UFRJ

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Lucas Garcia Santiago de Abreu
DRE – 121039536

Trabalho Final da Disciplina
Estatística e Modelos Probabilísticos

Rio de Janeiro

Sumário

1- Os dados

1.1- Os datasets

Os dois datasets utilizados foram fornecidos pela professoram e consistem em dados de tráfego -velocidade de *download* e *upload*- de dispositivos -Chromecasts e Smart Tvs-, realizados de minuto a minuto.

1.2- Tratamento dos dados

Para manipulação dos datasets, a biblioteca utilizada foi a 'pandas', que armazena arquivos tabulares em estruturas chamadas 'dataframes' e possui diversas funções para manipulações dos dados.

Ao iniciar o trabalho, se fez necessário tratar os dados de alguma forma, visto que foi sugerido o uso dos dados log10.

Para isso, foi adicionado uma constante (1) a todos os valores antes deles serem convertidos a log10. Como resultado, evitam-se divisões por zero nas conversões de base.

As conversões para log10 foram realizadas utilizando a biblioteca 'numpy', com a função 'log10'.

Outro tratamento realizado foi a adição de uma coluna referente às horas, dado que posteriormente seria necessário agrupar os dados por horário.

2- Estatísticas Gerais

2.1- Histograma

Os histogramas abaixo demonstram a distribuição das frequências dos valores das velocidades.

A quantidade de barras foi determinada a partir do método de Sturges, dado que 'n' é a quantidade total de dados:

$$1 + 3.3322\, log(n)$$

Fórmula 1 – Método de Sturges

E os gráficos foram feitos utilizando a biblioteca 'matplotlib'

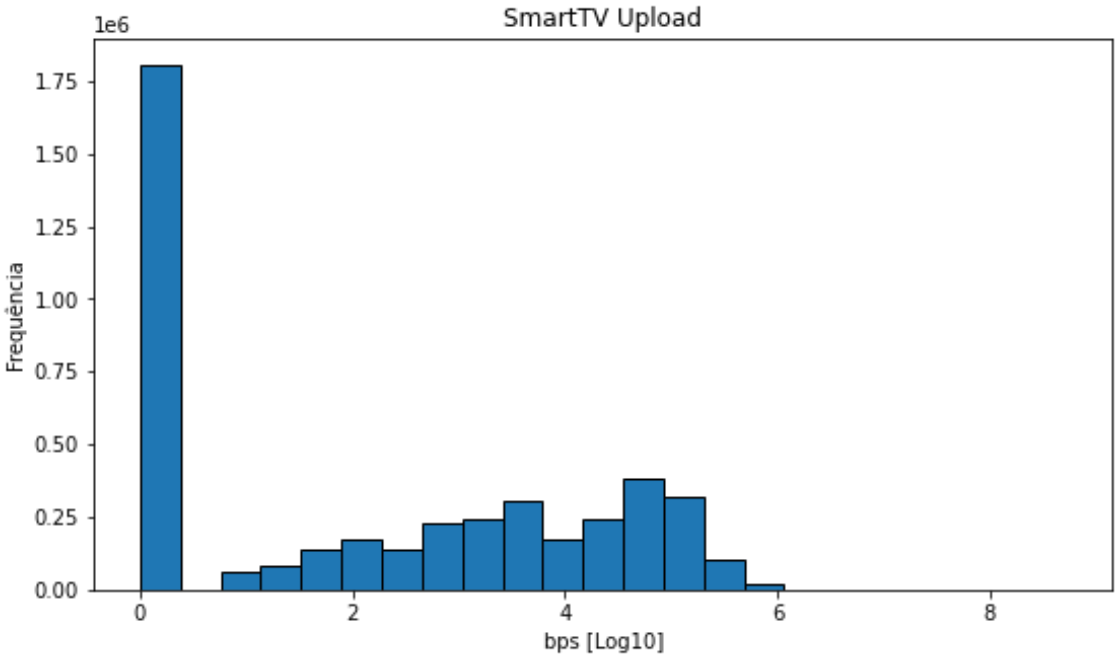

Figura 1 – Histograma das velocidades de upload da SmartTv

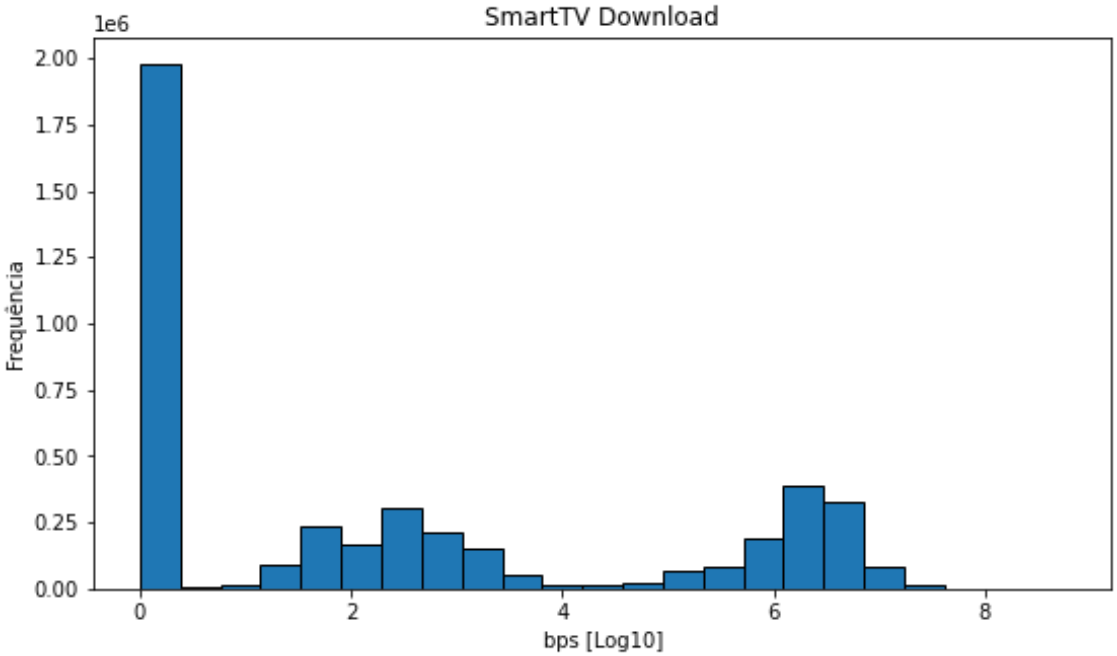

Figura 2 – Histograma das velocidades de Download da SmartTv

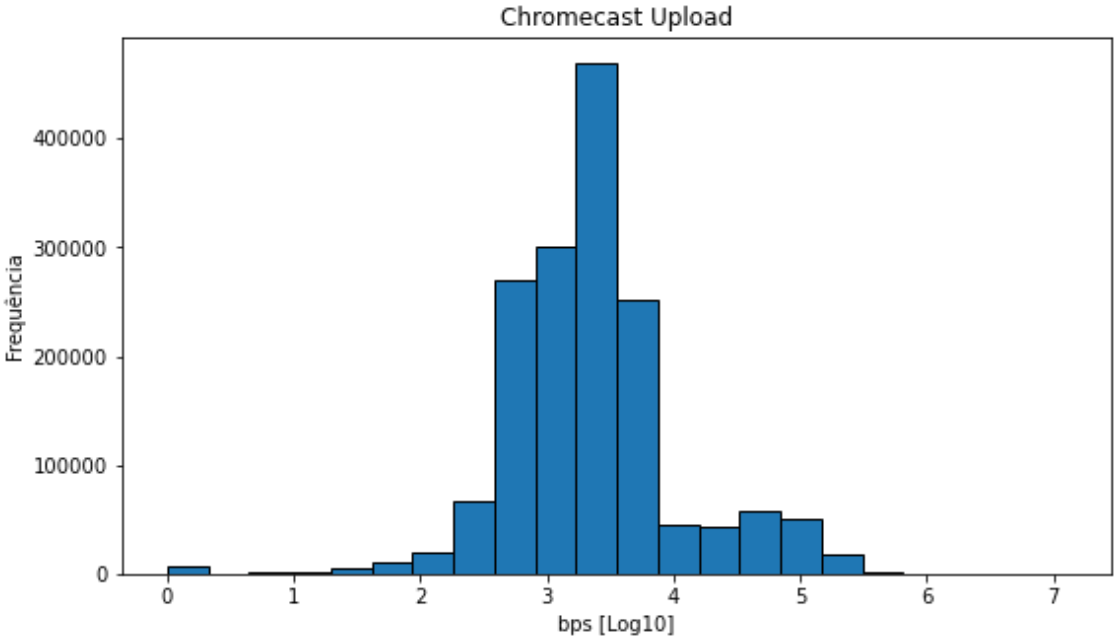

Figura 3 – Histograma das velocidades de upload do Chromecast

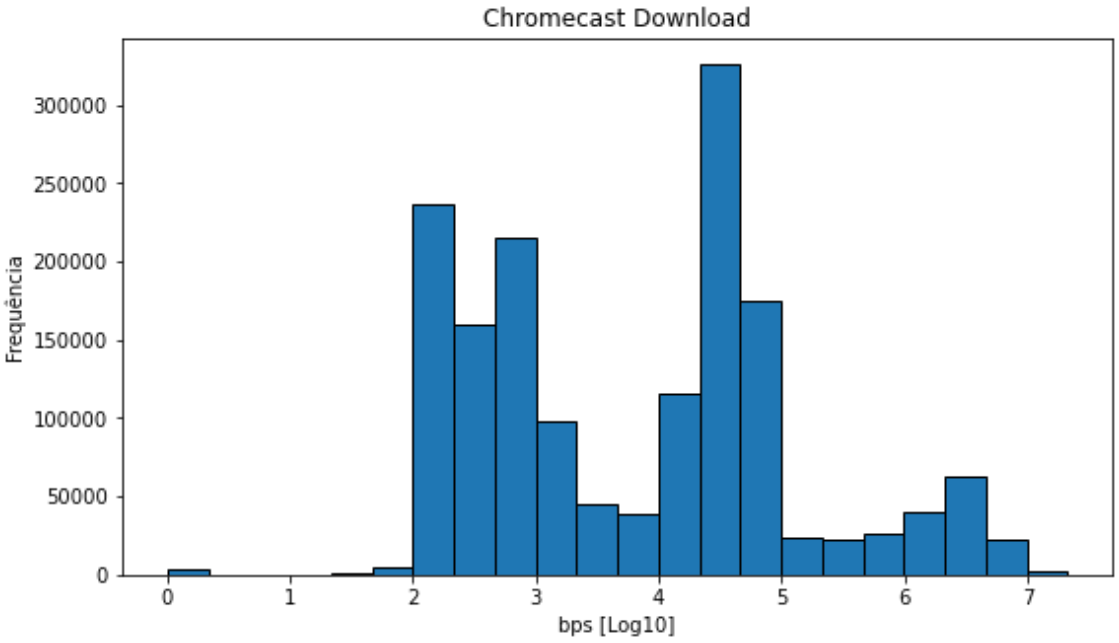

Figura 4 – Histograma das velocidades de download do Chromecast

2.2 – Função de Distribuição Empírica

Dado que a função de distribuição empírica é definida por:

f(x) = quantidade de observações menores que x/total de observações

Foi utilizado a função ‘linspace’ do ‘numpy’, que cria um ‘array’ com valores espaçados dentro de um intervalo definido, que no caso foi um intervalo de 0 a 1, sendo espaçado pelo tamanho do dataset.

