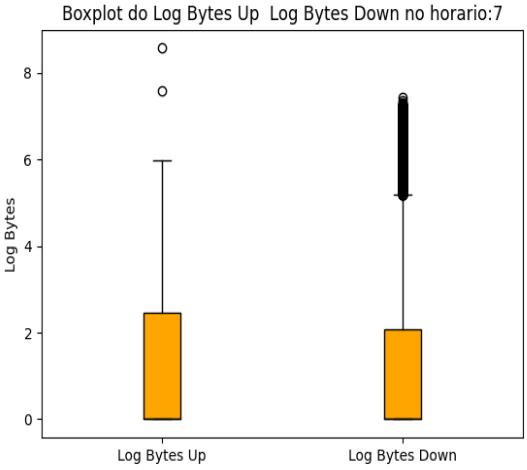
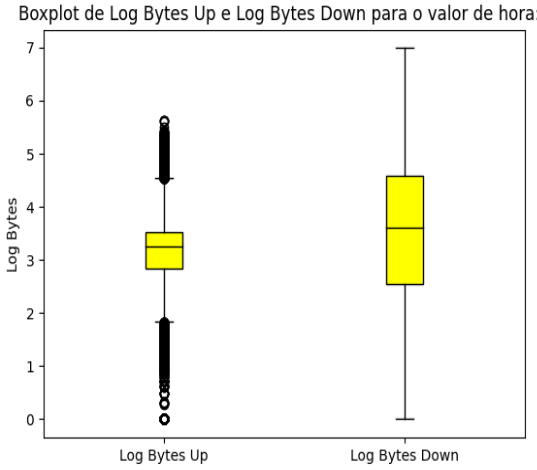


Figura 27



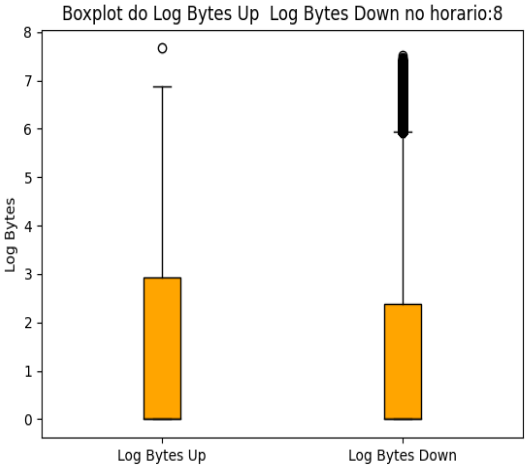
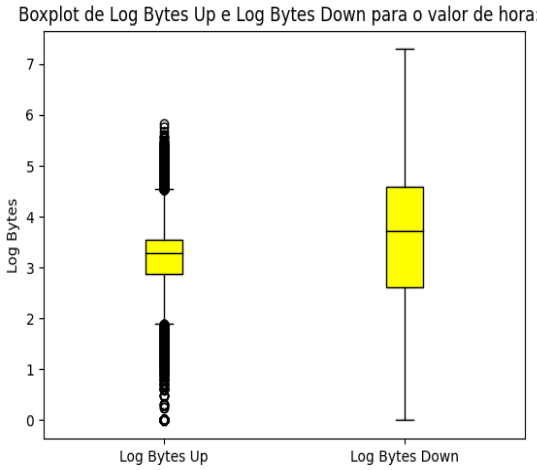


Figura 28

Figura 29



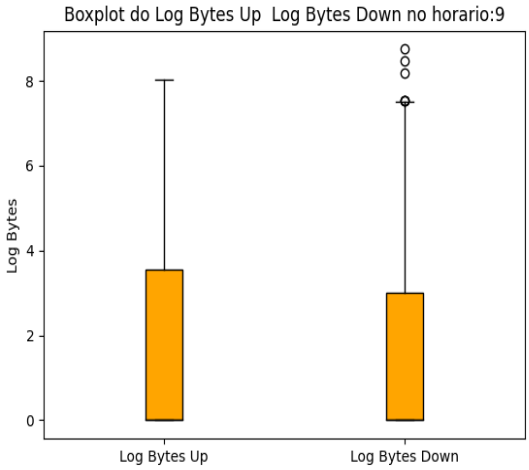
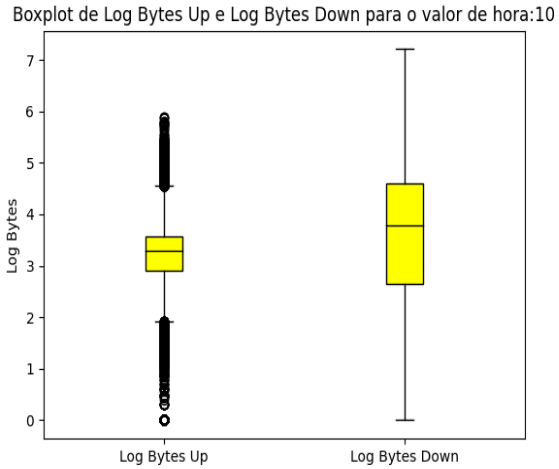


Figura 30

Figura 31



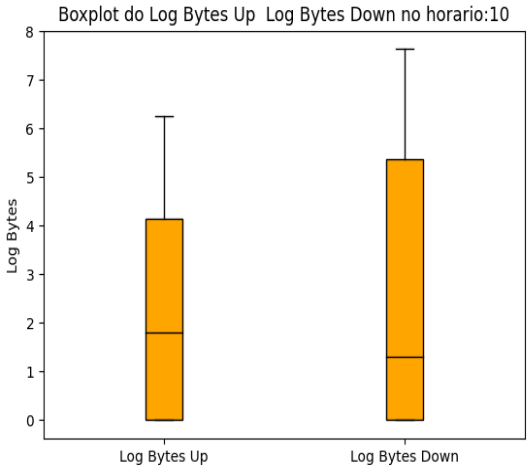


Figura 32

Figura 33

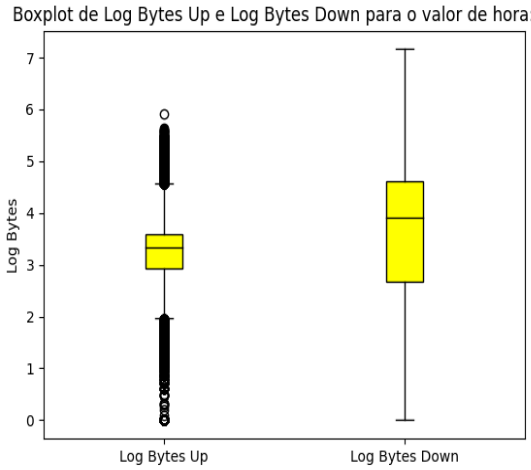


Figura 34

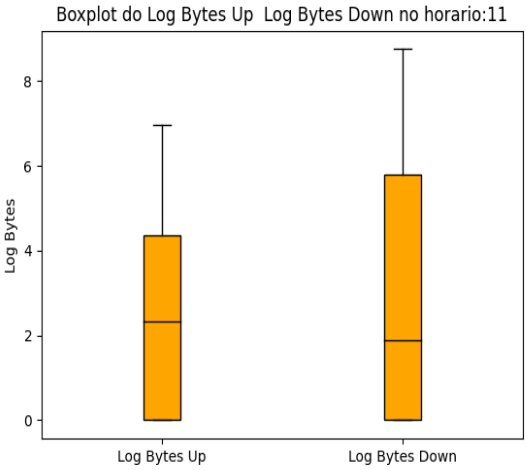


Figura 35

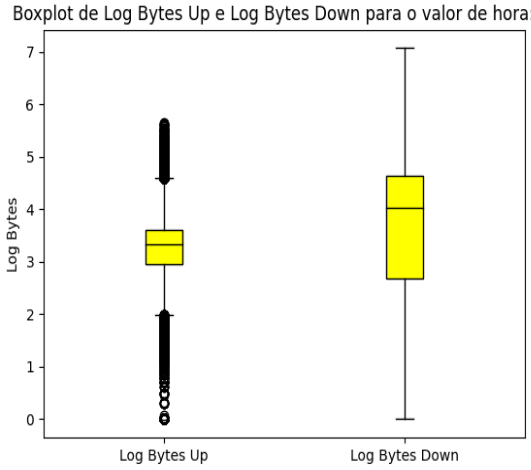


Figura 36

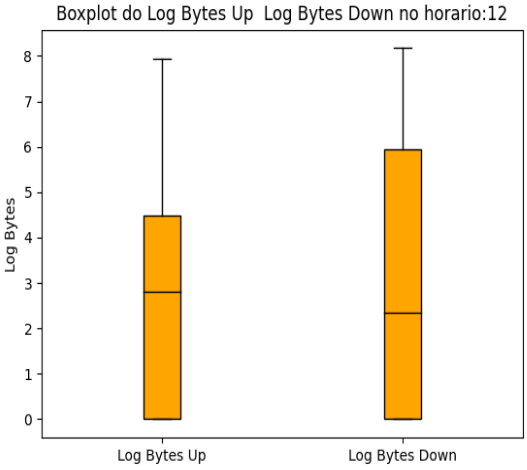
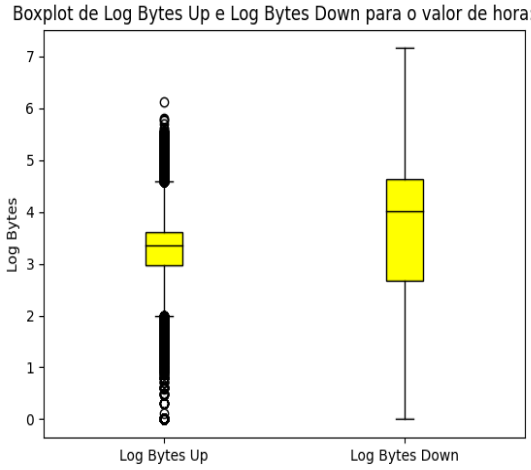