Cruzando o resultado dessa função linspace com uma lista ordenada dos valores de cada ‘dataset’, foram obtidas as seguintes funções:

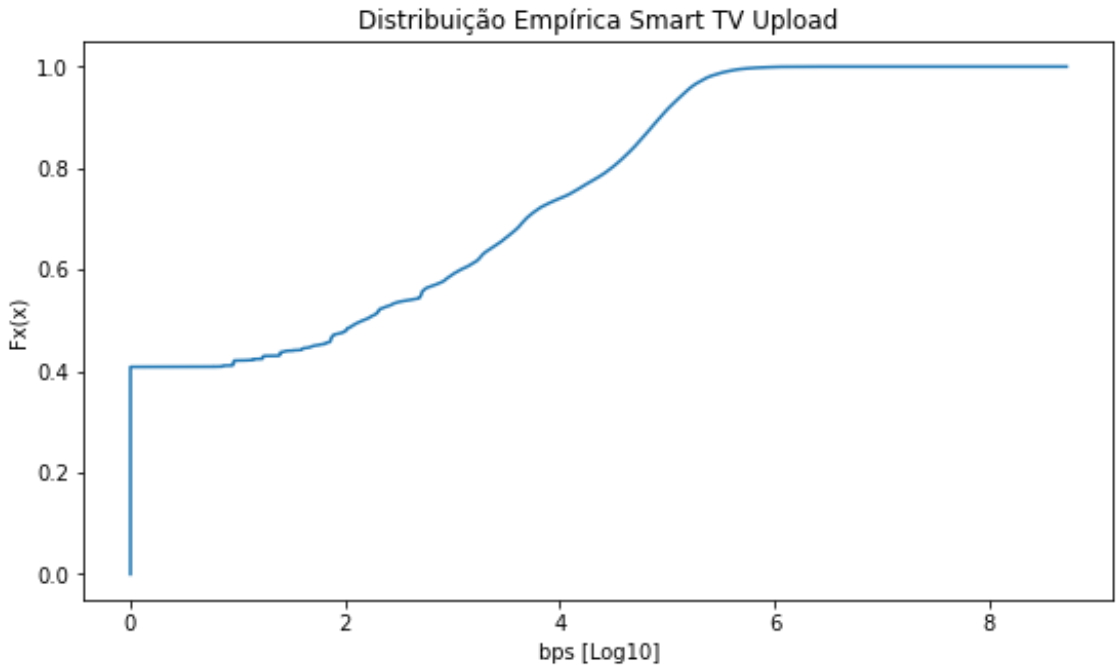

Figura 5 – Função de distribuição empírica das velocidades de upload da Smart Tv

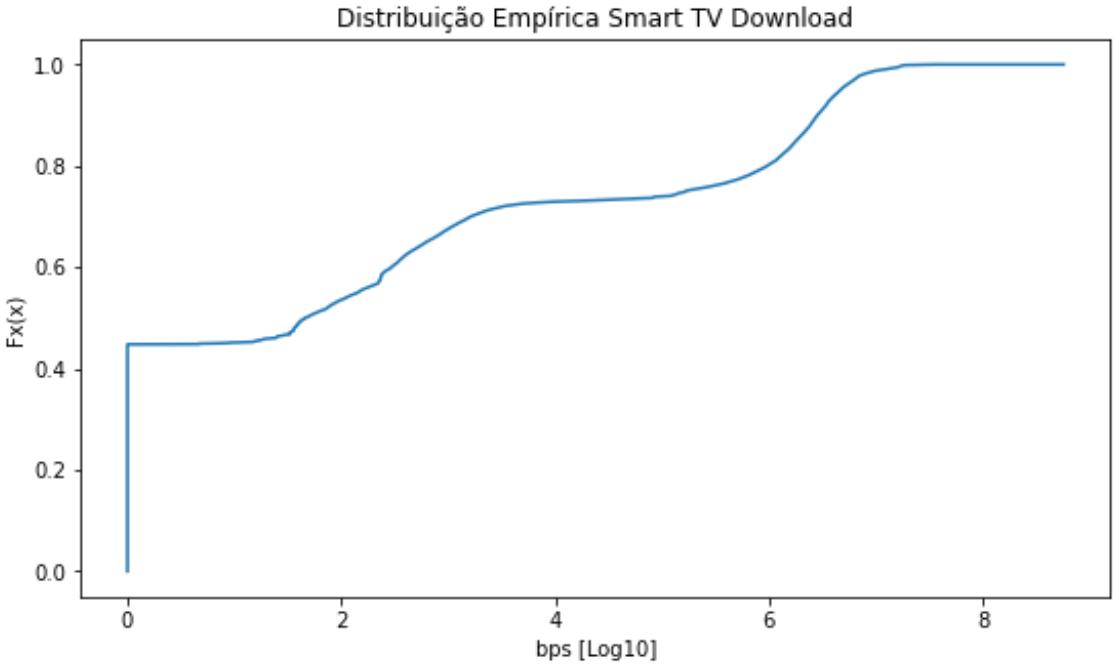

Figura 6 – Função de distribuição empírica das velocidades de download da Smart Tv

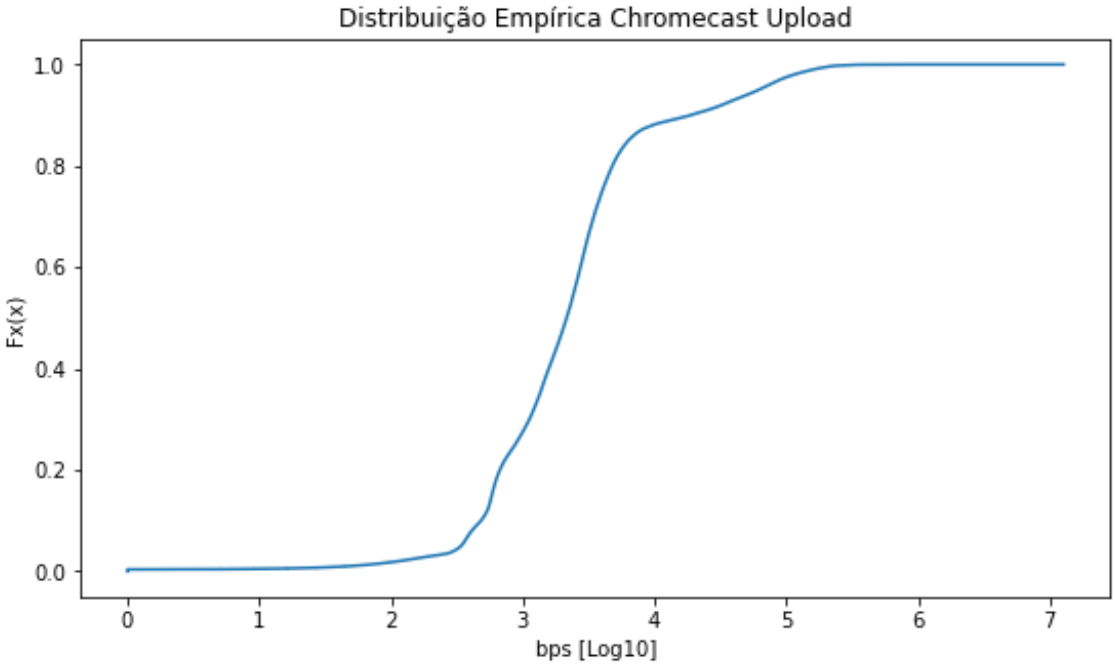

Figura 7 – Função de distribuição empírica das velocidades de upload do Chromecast

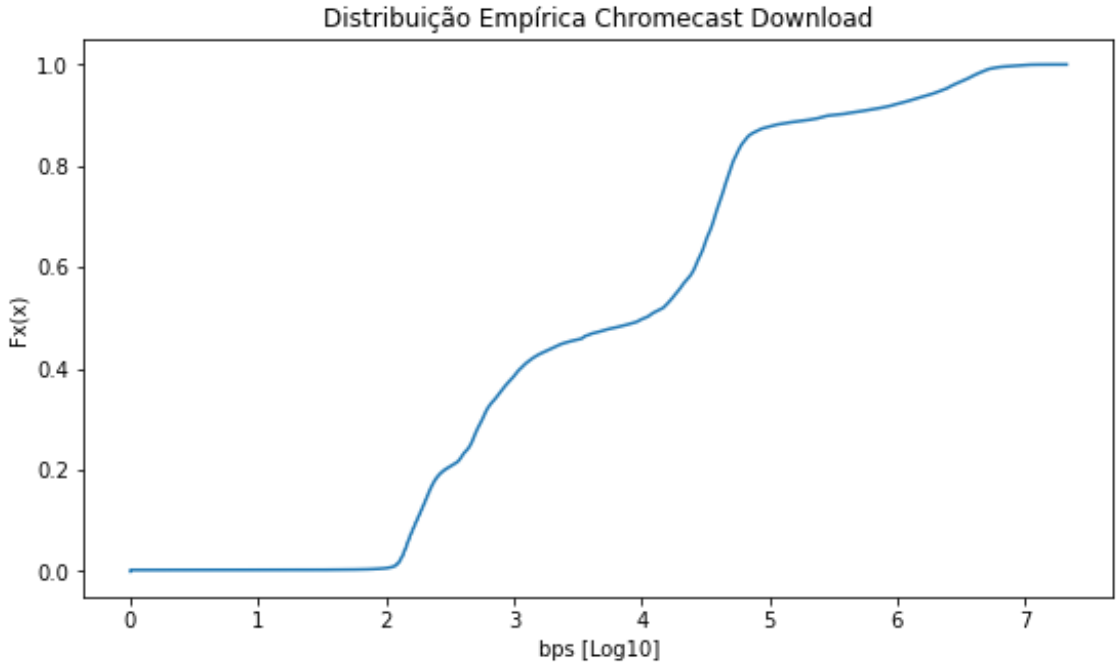

Figura 8 – Função de distribuição empírica das velocidades de download do Chromecast

2.3 – Boxplot

Os 'boxplots' foram obtidos através da função de 'boxplot' do próprio matplot lib. A função utiliza a distribuição dos dados e as medianas para montar o gráfico.

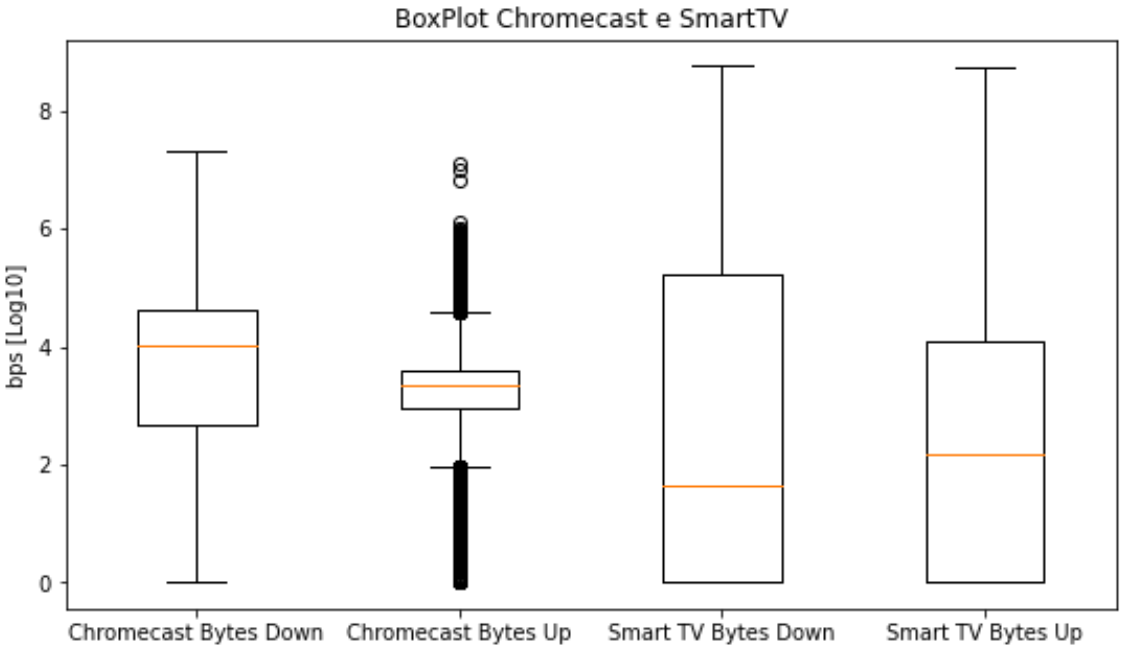

Figura 9 – Boxplots Download e Upload, Chromecast e Smart Tv

2.4 – Média, Variância e Desvio Padrão

Utilizando as funções do ‘pandas’, os valores de média, variância e desvio padrão foram calculados para cada um dos datasets e tipo de tráfego.

Estatísticas Chromecast

Download

Média: 3.8000457060383566

Desvio Padrão: 1.2899207725150774

Variância: 1.6638955993658944

Upload

Média: 3.3502996618095193

Desvio Padrão: 0.6782098836639917

Variância: 0.45996864629952516

Estatísticas Smart TV

Download

Média: 2.351678620482843

Desvio Padrão: 2.5925516190334212

Variância: 6.721323897352814

Upload

Média: 2.1582882065066804

Desvio Padrão: 2.0273478597975836

Variância: 4.110139344625843

2.5 – Resultados

No histograma das Smart Tvs pode-se observar um comportamento muito característico onde o número de dados na faixa do zero é extremamente grande, isso pode representar um comportamento do dispositivo de realizar medições mesmo quando não há tráfego ou algum outro comportamento do tipo.

O mesmo comportamento também pode ser observado nos gráficos de distribuição empírica, onde a quantidade de zeros altera o formato da distribuição. Já no caso dos Chromecasts, é possível observar um aumento considerável a partir do 3 – lembrando que os dados estão em log10 – algo que também pode ser observado nos histogramas.

Observando os boxplots é possível notar que a mediana das taxas de download e upload do mesmo dispositivo é relativamente parecida, entretanto suas distribuições apresentam diferenças. Principalmente nas taxas de upload do Chromecast que pode estar significando um

comportamento anômalo do dispositivo exigindo muito tráfego mais tráfego em alguns casos que em outros.

Quanto as outras estatísticas, o comportamento explicado anteriormente ainda pode ser visto onde as taxas de um mesmo dispositivo tem valores parecidos, mas comportamentos diferentes entre si. Talvez essa diferença seja suficiente para o provedor detectar qual tipo de aparelho é utilizado em sua rede, principalmente observando os valores de variância que são bem distintos entre os dispositivos.

3- Estatísticas por Horário

3.1- Divisão dos horários

A obtenção das estatísticas por horário foi realizada diretamente na criação dos gráficos, tanto os boxplots quanto média, variância e desvio padrão. Para cada valor na coluna de hora, os dados foram agrupados e seus gráficos construídos.

3.2- Boxplots

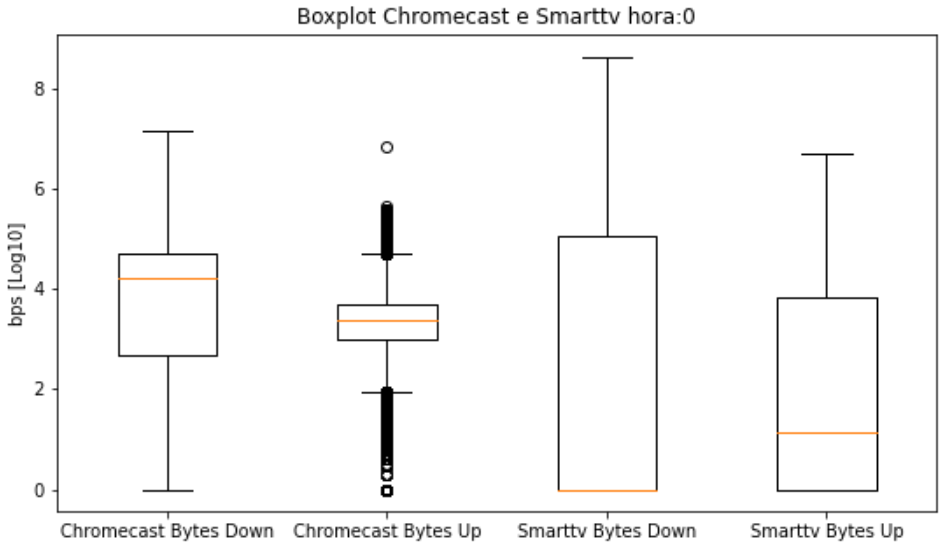

Figura 10 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 0

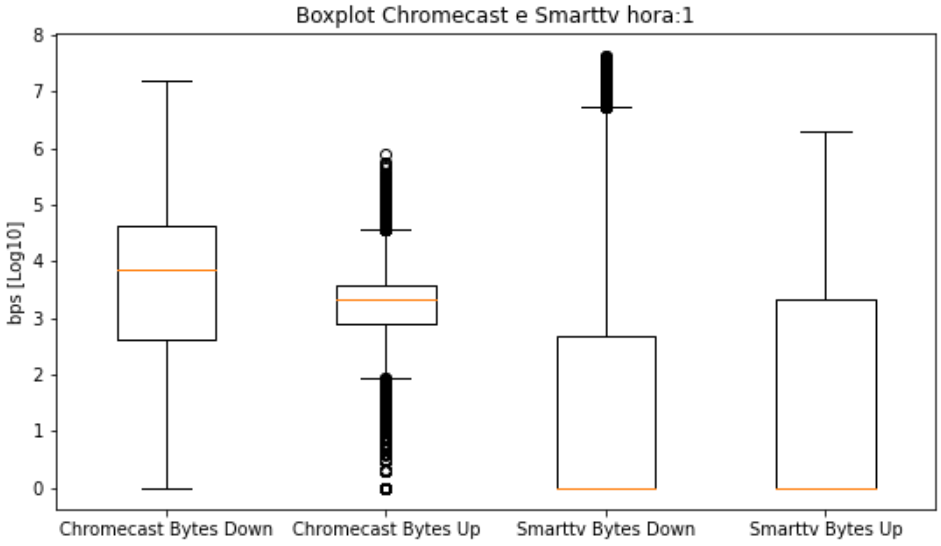

Figura 10 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 1

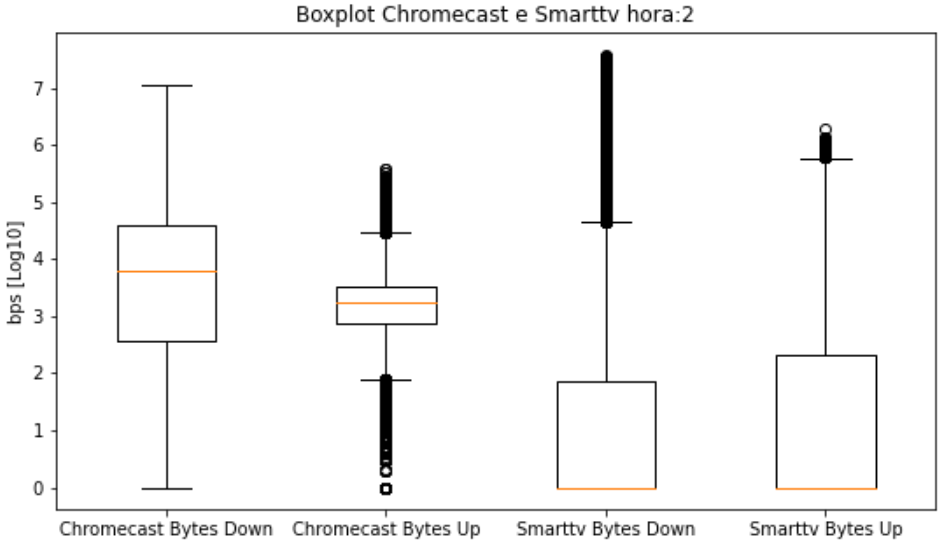

Figura 11 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 2

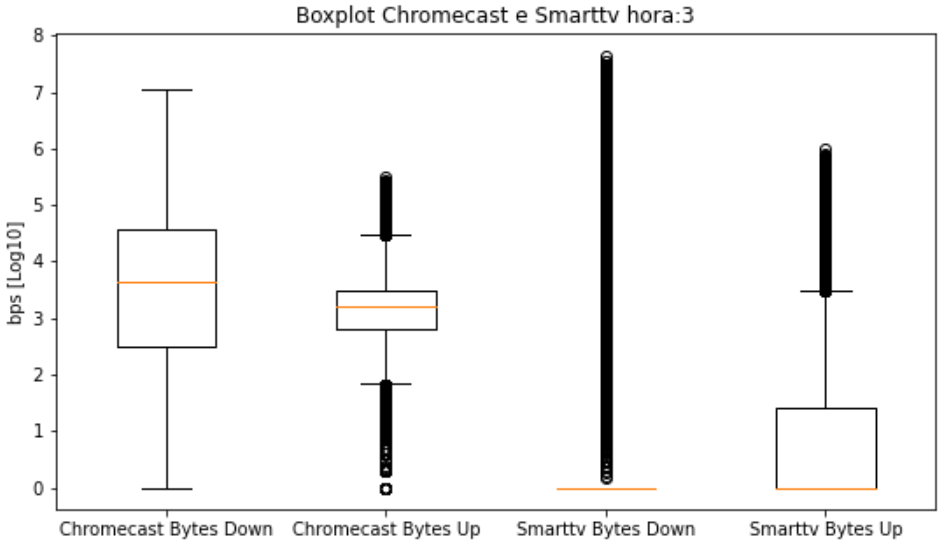

Figura 12 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 3

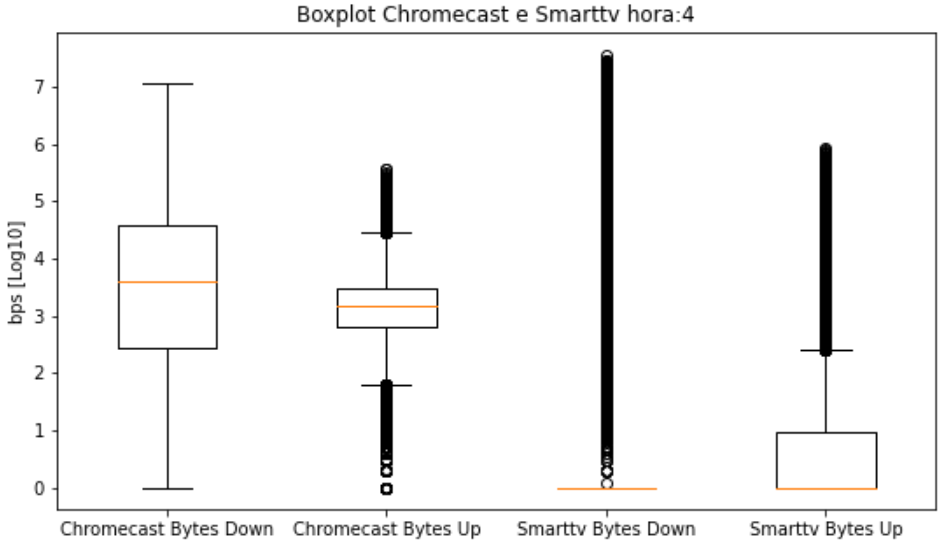

Figura 13 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 4

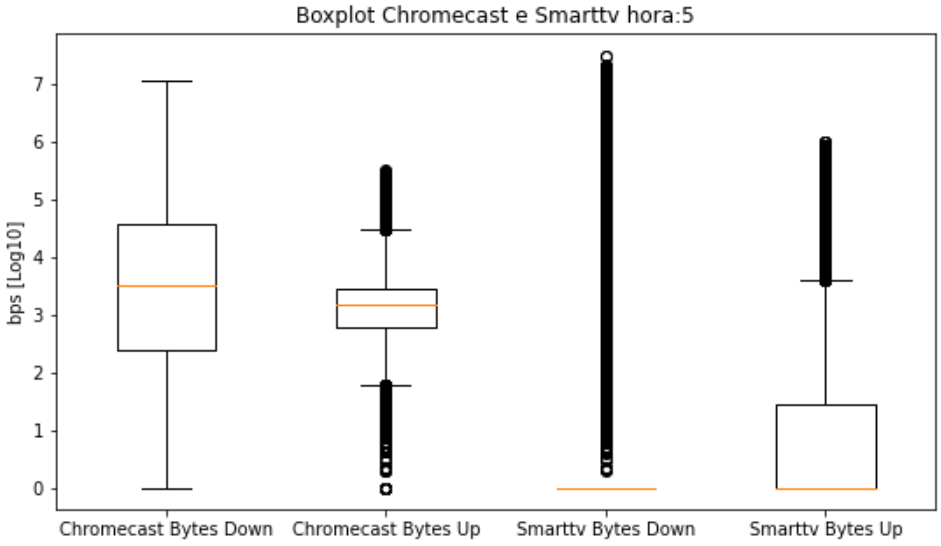

Figura 14 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 5

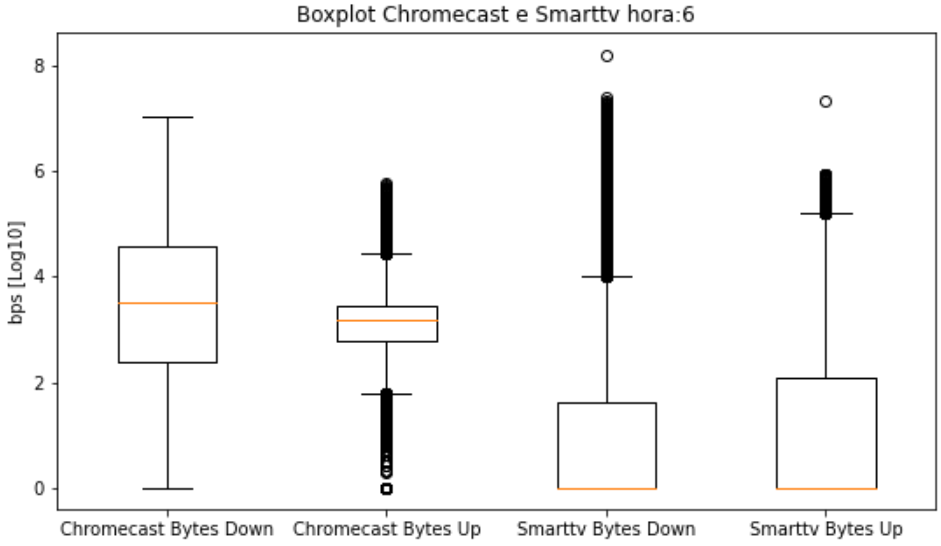

Figura 15 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 6

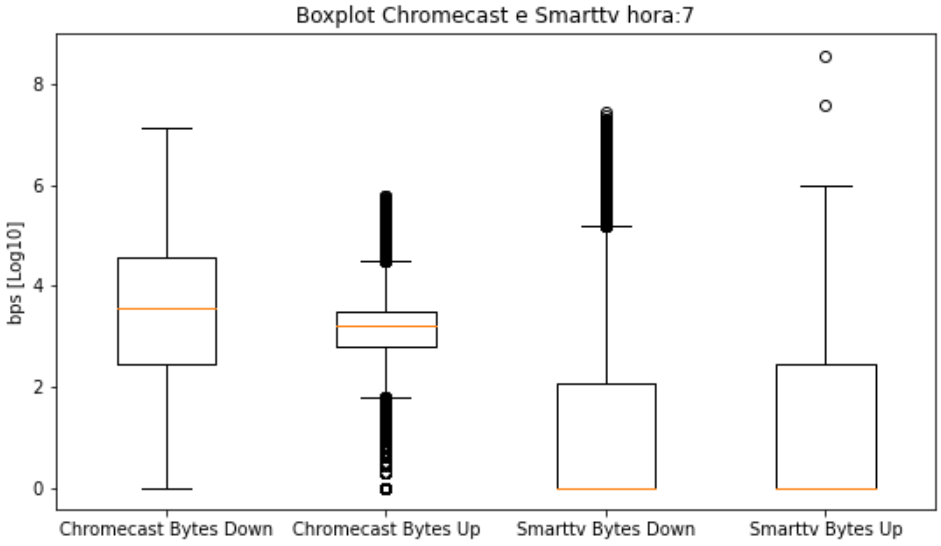

Figura 16 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 7

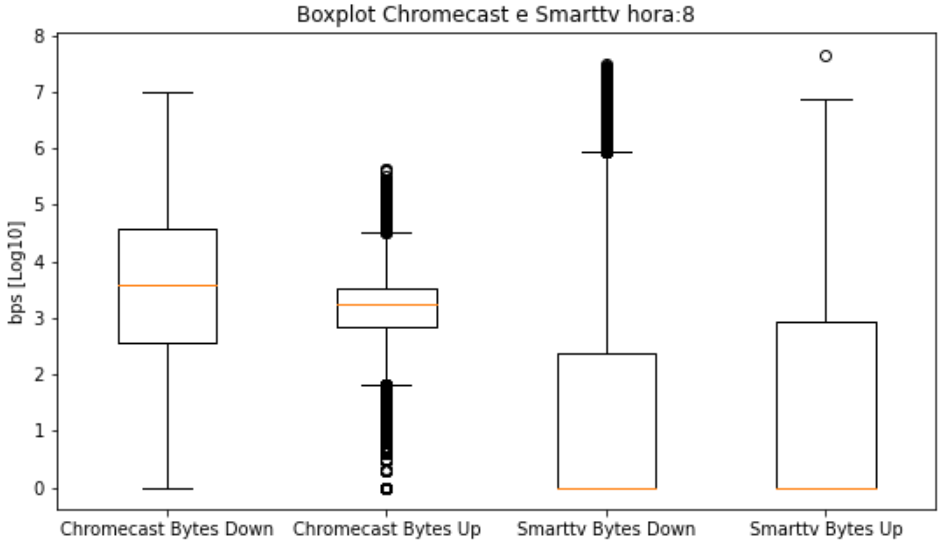

Figura 17 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 8