Figura 37



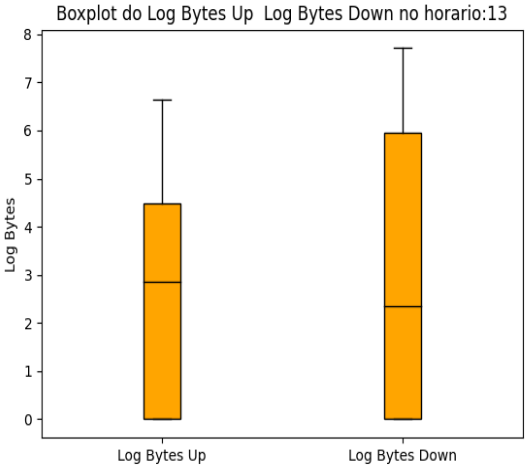
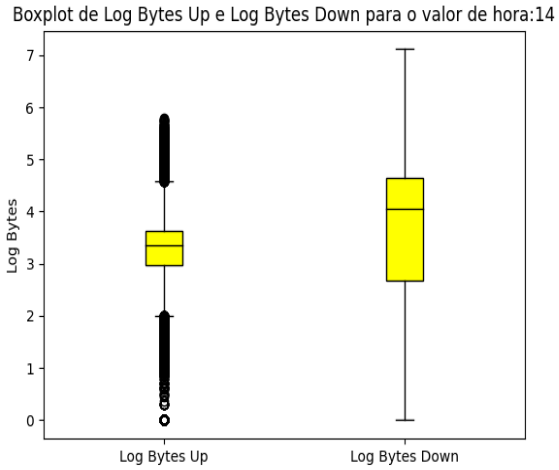


Figura 38

Figura 39



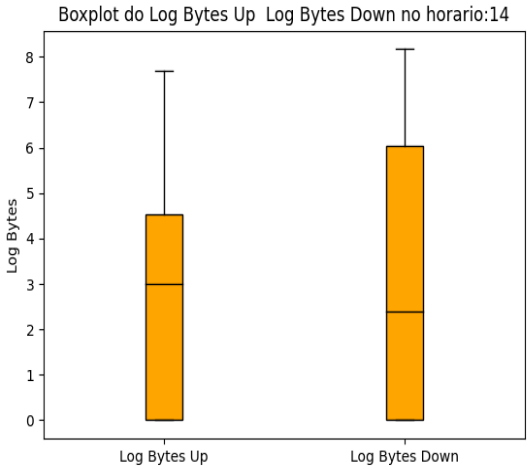
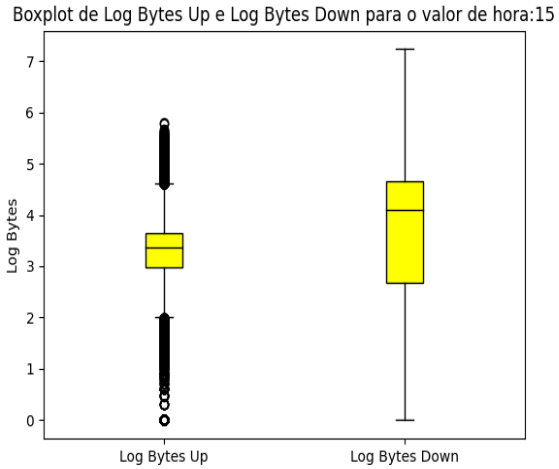


Figura 40

Figura 41



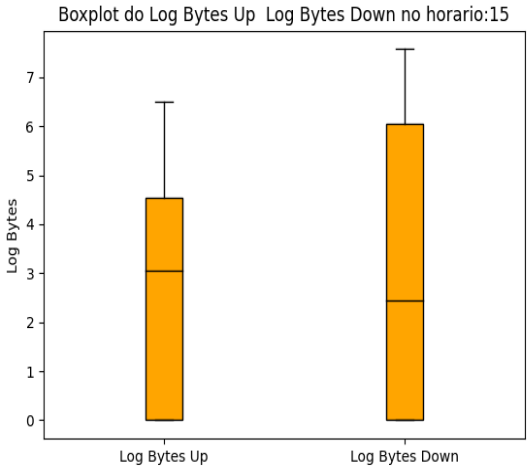


Figura 42

Figura 43

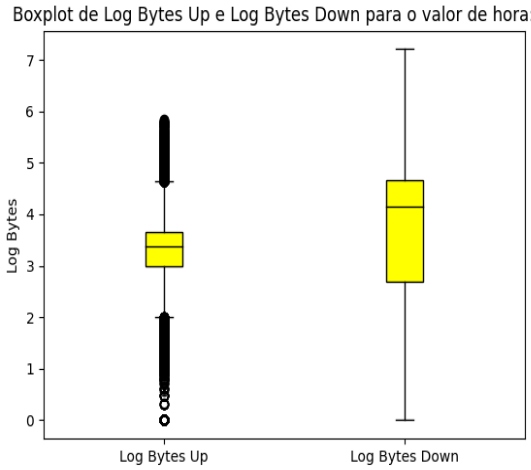


Figura 44

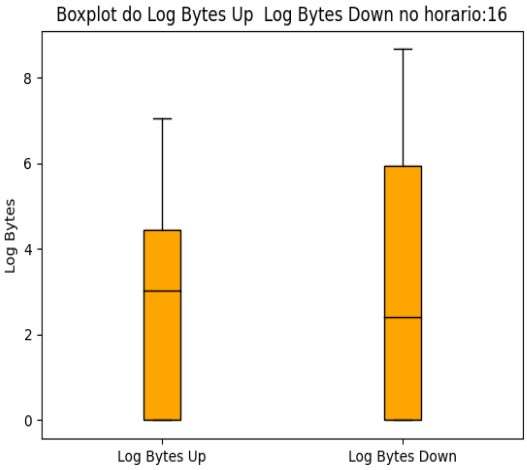


Figura 45

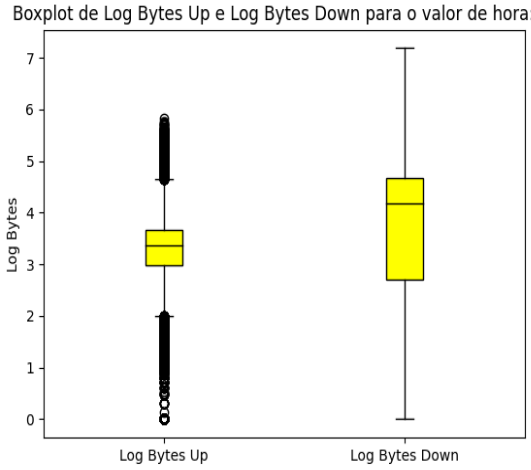


Figura 46

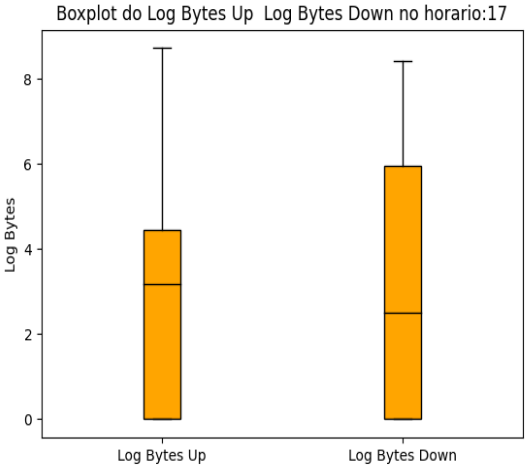
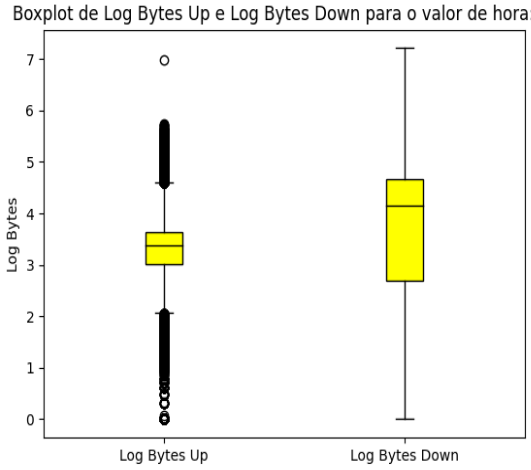


Figura 47



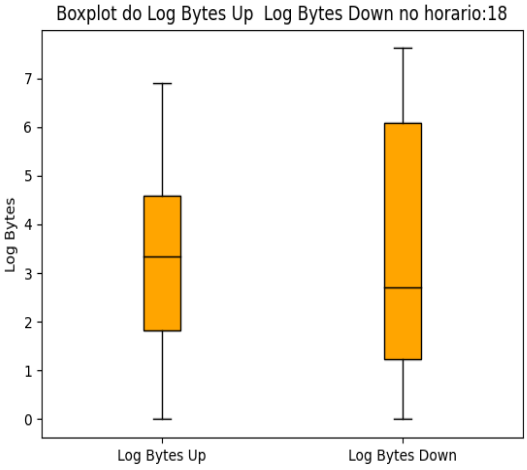
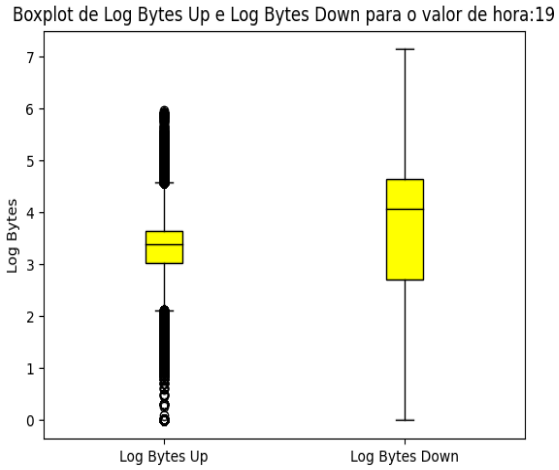