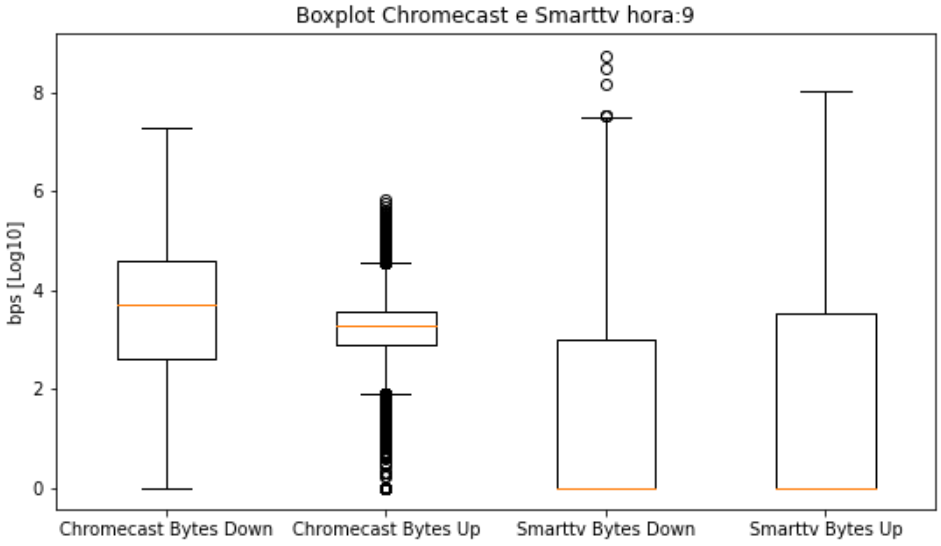

Figura 18 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 9

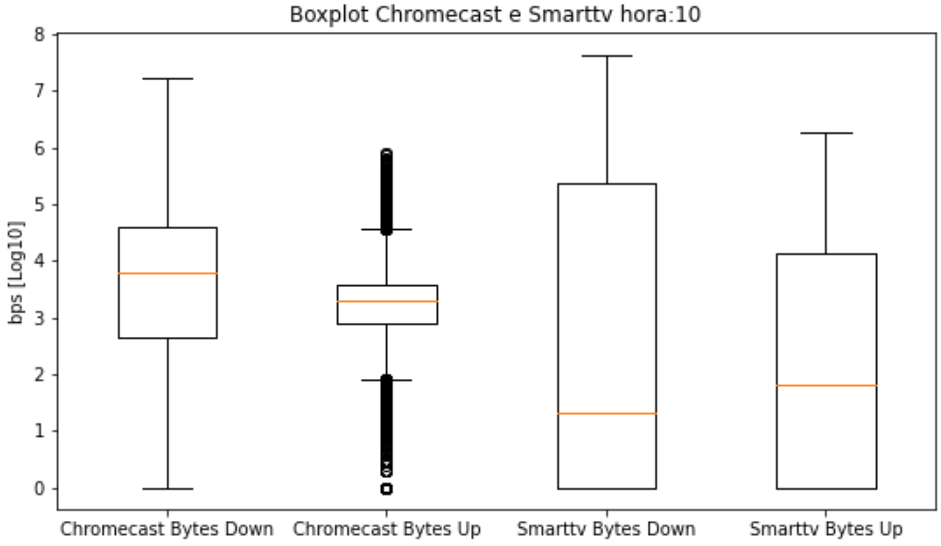

Figura 19 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 10

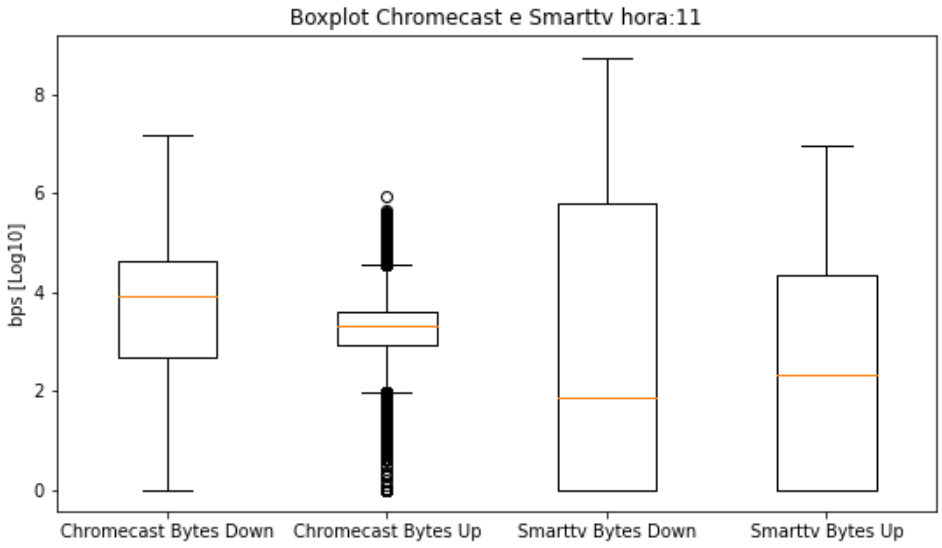

Figura 20 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 11

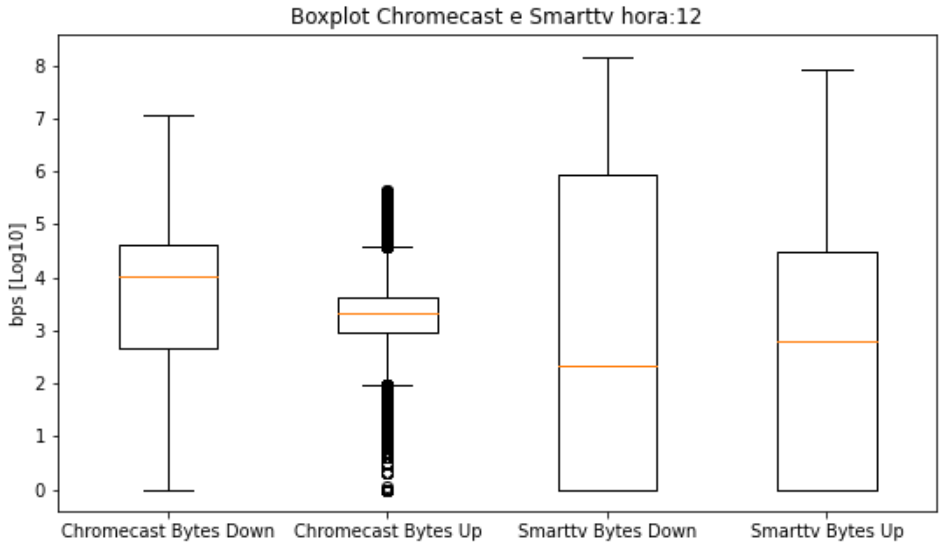

Figura 21 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 12

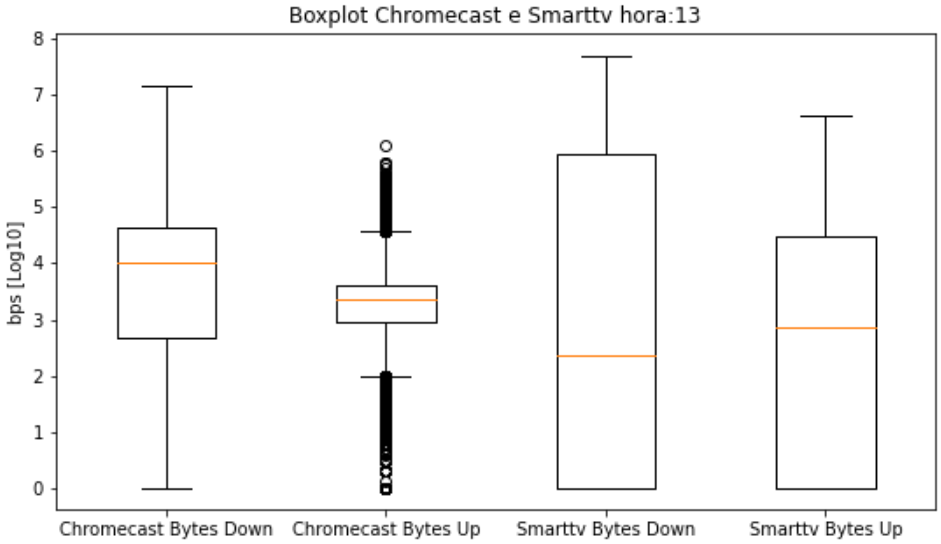

Figura 22 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 13

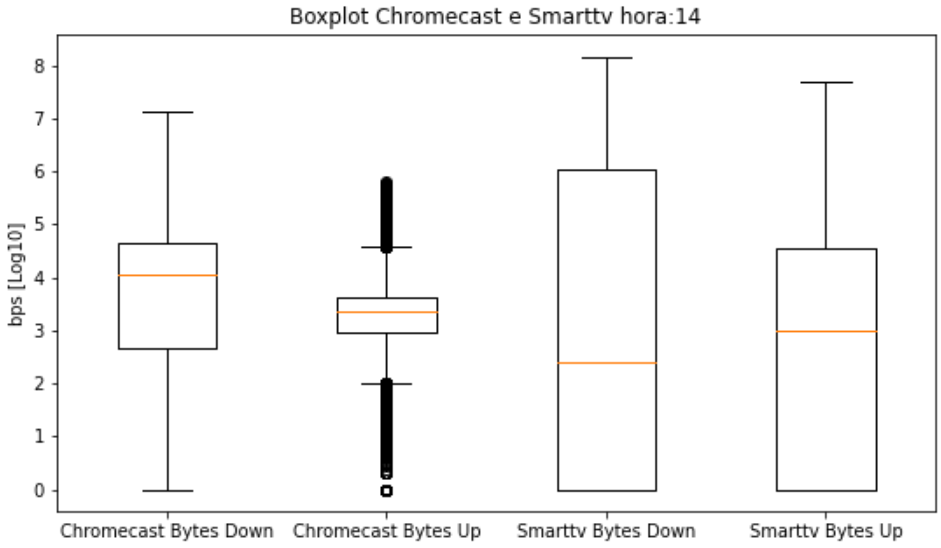

Figura 23 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 14

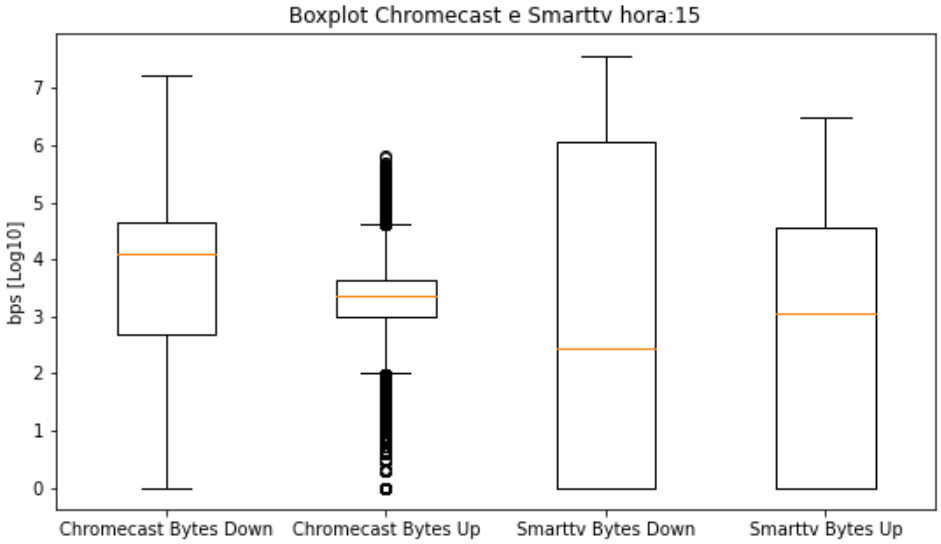

Figura 24 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 15

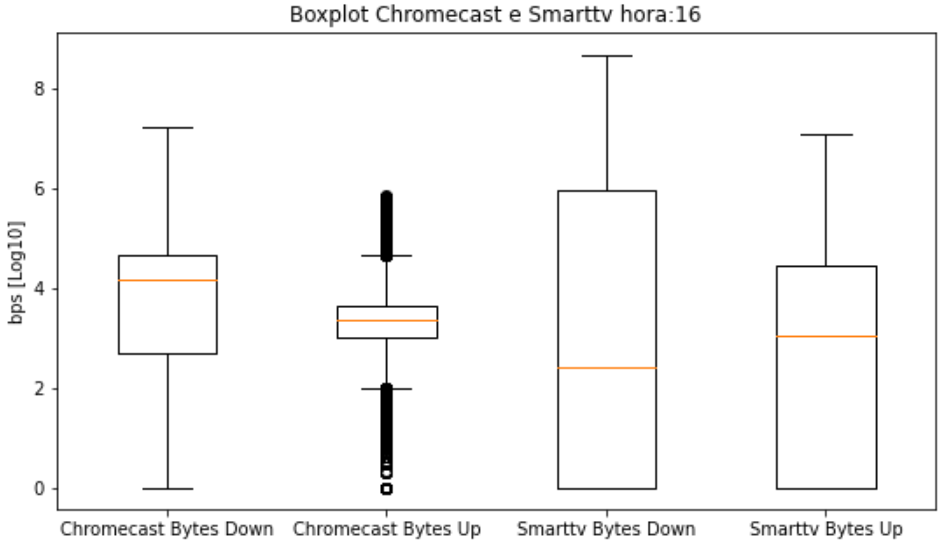

Figura 25 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 16

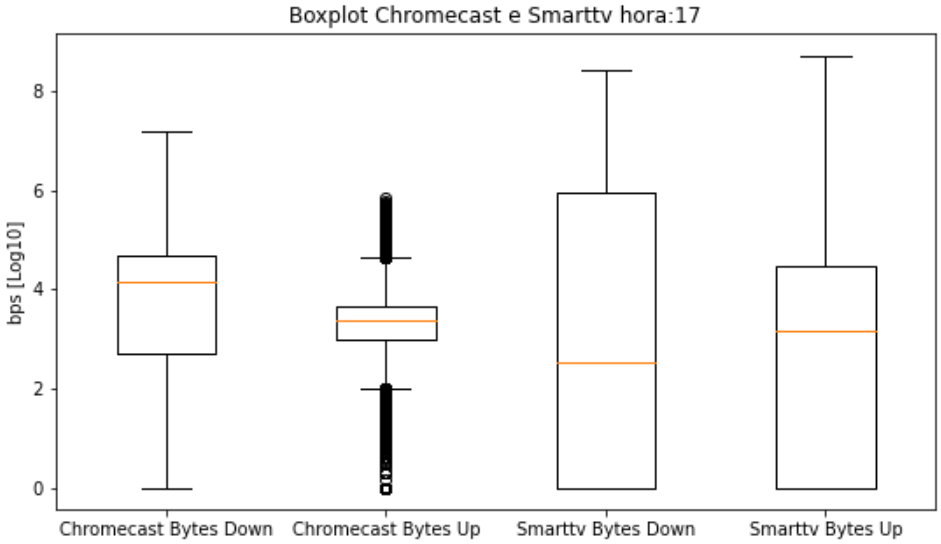

Figura 26 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 17

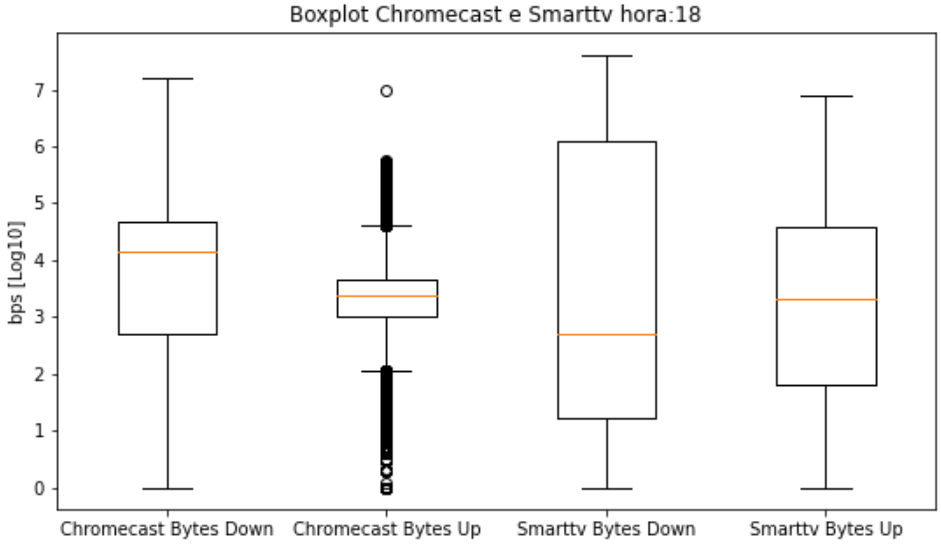

Figura 27 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 18

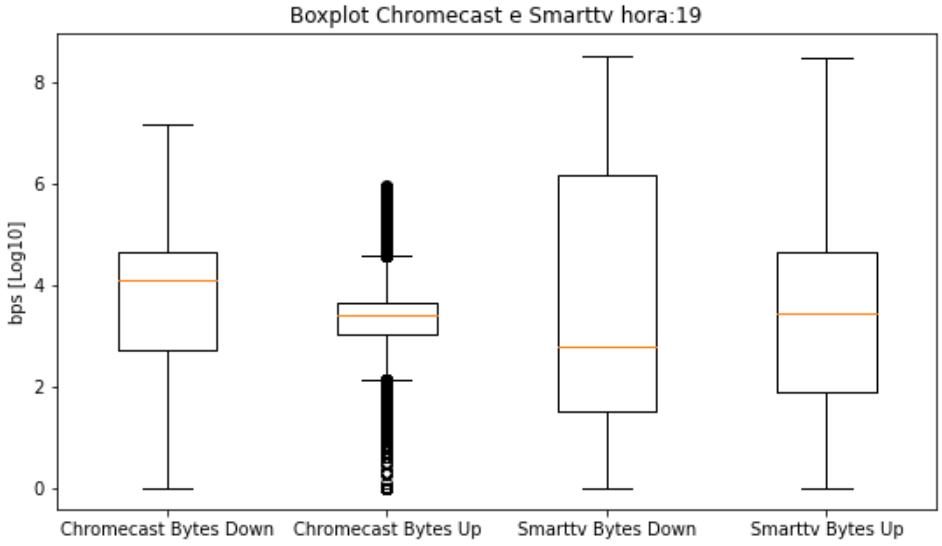

Figura 28 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 19

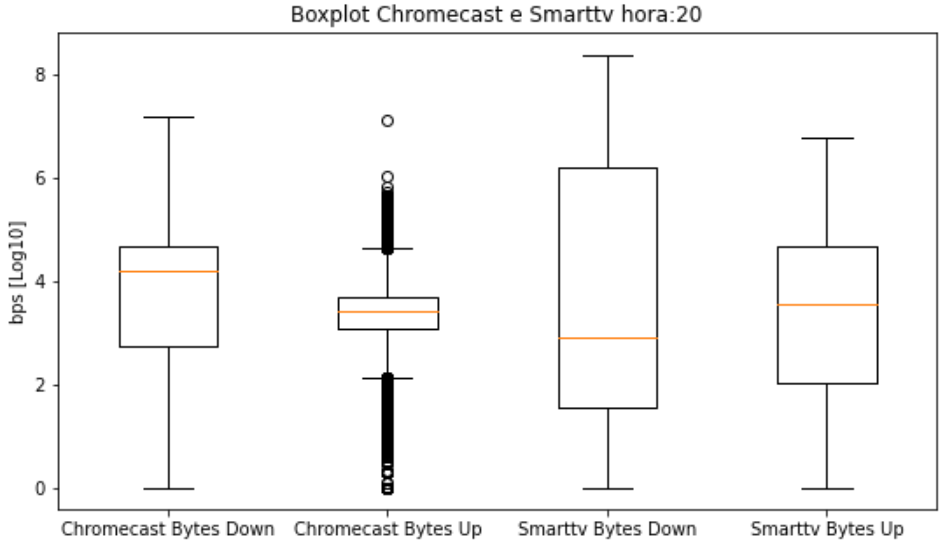

Figura 29 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 20

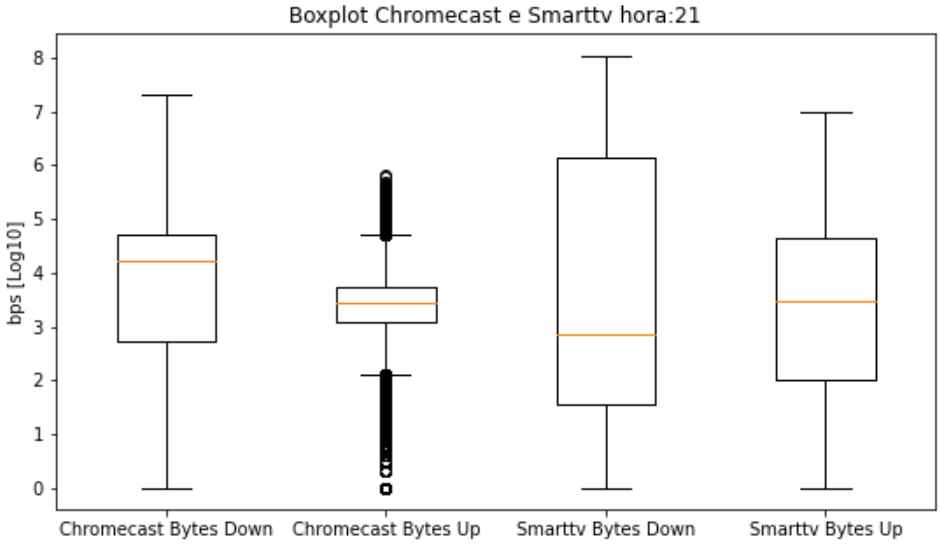

Figura 30 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 21

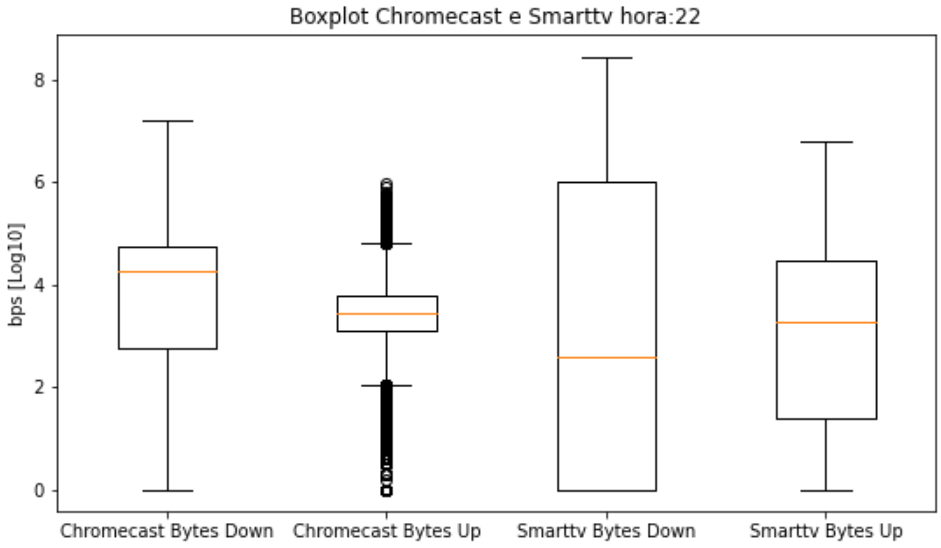

Figura 31 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 22

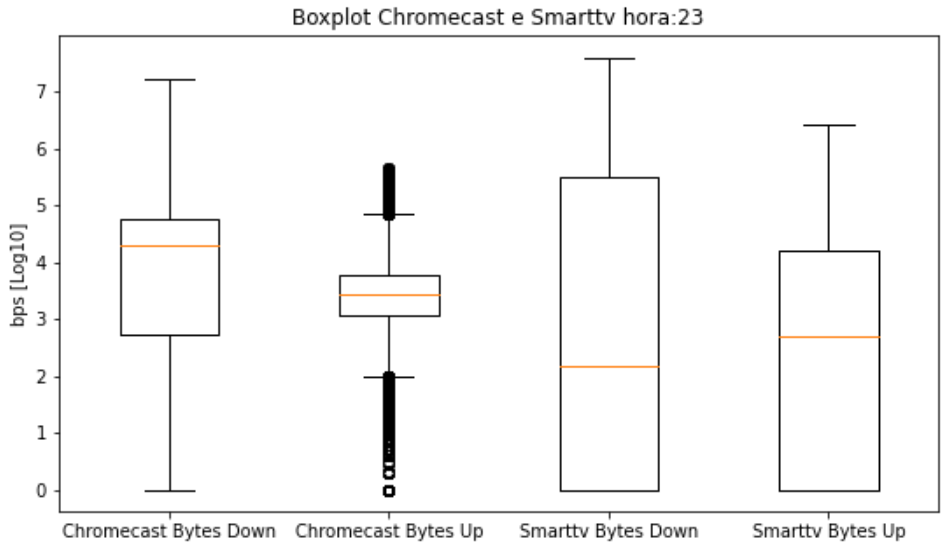

Figura 32 – Boxplots Download e Upload, Chromecast e Smart Tv para a hora 23

3.3- Média, Variância e Desvio Padrão

As estatísticas foram obtidas a partir da função 'groupby' do 'pandas', onde cada um dos agrupamentos foi realizado com uma estatística diferente. Isto é, para cada hora foram realizados três agrupamentos, um para média, um para variância e um para desvio padrão.