Figura 48

Figura 49



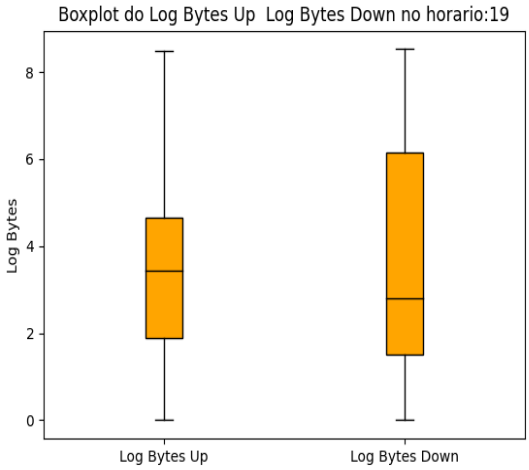
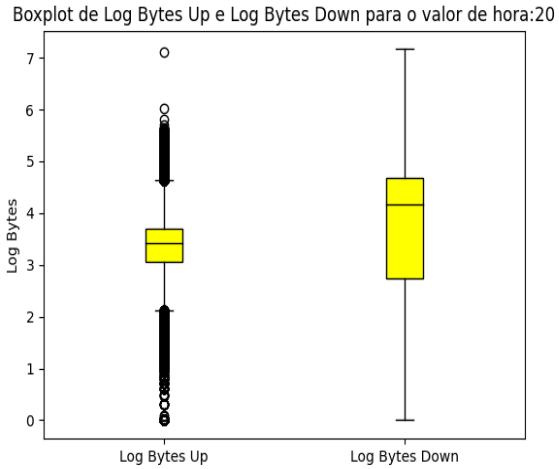


Figura 50

Figura 51



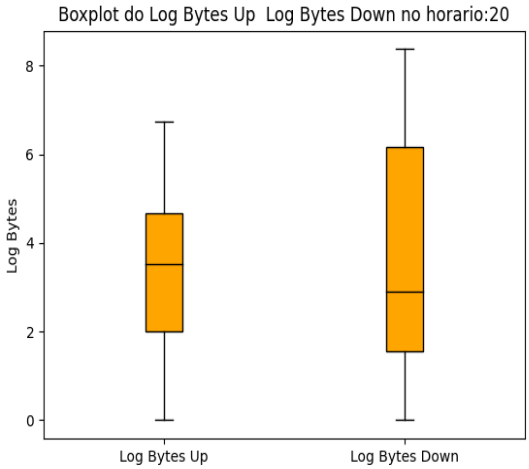


Figura 52

Figura 53

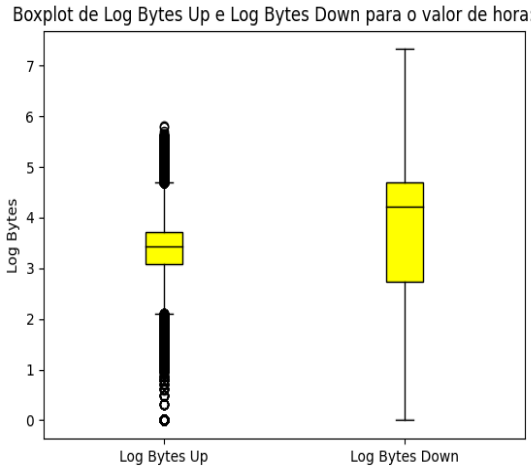


Figura 54

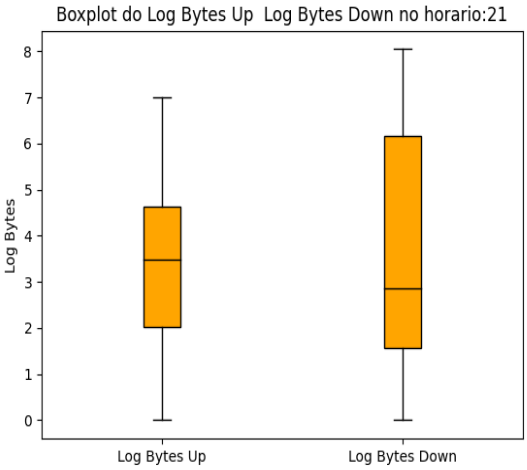


Figura 55

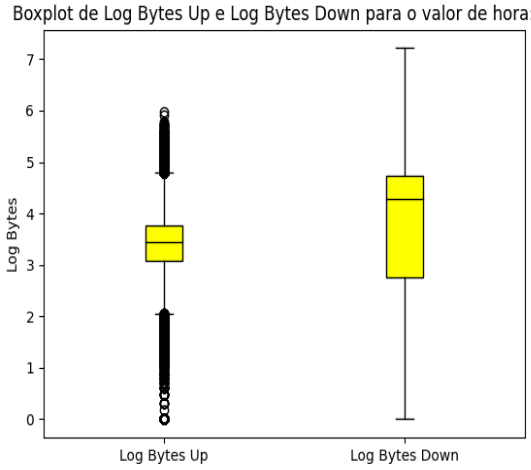


Figura 56

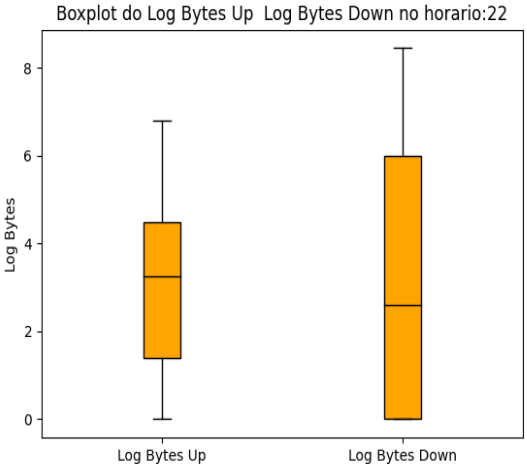
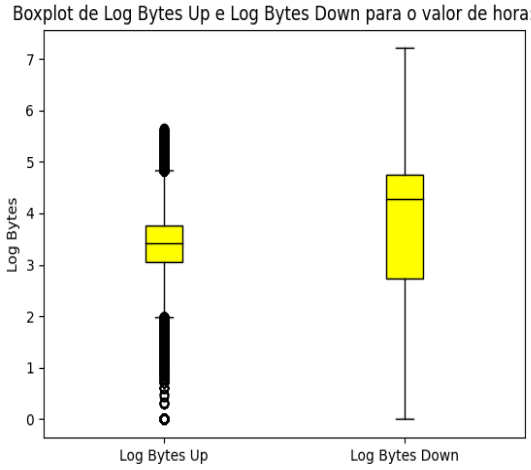


Figura 57



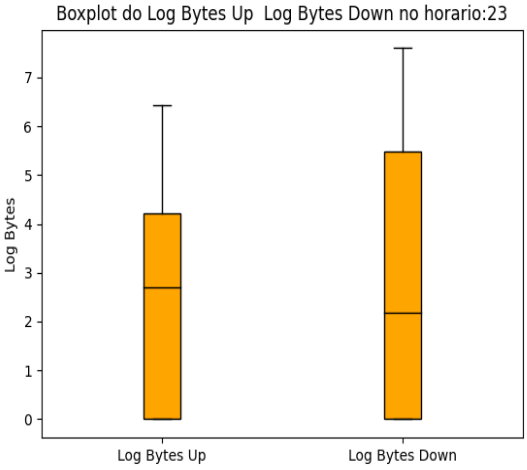


Figura 58 Figura 59

3.2 Média, Variância e Desvio Padrão

Os valores estatísticos desta seção foram obtidos seguindo a teoria explicada na **Seção 2.4**. Contudo, foi necessário fazer uma nova filtragem de dados, agrupando as colunas log_bytes_up e log_bytes_down por cada hora e, em seguida, aplicando as funções de Média, Variância e Desvio Padrão para cada um desses agrupamentos.