O resultado pode ser observado nos gráficos:

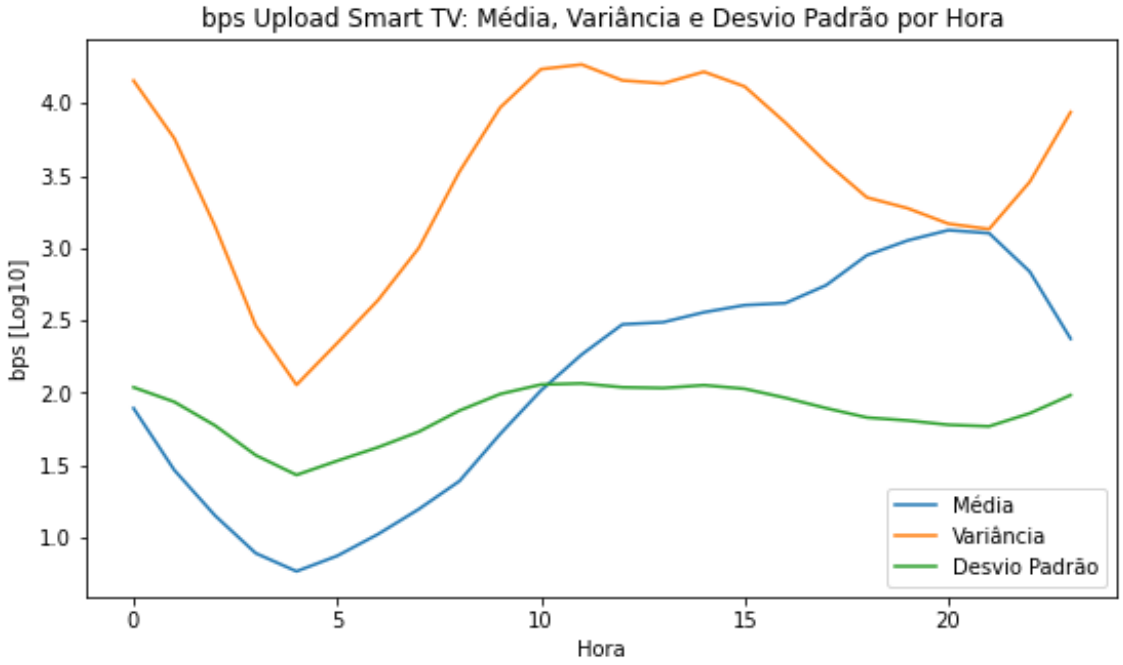

Figura 33 – Gráficos da Média, Variância e Desvio Padrão por hora – Upload Smart TV

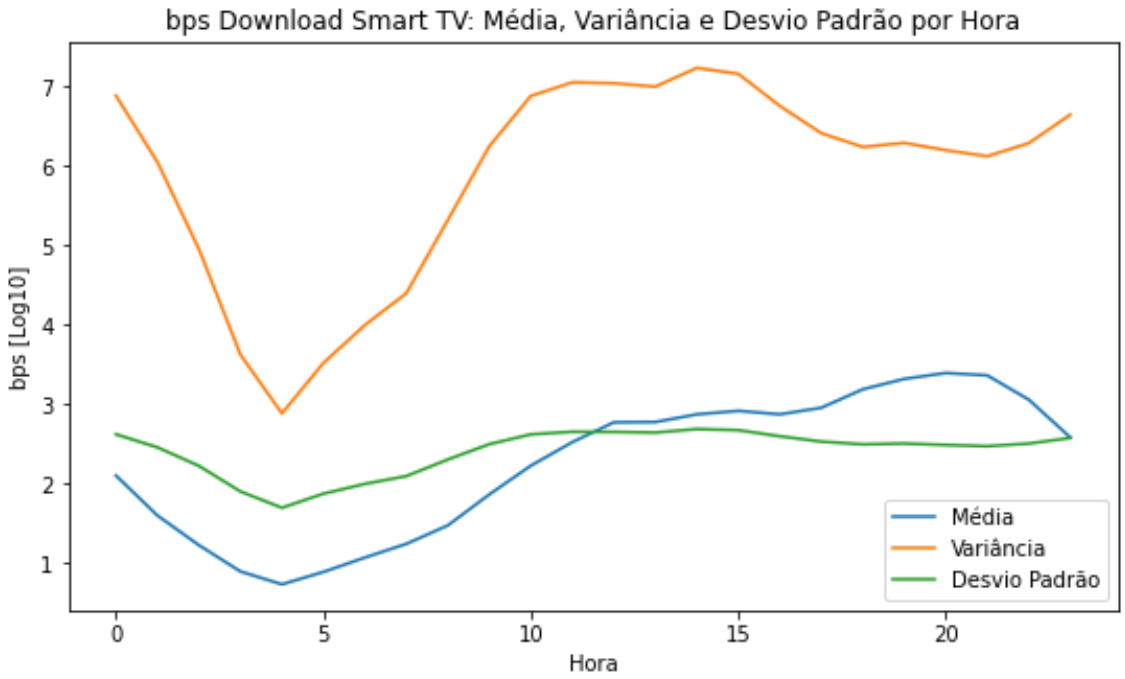

Figura 34 – Gráficos da Média, Variância e Desvio Padrão por hora – Download Smart TV

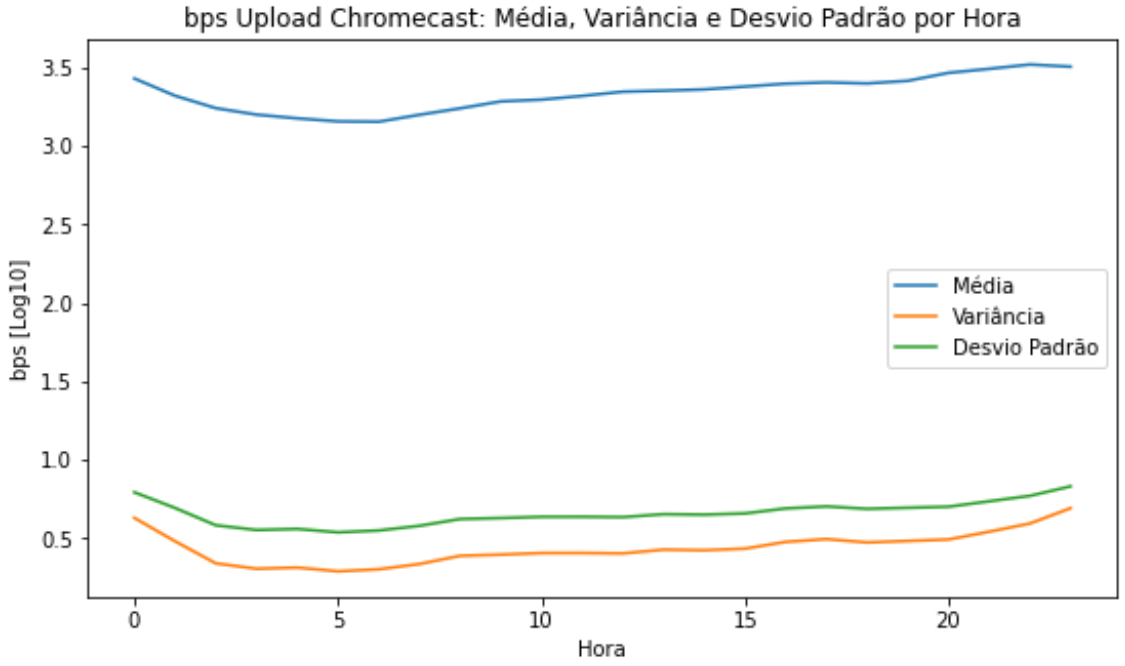

Figura 35 – Gráficos da Média, Variância e Desvio Padrão por hora – Upload Chromecast

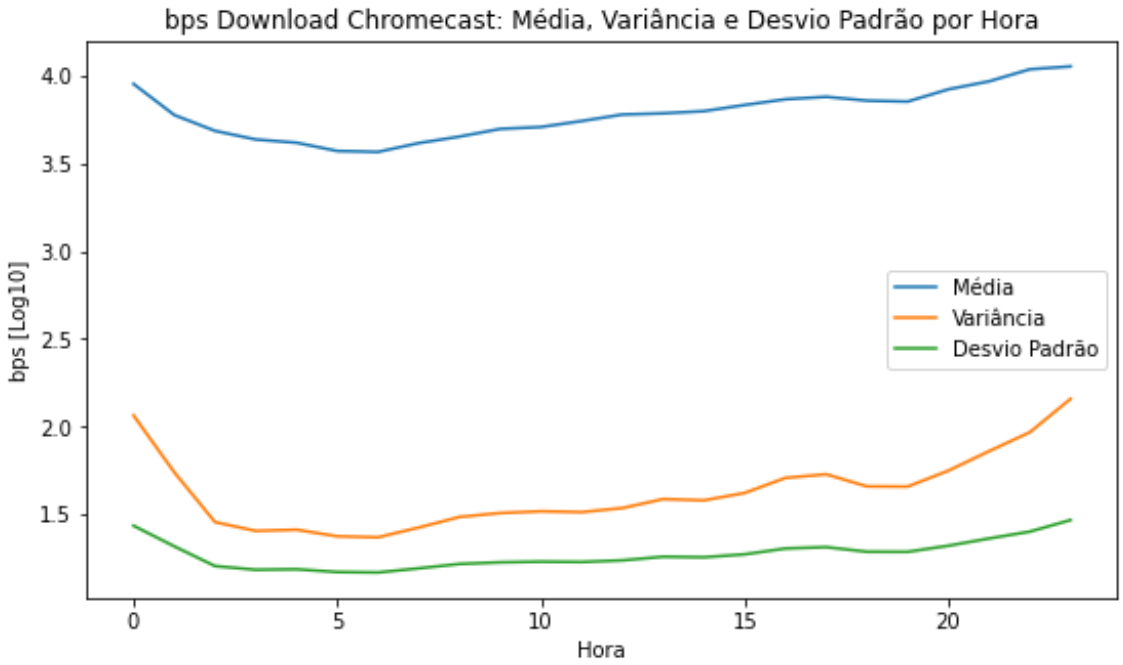

Figura 36 – Gráficos da Média, Variância e Desvio Padrão por hora – Download Chromecast

3.4 – Resultados

Algo que fica muito claro é que as médias são maiores na parte da noite, tendo um aumento por volta das 18 horas e retornando a reduzir por volta de 01 hora da manhã.

Um ponto que fica evidenciado nos boxplots das Smart Tvs é que o número de outliers para a parte superior de taxa de download é significativamente maior entre 01 hora e 08 horas, o que pode caracterizar um público específico que consome conteúdos durante o horário da madrugada/manhã.

Além disso, é possível notar que nos gráficos de média, os valores do Chromecast se mantem num fluxo parecido por várias horas do dia, enquanto os valores das Smart Tvs parecem ter um ciclo de consumo.

Por fim, perceptível que os fluxos de download e upload são proporcionais, mostrando um pouco sobre o comportamento dos dispositivos e como eles estão sempre trocando dados e não apenas recebendo.

4 - Caracterizando os horários com maior valor de tráfego

4.1- Criação dos datasets

Foram criados oito datasets para verificar os horários com maiores valores de tráfego – Upload e Download – em cada um dos dispositivos – Smart Tv e Chromecast.

Cada dataset foi construído utilizando a função ‘groupby’ do ‘pandas’, utilizando a média e a mediana e a função ‘idmax’ para verificar qual a hora com maior volume de tráfego.