3.2.1 Log Bytes Up

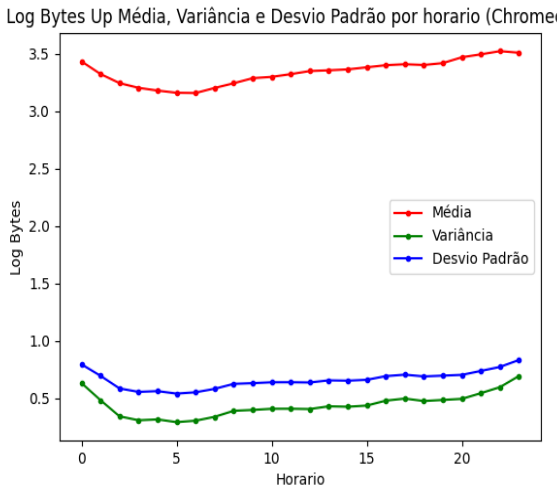


Figura 60

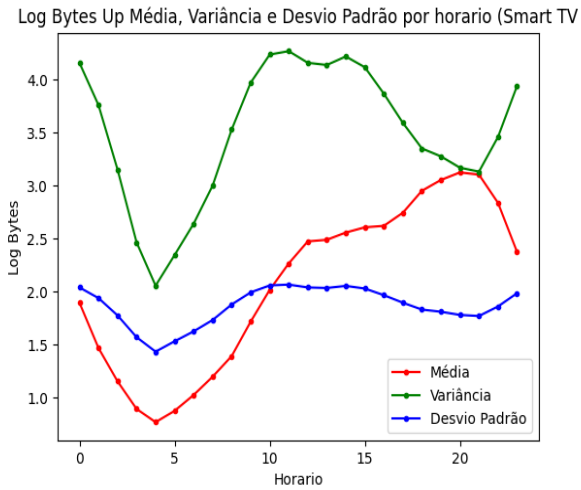


Figura 61

3.2.2 Log Bytes Up

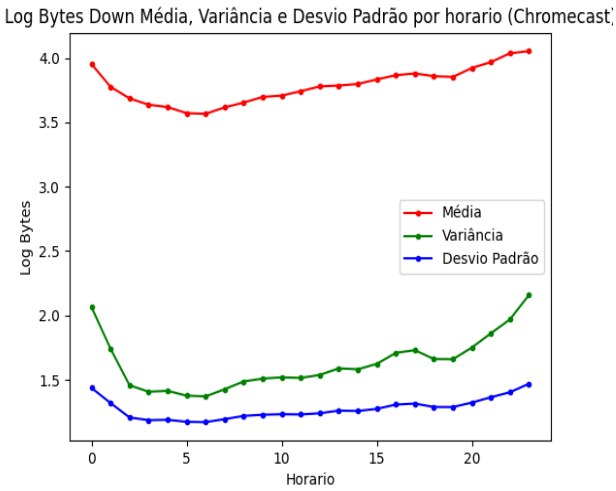


Figura 62

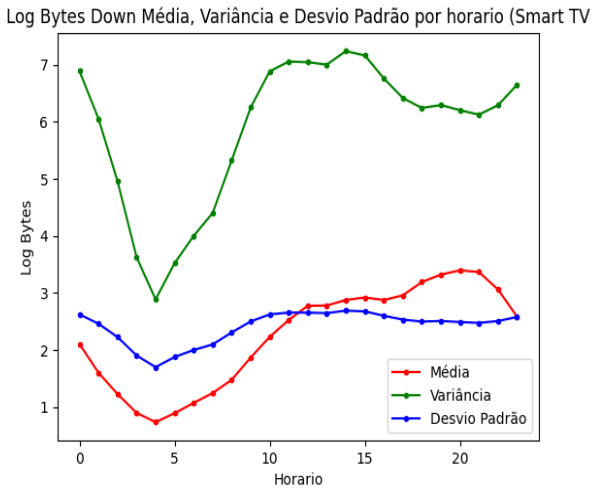


Figura 63

Apesar de ambos os dispositivos terem outliers, percebemos que o upload do Chromecast, entre às 22h e 23h, possui uma frequência de outliers na parte superior bem menor comparado aos outros casos. No que diz respeito aos downloads desse mesmo aparelho, nota-se que o outlier da parte inferior aconteceu no horário 23h.
Ainda sobre o Chromecast, podemos concluir que há um destaque maior de desvio padrão entre às 20h e 3h. Nesse caso, podemos inferir que isso ocorre devido à frequência de utilização do aparelho que pode ser menor nesse intervalo de tempo. Além disso, é perceptível que a Smart TV inclina-se para a tendência de funcionar de acordo com a utilização dos usuários, uma vez que, entre às 10h e às 20h, a média de download acende em igual proporção em relação a taxa de upload.

4 Caracterizando os horários com maior valor de tráfego

4.1 Filtragem dos Dados Necessários

Para esta Seção, foi necessário, novamente, utilizar o mesmo tipo de agrupamento utilizado na Seção acima. Contudo, dessa vez, selecionaremos o valor máximo da mediana e média de cada agrupamento. Em seguida, foram formados novos data_sets com os quais iremos trabalhar.

4.2 Histogramas

Os Histogramas foram contruídos conforme a teoria espcificada na **Seção 2.1** e estão organizados da seguinte forma: As figuras da esquerda representam o Chromecast e as figuras da direita representam a Smart TV.

4.2.1 Mediana Máxima do Log(Bytes Up)

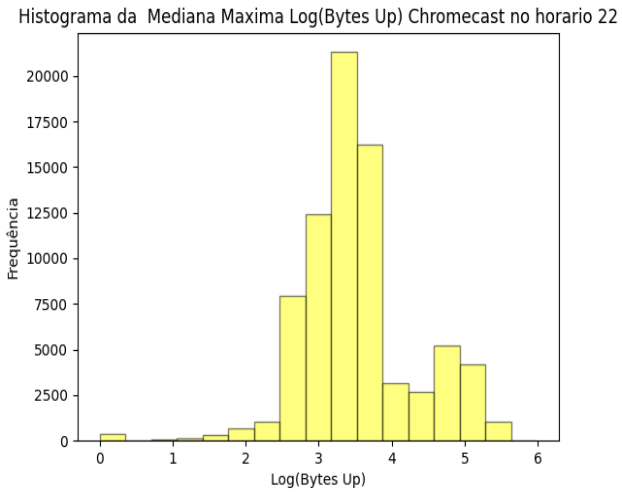


Figura 64

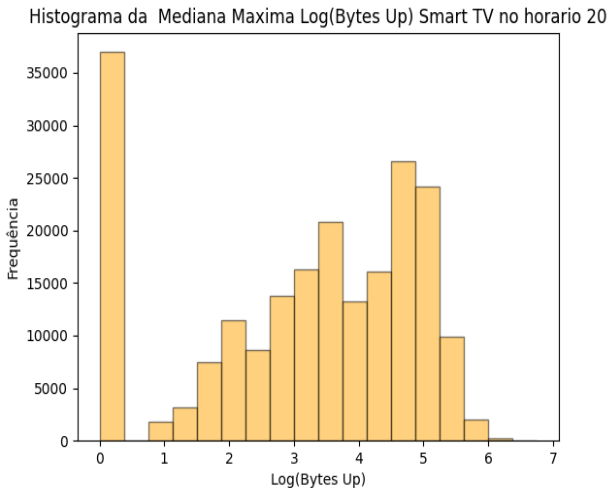


Figura 65

4.2.2 Média Máxima do Log(Bytes Up)

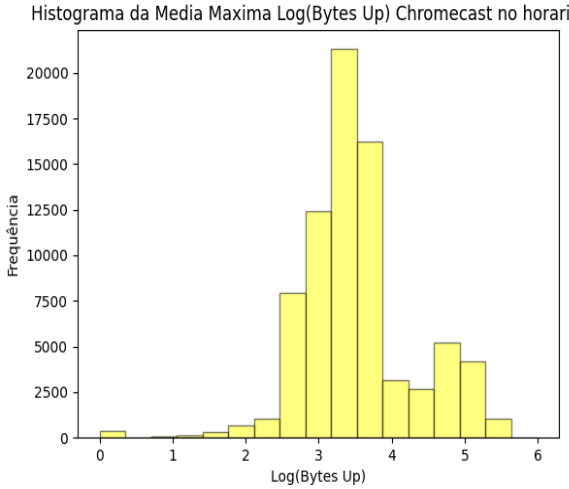


Figura 66

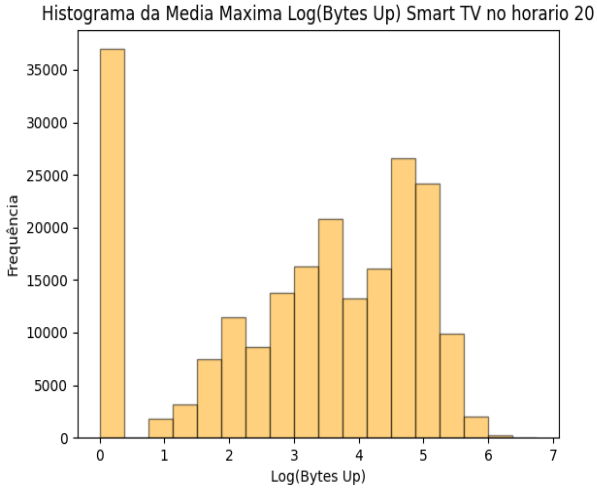


Figura 67

4.2.3 Mediana Máxima do Log(Bytes Down)

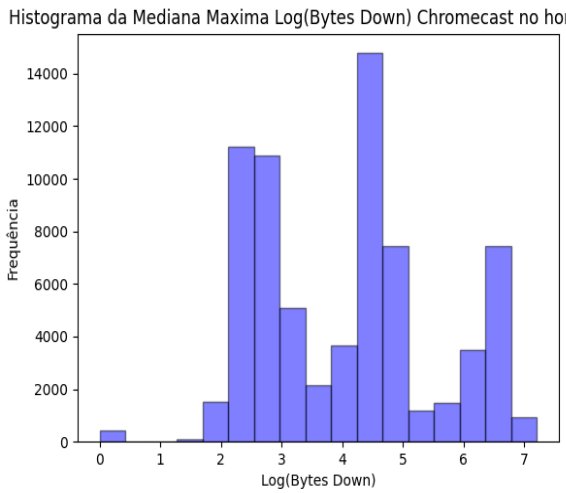


Figura 68

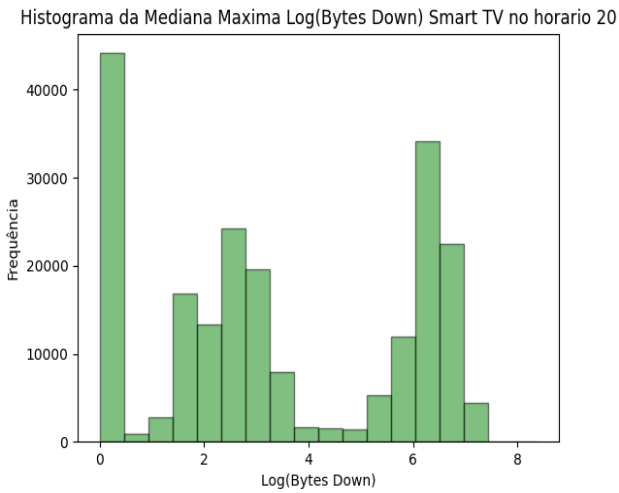


Figura 69

4.2.4 Média Máxima do Log(Bytes Down)

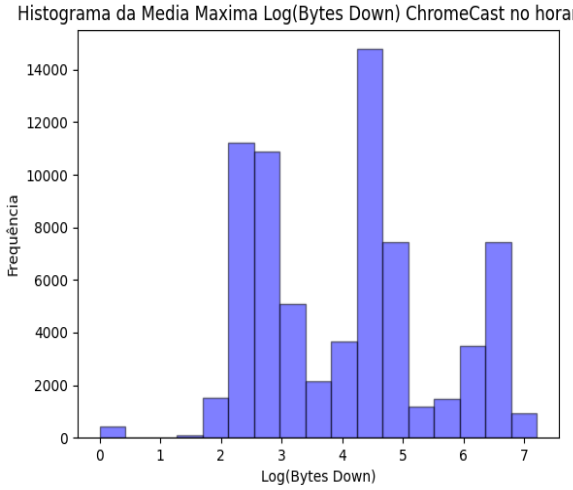


Figura 70

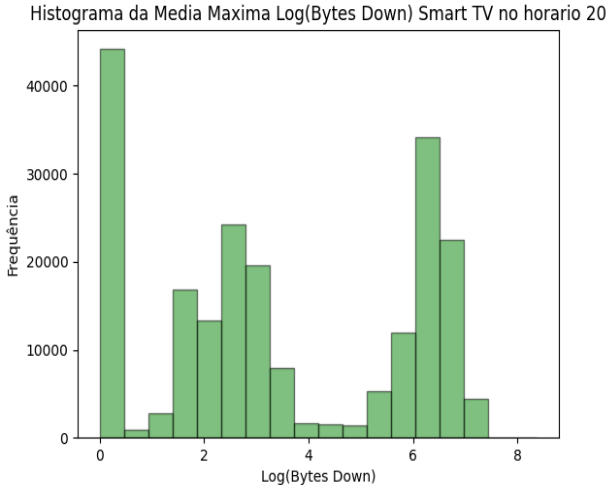


Figura 71

4.3 MLE Distribuição Gamma

Existem muitos métodos para estimar parâmetros desconhecidos a partir de dados. Aqui, nós vamos utilizar o Maximum Likelihood Estimate (MLE) que responde à seguinte pergunta: Para qual valor de parâmetro os dados observados têm a maior probabilidade? O MLE é um exemplo de estimativa pontual porque fornece um valor único para a incógnita Duas vantagens de utilizar o MLE é que muitas vezes é fácil de calcular e que, também, concorda com nossa intuição.
Para calcular o MLE, é necessário obter a função likelihood da distribuição, substituir os valores, derivar e igualar a 0. Fazendo isso, obtém-se as variáveis que maximizam a função. Contudo, utilizei, para este projeto, as funções prontas de bibliotecas da linguagem de programação Python para resolver essas equações.
Para esta distribuição, utilizei a biblioteca scipy.stats. A função utilizada retorna 3 valores: shape, loc e scale.
Loc refere-se à localização da distribuição, shape e scale são parametrôs próprios da distribuição Gamma.

4.3.1 Mediana Maxima Log(Bytes Up) Chromecast no horario 22

Tabela 3: Distribuição Gamma 1

Shape	Loc	Scale
3148.88	-39.809	0.0137606

4.3.2 Mediana Maxima Log(Bytes Down) Chromecast no horario 23

Tabela 4: Distribuição Gamma 2

Shape	Loc	Scale
27.1301	-3.63137	0.28323

4.3.3 Media Maxima Log(Bytes Up) Chromecast no horario 22

Tabela 5: Distribuição Gamma 3

Shape	Loc	Scale
3148.88	-39.809	0.0137606

4.3.4 Media Maxima Log(Bytes Down) ChromeCast no horario 23

Tabela 6: Distribuição Gamma 4

Shape	Loc	Scale
27.1301	-3.63137	0.28323

4.3.5 Mediana Maxima Log(Bytes Up) SmartTV no horario 20

Tabela 7: Distribuição Gamma 5

Shape	Loc	Scale
217.147	-23.8596	0.124245

4.3.6 Mediana Maxima Log(Bytes Down) SmartTV no horario 20

Tabela 8: Distribuição Gamma 6

Shape	Loc	Scale
896.547	-71.0622	0.0830499

4.3.7 Media Maxima Log(Bytes Up) SmartTV no horario 20

Tabela 9: Distribuição Gamma 7

Shape	Loc	Scale
217.147	-23.8596	0.124245

4.3.8 Media Maxima Log(Bytes Down) SmartTV no horario 20

Tabela 10: Distribuição Gamma 8

Shape	Loc	Scale
896.547	-71.0622	0.0830499

4.4 Distribuição Gaussiana

No que diz respeito a esta distribuição, apenas se fez necessário reutilizar os cálculos da gaussiana e da média feitos anteriormente como argumentos da função. Ou seja, não houve a urgência de utilizar métodos prontos.

4.4.1 Mediana Maxima Log(Bytes Up) Chromecast no horario 22

Tabela 11: Distribuição Gaussiana 1

Média	Mediana
3.52155	3.4438

4.4.2 Mediana Maxima Log(Bytes Down) Chromecast no horario 23

Tabela 12: Distribuição Gaussiana 2

Média	Mediana
4.0527	4.28566

4.4.3 Media Maxima Log(Bytes Up) Chromecast no horario 22

Tabela 13: Distribuição Gaussiana 3

Média	Mediana
3.52155	3.4438

4.4.4 Media Maxima Log(Bytes Down) ChromeCast no horario 23

Tabela 14: Distribuição Gaussiana 4

Média	Mediana
4.0527	4.28566

4.4.5 Mediana Maxima Log(Bytes Up) SmartTV no horario 20

Tabela 15: Distribuição Gaussiana 5

Média	Mediana
3.12426	3.53052

4.4.6 Mediana Maxima Log(Bytes Down) SmartTV no horario 20

Tabela 16: Distribuição Gaussiana 6

Média	Mediana
3.39609	2.88961

4.4.7 Media Maxima Log(Bytes Up) SmartTV no horario 20

Tabela 17: Distribuição Gaussiana 7

Média	Mediana
3.12426	3.53052

4.4.8 Media Maxima Log(Bytes Down) SmartTV no horario 20

Tabela 18: Distribuição Gaussiana 8

Média	Mediana
3.39609	2.88961

4.5 Histograma Com MLE

Para construir o histograma com MLE, utilizei o argumento density do método plt.hist() da biblioteca matplotlib, visando padronizar o histograma com as distribuições acima.