A partir disso foram obtidos os oito datasets abaixo:

- Dataset 1: Horário com a maior mediana da taxa de upload em uma hora, Smart-TV
- Dataset 2: Horário com a maior média da taxa de upload em uma hora, Smart-TV
- Dataset 3: Horário com a maior mediana da taxa de download em uma hora, Smart-TV
- Dataset 4: Horário com a maior média da taxa de download em uma hora, Smart-TV
- Dataset 5: Horário com a maior mediana da taxa de upload em uma hora, Chromecast

- Dataset 6: Horário com a maior média da taxa de upload em uma hora, Chromecast
- Dataset 7: Horário com a maior mediana da taxa de download em uma hora, Chrome-
- cast
- Dataset 8: Horário com a maior média da taxa de download em uma hora, Chromecast

E os horários com maior volume de tráfego para cada um dos datasets:

- Dataset 1: 20 horas
- Dataset 2: 20 horas
- Dataset 3: 20 horas
- Dataset 4: 20 horas
- Dataset 5: 22 horas
- Dataset 6: 22 horas
- Dataset 7: 23 horas
- Dataset 8: 23 horas

4.2- Histogramas

O número de bins foi definido pela Fórmula 1 -Método de Sturges- e, como cada dataset só contém as observações da sua hora correspondente não foi necessário realizar nenhum agrupamento.