4.5.1 Mediana Máxima e Média Máxima Log(Bytes Up) Chromecast

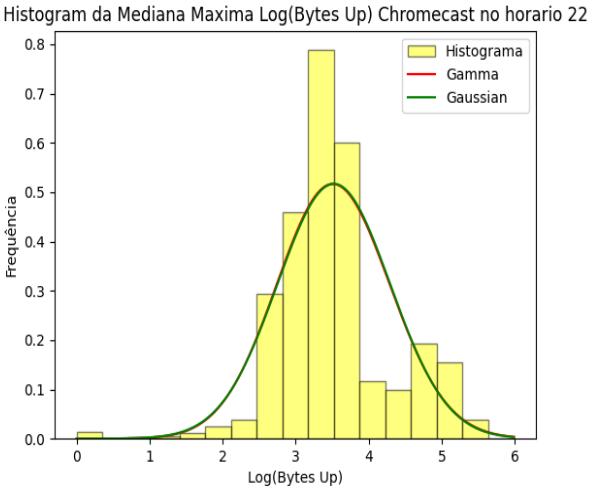


Figura 72

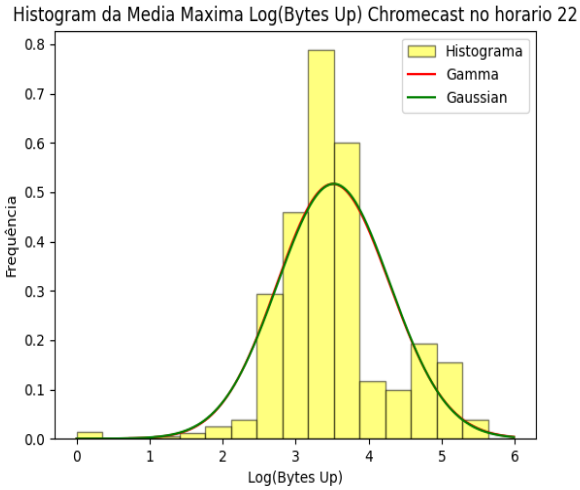


Figura 73

4.5.2 Mediana Máxima e Média Máxima Log(Bytes Down) Chromecast

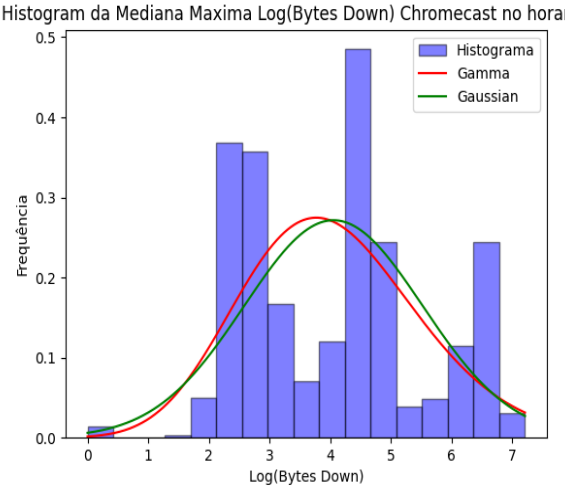


Figura 74

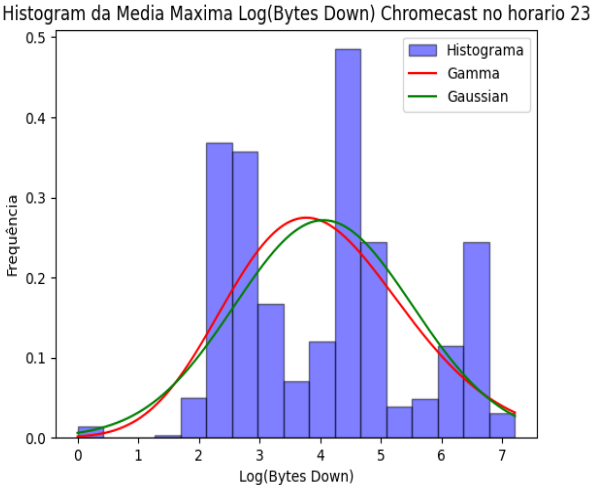


Figura 75

4.5.3 Mediana Máxima e Média Máxima Log(Bytes Up) Smart TV

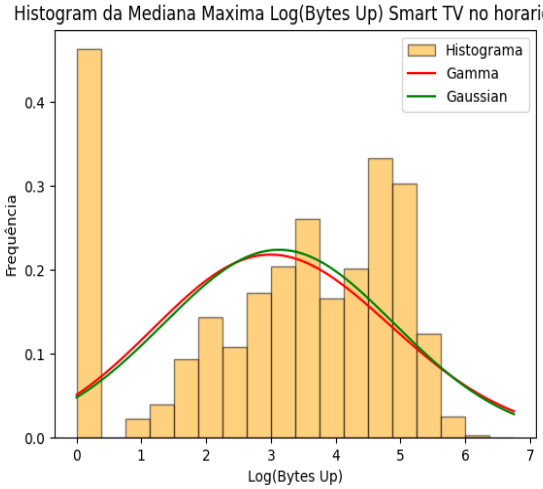


Figura 76

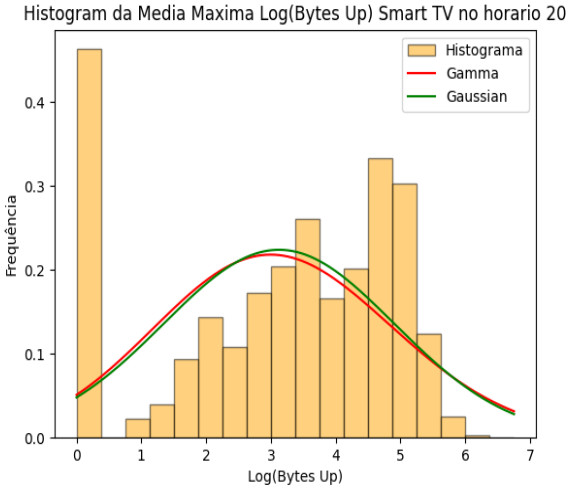


Figura 77

4.5.4 Mediana Máxima e Média Máxima Log(Bytes Down) Smart TV

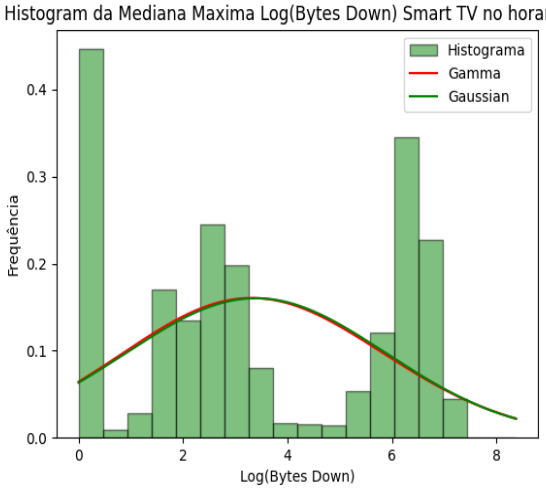


Figura 78

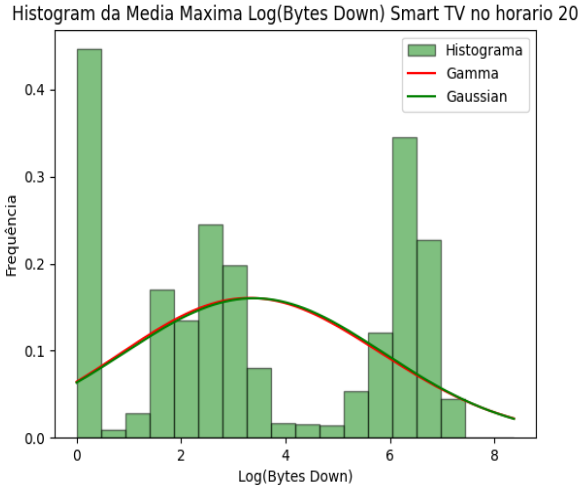


Figura 79

4.6 Gráfico Probability Plot

Para construir o Probability Plot, foi-se necessário chamar o método probplot() em que os argumentos eram o MLE da Distribuição Gamma e o MLE da Distribuição Gaussiana

4.6.1 Chromecast

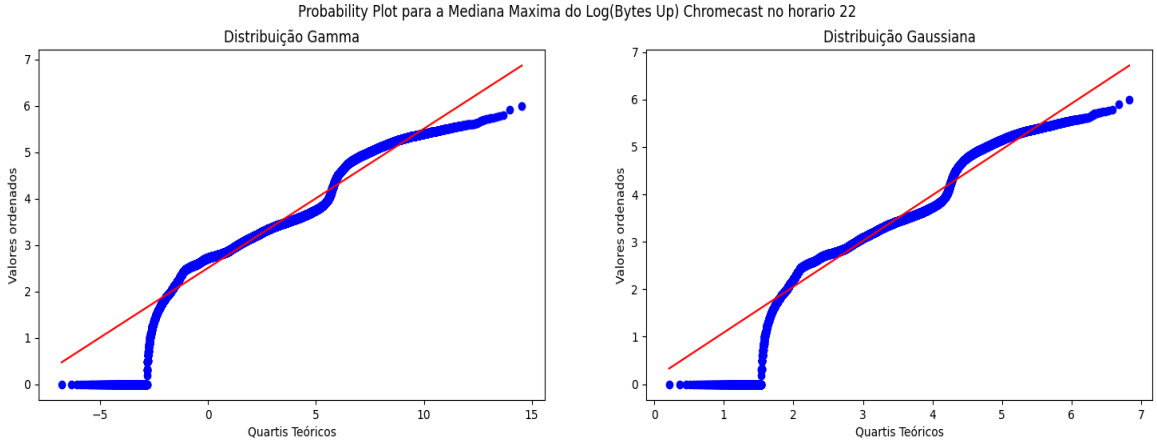


Figura 80

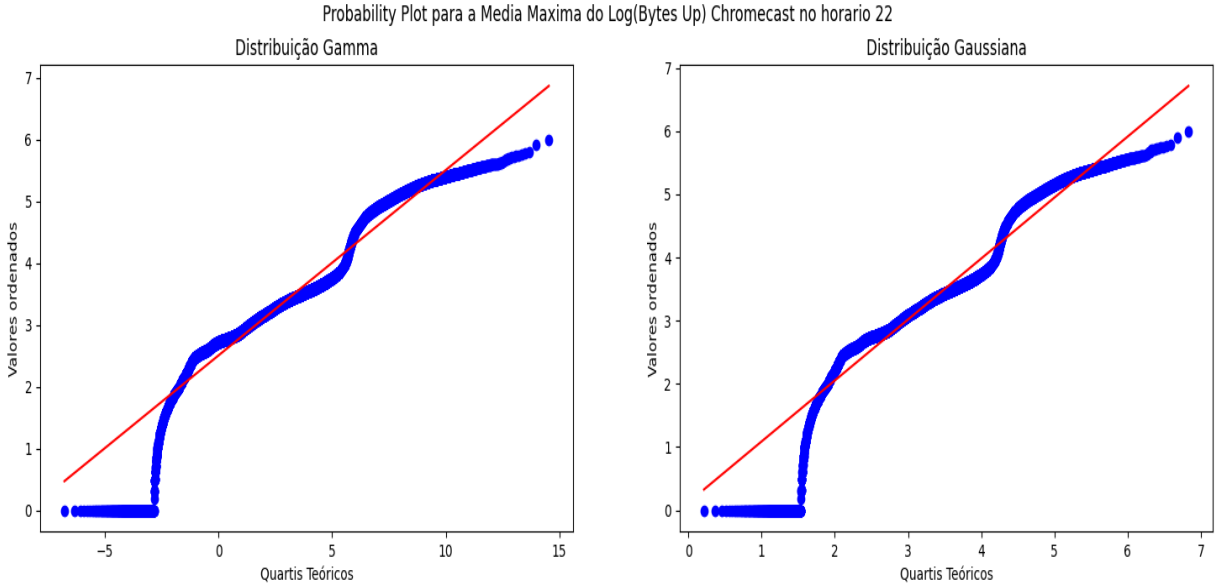


Figura 81

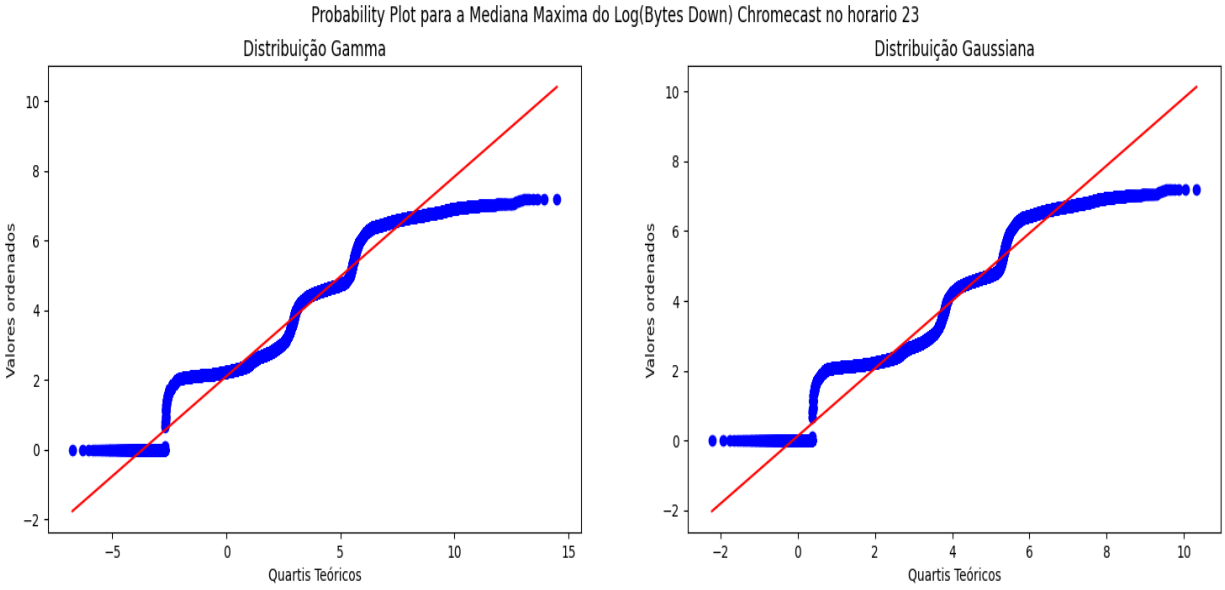


Figura 82

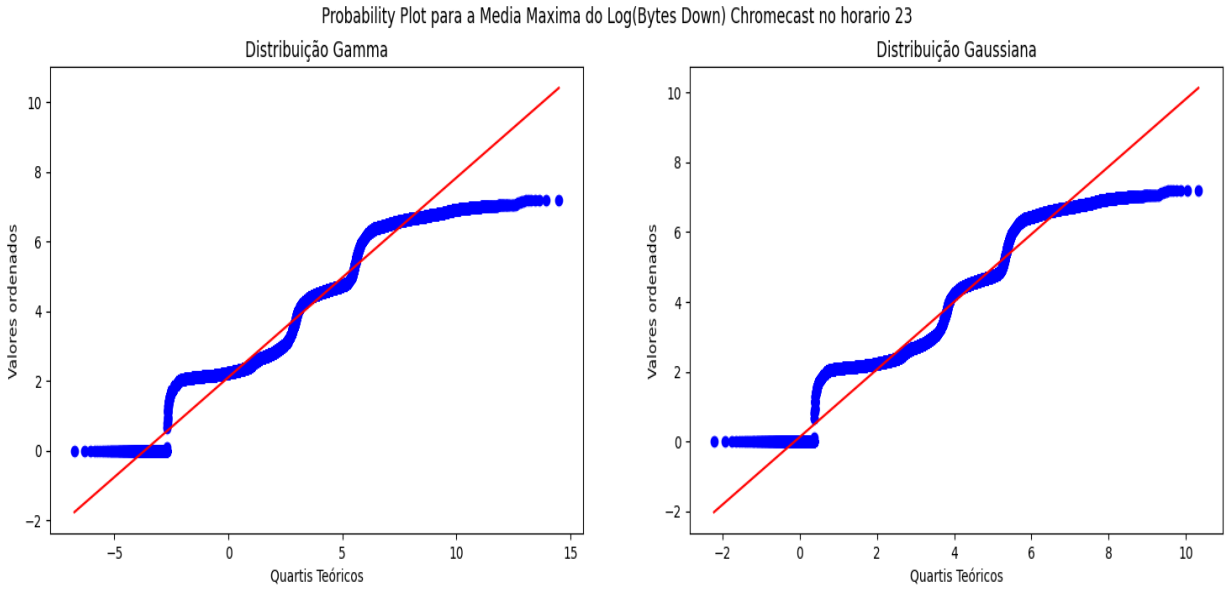


Figura 83

4.6.2 Smart TV

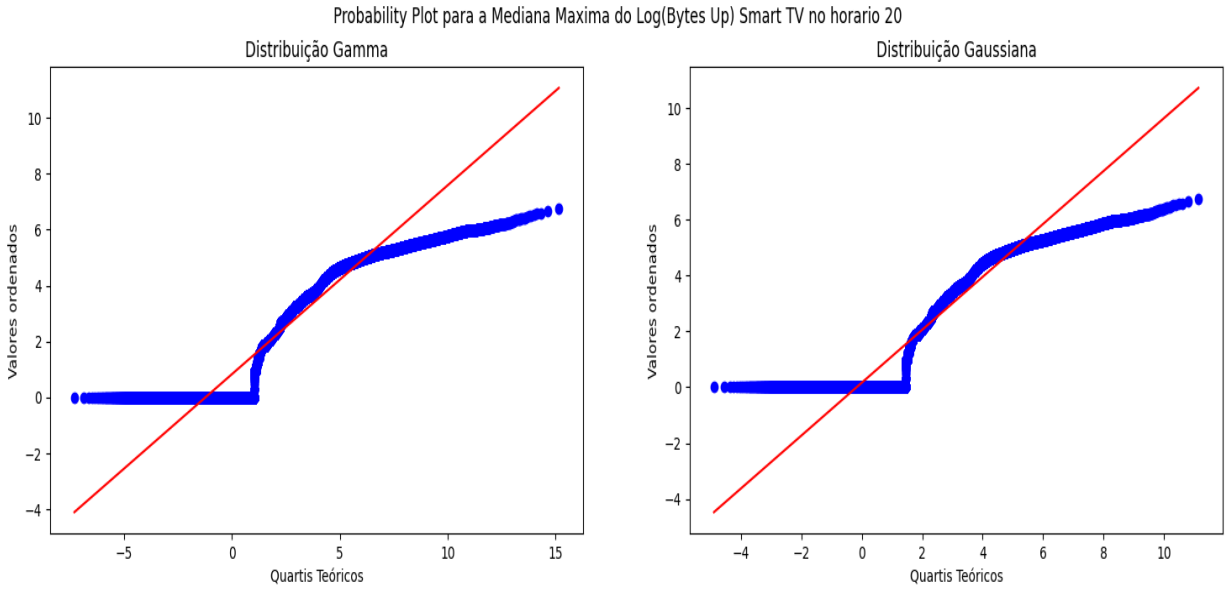


Figura 84

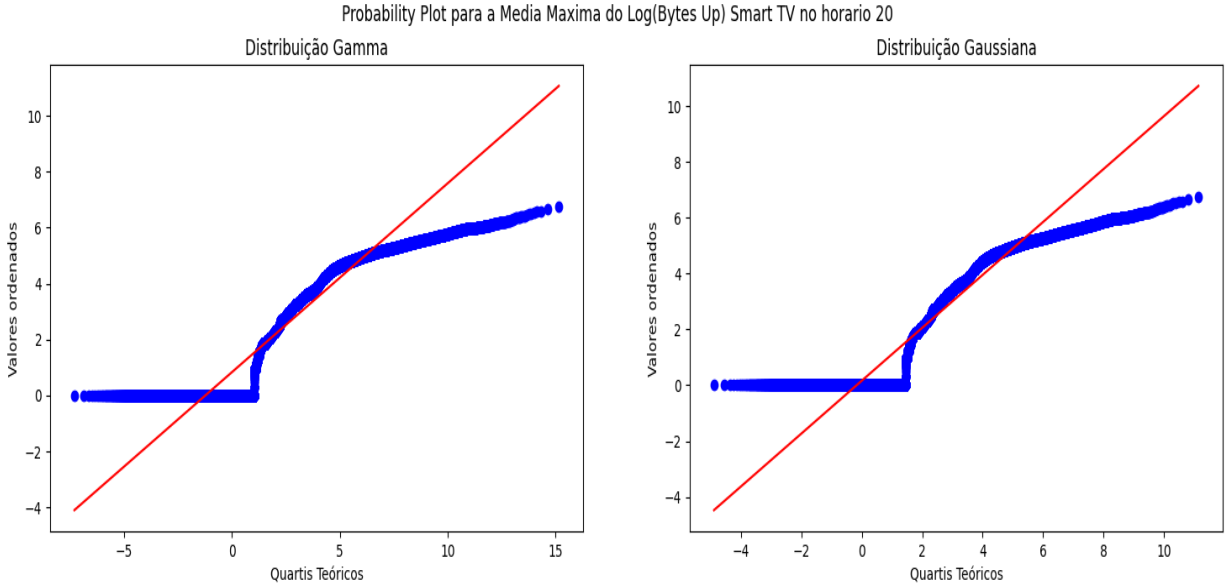


Figura 85

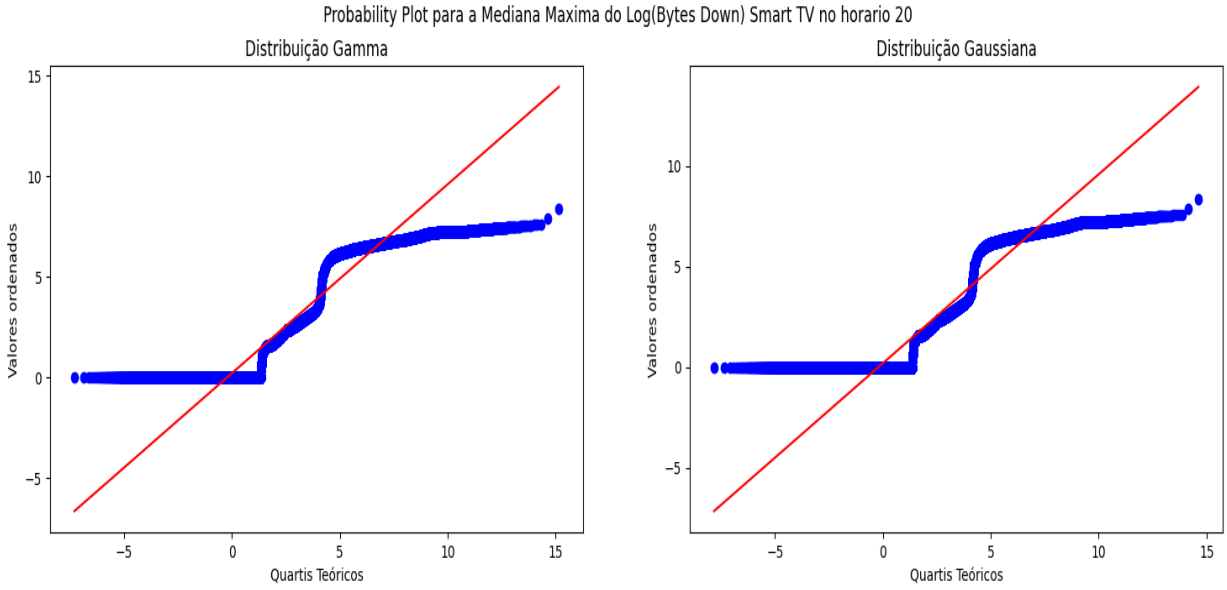


Figura 86

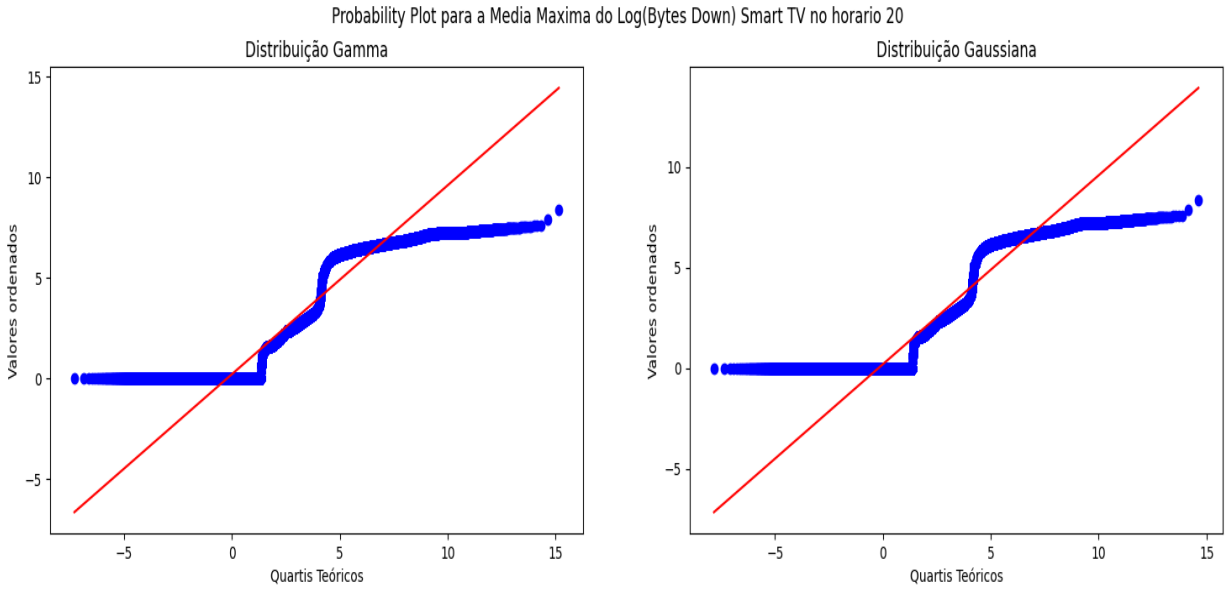


Figura 87

4.7 Análise

Dado as estatísticas acima, podemos perceber que o chromecast possui uma regularidade mais forte em quantidade de bytes baixados quando comparamos com a regularidade do upload. Por outro lado a Smart TV continua com o 0 bem nítido, ressaltando que muitos dispositivos não estão realizando download de dados ainda que esteja no horário com maior mediana.Também é possível notar que os horários de pico de ambas as taxas funcionam de forma bem distinta.

Além disso, podemos perceber que as taxas de download variam entre taxas pequenas e taxas relativamente grandes. Acredito que isso deve ocorrer devido a qualidade de internet de cada cliente que esteja utilizando os serviços e, essa taxa, será configurada de acordo com a estabilidade da banda.
Entretanto, a taxa de upload permanece constante até certo ponto e depois só varia de forma crescente.
Todos os 8 datasets possuem comportamentos bem similares, uma vez que a média e a mediana têm a mesma configuração gráfica para o Chromecast e para a Smart TV. Ou seja, não foi possível observar alguma diferença entre eles, filtrando pelo horário.
Nesse sentido, podemos concluir que há uma possibilidade de inferir que o dataset de upload, para a maior média do chromecast, pode ser mapeado tanto por uma Gaussiana quanto por uma Distribuição Gamma. Todavia, infelizmente, não podemos falar o mesmo para os datasets de download.
Sobre a Smart Tv, podemos concluir que esse dispositivo não apresentou um bom resultado. Isso pode ter acontecido devido à alta frequência de zeros supracitados.
Quando analisamos os gráficos de Probability Plot, reforçamos a ideia de que a taxa de download do chromecast pode seguir, ainda que com algumas perdas, distribuição com variáveis aleatórias da literatura.

5 Análise da correlação entre as taxas de upload e download para os horários com o maior valor de tráfego

5.1 Coeficiente de Correlação de Amostragem

Para esta Seção, fez-se necessário a utilização do Coeficiente de Pearson para a obtenção dos valores estatísticos requeridos neste tópico.
Calcula-se o coeficiente de correlação de Pearson segundo a seguinte fórmula:

$$p = \frac{Cov(X,Y)}{\sqrt{Var(X) \times Var(Y)}}$$

onde,

$$var(Y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n},$$

$$var(Y) = \frac{\sum_{i=1}^{n} (y_i - \bar{y})^2}{n},$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n},$$

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n},$$

$$Cov(X,Y) = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{n}$$

e $x_1, x_2, ..., x_n$ e $y_1, y_2, ..., y_n$ são os valores medidos para ambas as variáveis.
A correlação pode ser obtida em um intervalo [-1,1] em que 1 ilustra uma forte correlação positiva, -1 significa uma forte correlação negativa e, por fim, 0 indica que não há correlação.

5.1.1 Coeficiente de Correlação de Pearson para a Mediana Máxima Log(Bytes) Chromecast

Tabela 19: Coeficiente de Correlação de Pearson 1

Coeficiente de Pearson	p_valor
0.792504	0

5.1.2 Coeficiente de Correlação de Pearson para a Média Máxima Log(Bytes) Chromecast

Tabela 20: Coeficiente de Correlação de Pearson 2

Coeficiente de Pearson	p_valor
0.792504	0

5.1.3 Coeficiente de Correlação de Pearson para a Mediana Máxima Log(Bytes) Smart TV

Tabela 21: Coeficiente de Correlação de Pearson 3

Coeficiente de Pearson	p_valor
0.915609	0

5.1.4 Coeficiente de Correlação de Pearson para a Média Máxima Log(Bytes) Smart TV

Tabela 22: Coeficiente de Correlação de Pearson 4

Coeficiente de Pearson	p_valor
0.915609	0

Importante: As tabelas ilustradas acima têm seu respectivo p_valor = 0. Ou seja, isso remete, de acordo com a teoria, um grau de significância elevado.

5.2 Gráfico dos Coeficientes de Correlação de Amostragem

Para a ilustração deste gráfico, foi utilizada a função scatter() da biblioteca matplotlib com as tabelas exigidas como argumentos da função.

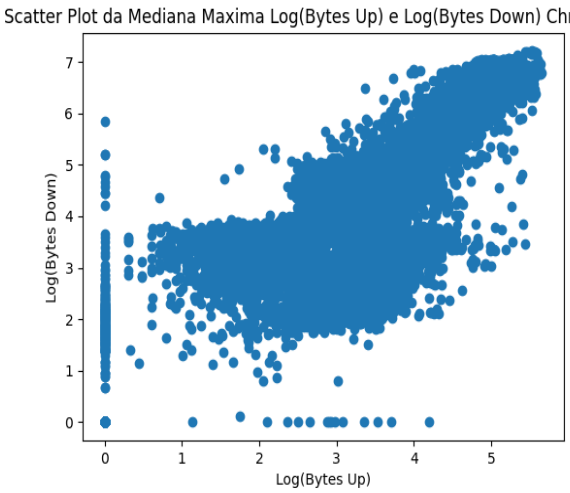


Figura 88

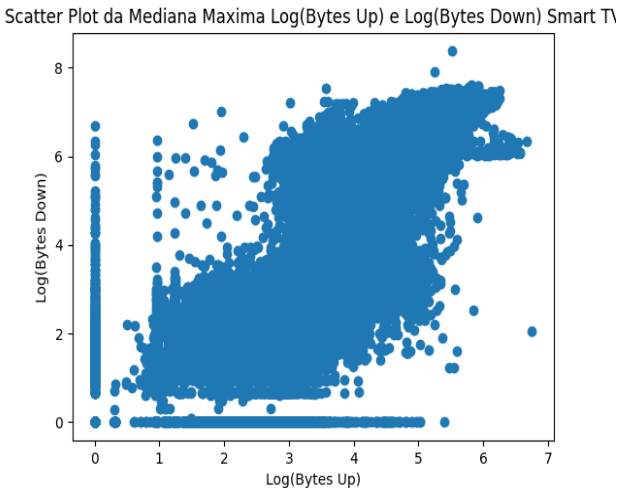


Figura 89

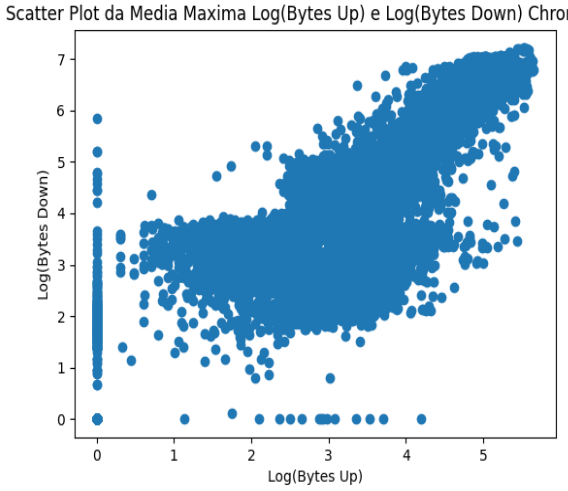


Figura 90

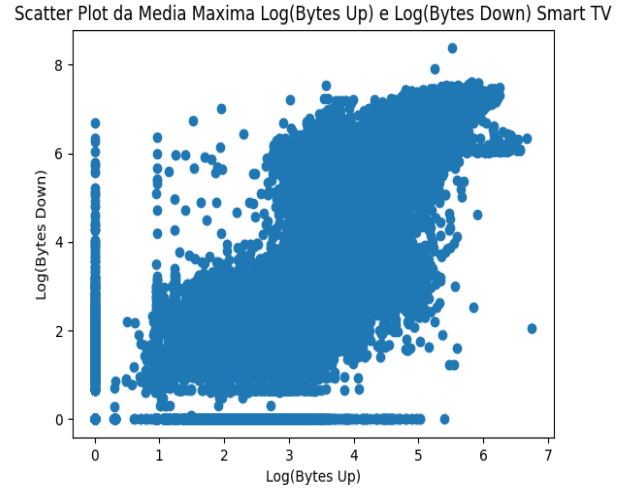


Figura 91

5.3 Análise

Nesse sentido, podemos notar que os dados têm uma correlação para ambos os dispositivos, ainda que o coeficiente de Pearson do Chromecast seja menor em relação à taxa dos outros dispositivos Smart TV.

6 Comparação dos dados gerados pelos dispositivos SmartTV e Chromecast

6.1 G-teste

Para construir este teste, fez-se necessário, antes de tudo, separar os dados em segmentos de bins iguais para que fosse possível realizar uma comparação. Sendo assim, com esses ajustes

feitos, usei a função power_divergence() para obter o G-test e o p_valor.
A fórmula geral para o G é:

$$G = 2 \sum_{i=1}^{n} O_i ln(\frac{O_i}{E_i}),$$

onde O_i são os valores observados e E_i são os valores esperados.

6.2 Resultados

Tabela 23: G-test

G_test	p_valor	Nome	Coluna
1.74065	0.999996	smart_tv_mediana_up_maxima_chromecast_mediana_up_maxima	log_bytes_up
1.74065	0.999996	smart_tv_media_up_maxima_chromecast_media_up_maxima	log_bytes_up
2.35922	0.999967	smart_tv_mediana_down_maxima_chromecast_mediana_down_maxima	log_bytes_down
2.35922	0.999967	smart_tv_media_down_maxima_chromecast_media_down_maxima	log_bytes_down

6.3 Análise

De acordo com a teoria, quando o p_valor do teste está próximo de 1, podemos afirmar que esse teste não tem uma grande significância. Logo, nada podemos inferir sobre esses dados.