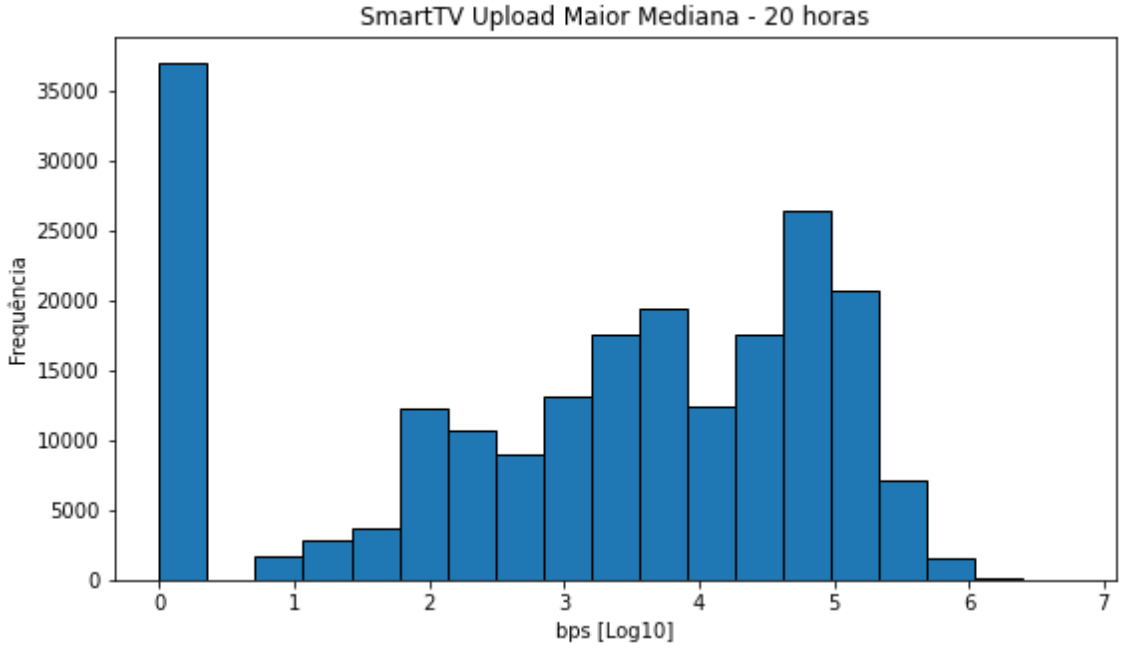

Figura 37 – Histograma – Smart Tv, Upload, Mediana – 20 Horas

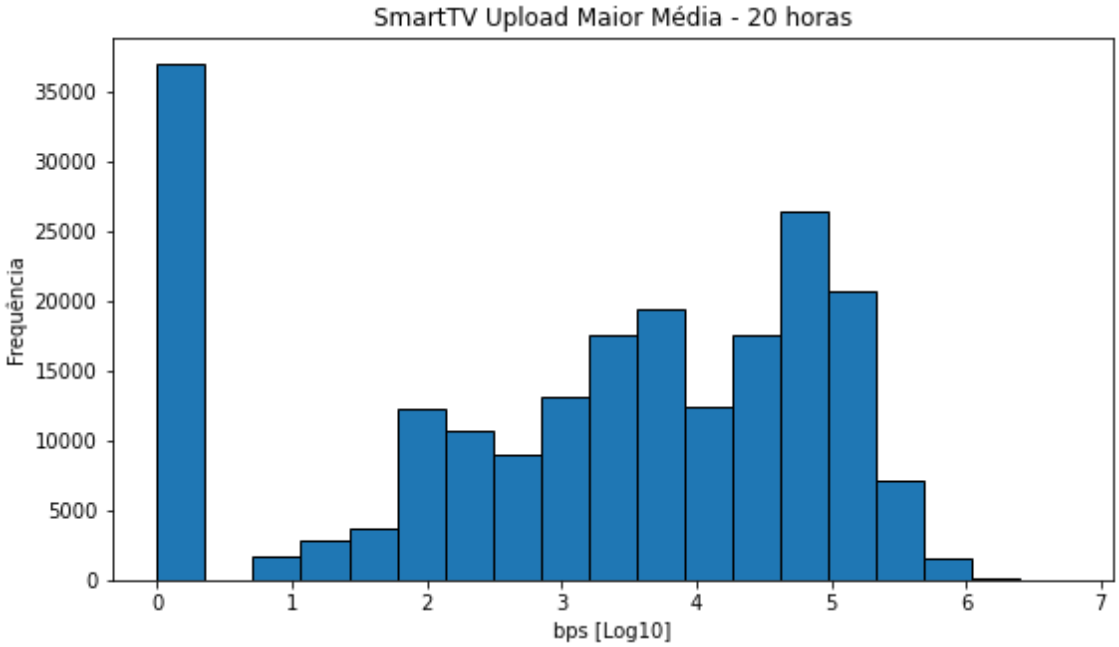

Figura 38 – Histograma – Smart Tv, Upload, Média – 20 Horas

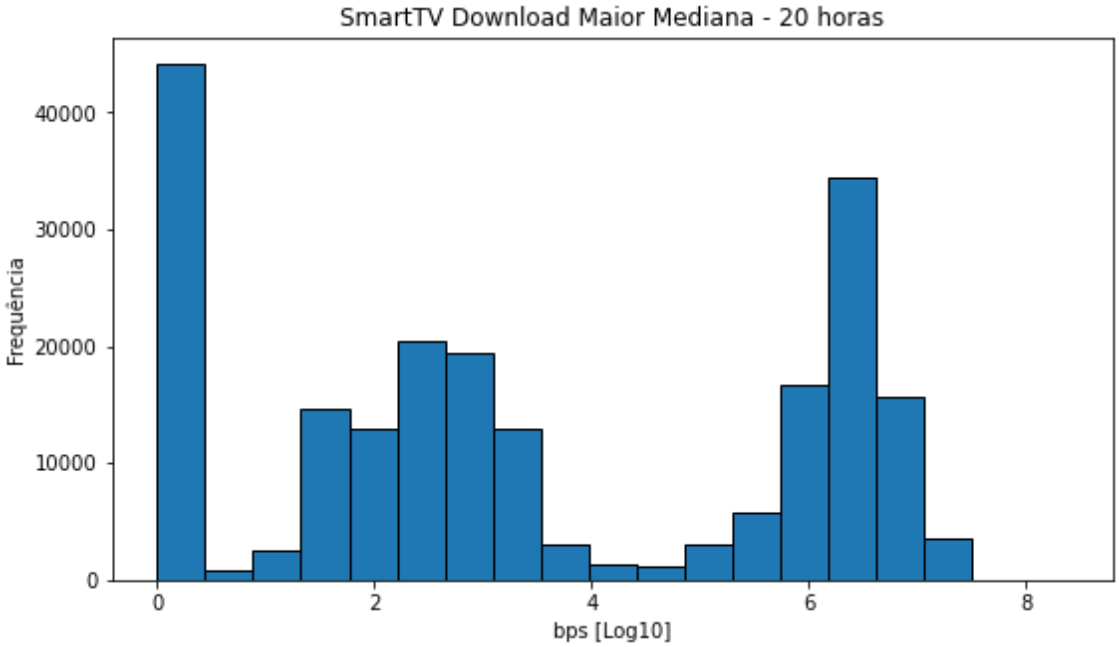

Figura 39 – Histograma – Smart Tv, Mediana, Download – 20 Horas

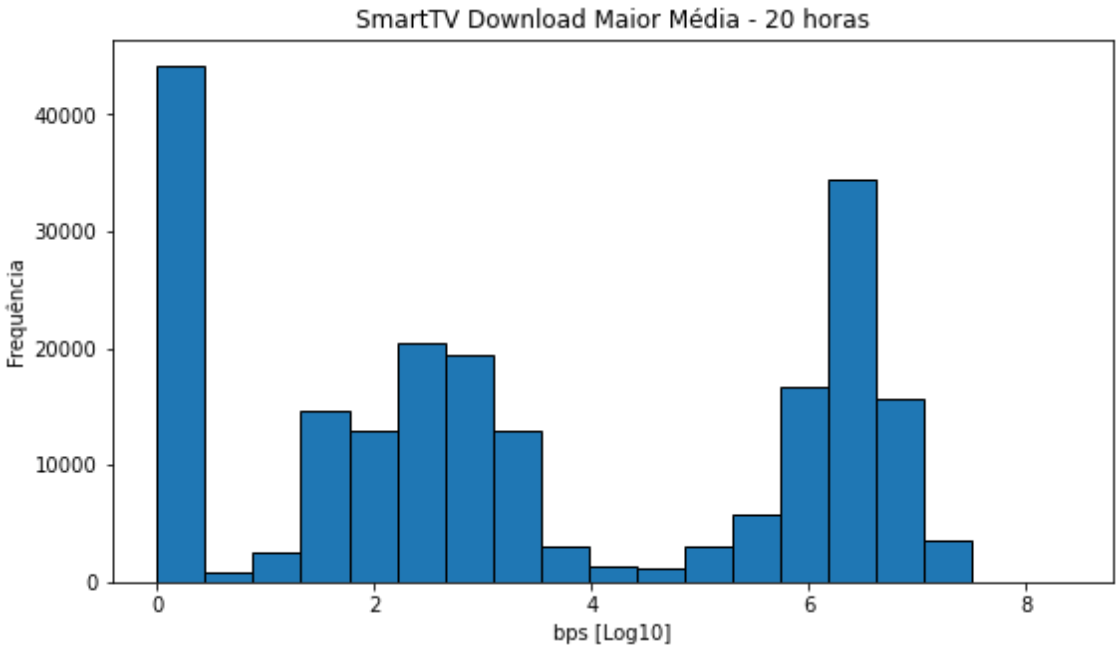

Figura 40 – Histograma – Smart Tv, Média, Download – 20 Horas

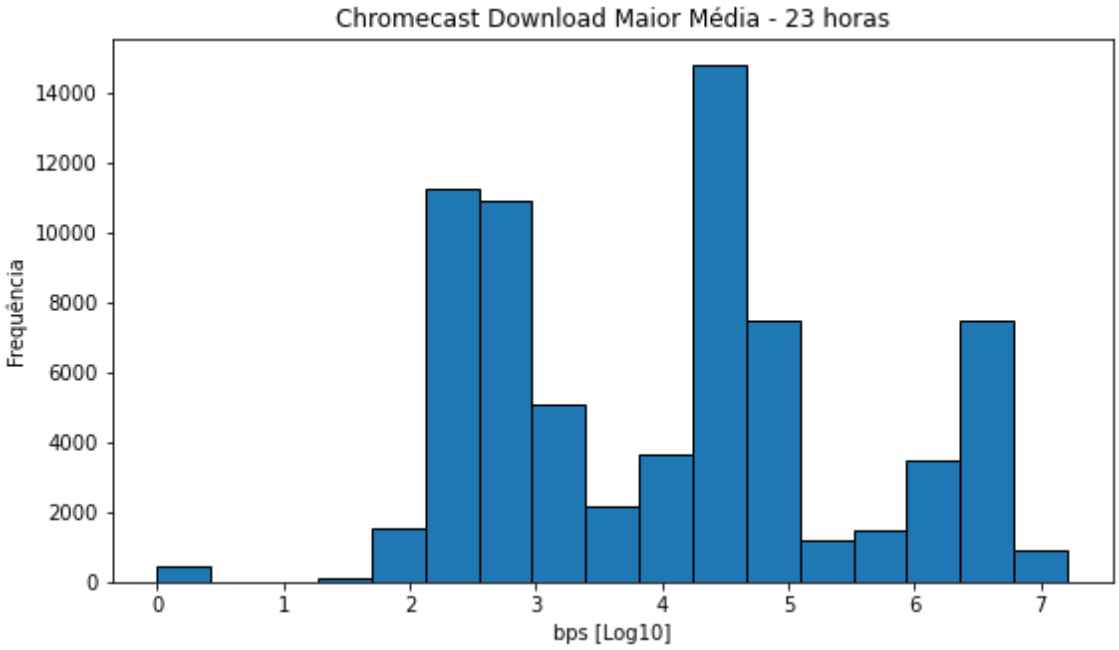

Figura 41 – Histograma – Chromecast, Média, Download – 23 Horas

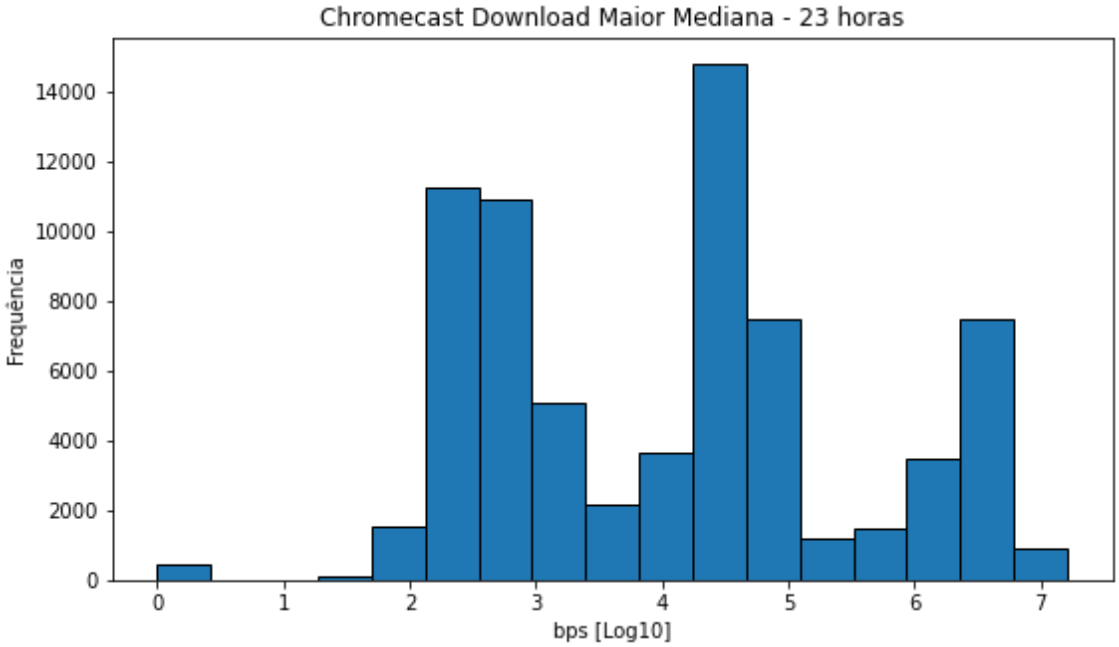

Figura 42 – Histograma – Chromecast, Mediana, Download – 23 Horas

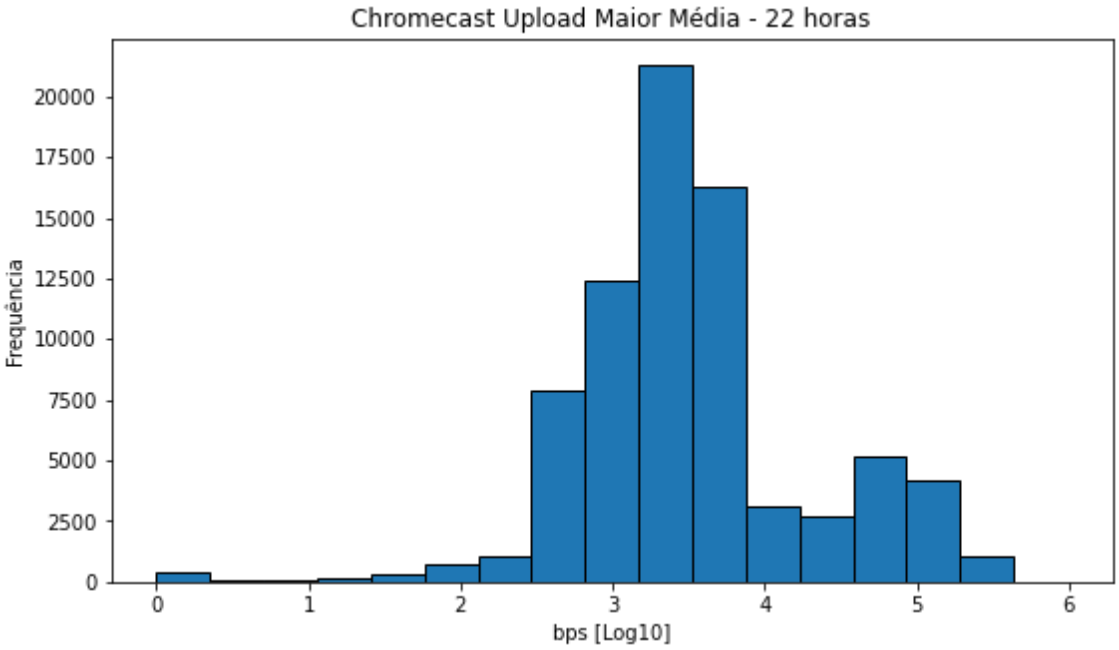

Figura 43 – Histograma – Chromecast, Média, Upload – 22 Horas

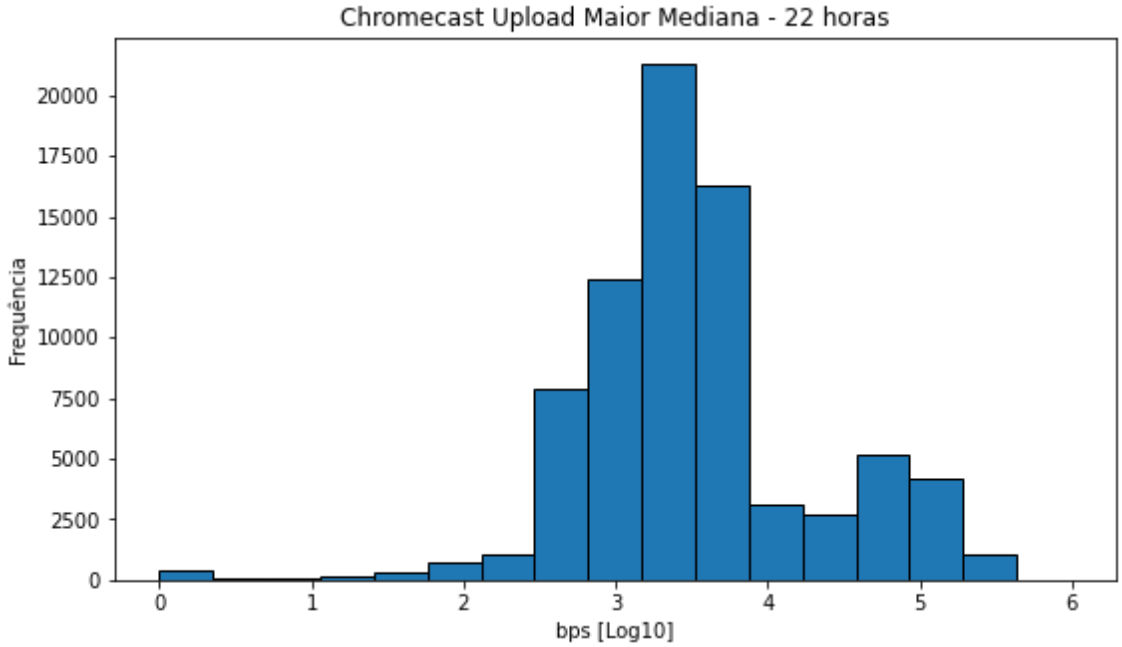

Figura 44 – Histograma – Chromecast, Mediana, Upload – 22 Horas

4.3- MLE

4.3.1- Gaussiana

A Gaussiana ou distribuição normal, necessita de dois parâmetros: a média e o desvio padrão. Que podem ser obtidos pelas funções ‘mean’ e ‘std’ do próprio ‘pandas’.

Os valores obtidos foram:

- MLE dataset1 - Média: 3.124258107506702 / Desvio Padrão: 1.7800995973532652
- MLE dataset2 - Média: 3.124258107506702 / Desvio Padrão: 1.7800995973532652
- MLE dataset3 - Média: 3.396094556436504 / Desvio Padrão: 2.490255525973841
- MLE dataset4 - Média: 3.396094556436504 / Desvio Padrão: 2.490255525973841
- MLE dataset5 - Média: 3.521546370674621 / Desvio Padrão: 0.7718286854202554
- MLE dataset6 - Média: 3.521546370674621 / Desvio Padrão: 0.7718286854202554
- MLE dataset7 - Média: 4.05269811265878 / Desvio Padrão: 1.4694860227345465
- MLE dataset8 - Média: 4.05269811265878 / Desvio Padrão: 1.4694860227345465

4.3.2- Gamma

A função Gamma tem três parâmetros: shape, offset e scale. Que foram obtidos a partir da função ‘gamma.fit’ do módulo ‘scipy’.

Que, por sua vez, obtém seus parâmetros com as seguintes heurísticas:

shape := 4/Assimetria(X, I)^2

offset := Média(X, I) - DesvioPadrão(X, I)*RaizQuadrada(shape)

scale := Variância(X, I)/(Média(X, I) - offset)

A partir disso, os seguintes valores foram obtidos:

- MLE dataset1 - Shape: 220.48073768362616 / Offset: -23.96174486447611 / Scale: 0.12272309565740211

- MLE dataset2 - Shape: 220.48073768362616 / Offset: -23.96174486447611 / Scale: 0.12272309565740211
- MLE dataset3 - Shape: 896.5469322463027 / Offset: -71.06216506397283 / Scale: 0.08304989773768084
- MLE dataset4 - Shape: 896.5469322463027 / Offset: -71.06216506397283 / Scale: 0.08304989773768084
- MLE dataset5 - Shape: 3148.8815211215233 / Offset: -39.80898262399852 / Scale: 0.013760617087012861
- MLE dataset6 - Shape: 3148.8815211215233 / Offset: -39.80898262399852 / Scale: 0.013760617087012861
- MLE dataset7 - Shape: 27.130143662628548 / Offset: -3.631368185066651 / Scale: 0.2832298335465073
- MLE dataset8 - Shape: 27.130143662628548 / Offset: -3.631368185066651 / Scale: 0.2832298335465073

4.3.3- Gráficos MLE

A partir dos valores definidos, um gráfico foi construído para cada dataset, contendo as curvas gerada pelos parâmetros das gaussianas e gamas encontradas e seus histogramas.

Vale ressaltar que os valores para o eixo y dos gráficos correspondem apenas as densidades das funções das gaussianas e gamas, visto que o histograma está fora de escala e desempenha papel visual nestes gráficos.

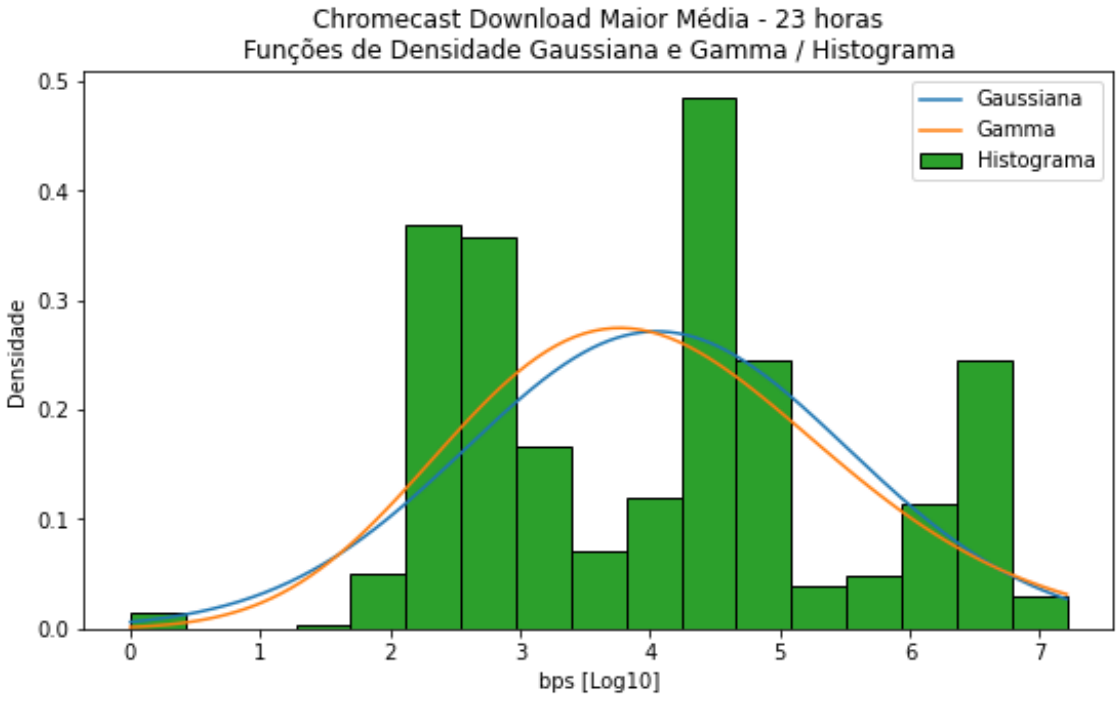

Figura 45 – Histograma e Função Densidade Gamma e Gaussiana – Chromecast, Média, Download – 23 Horas

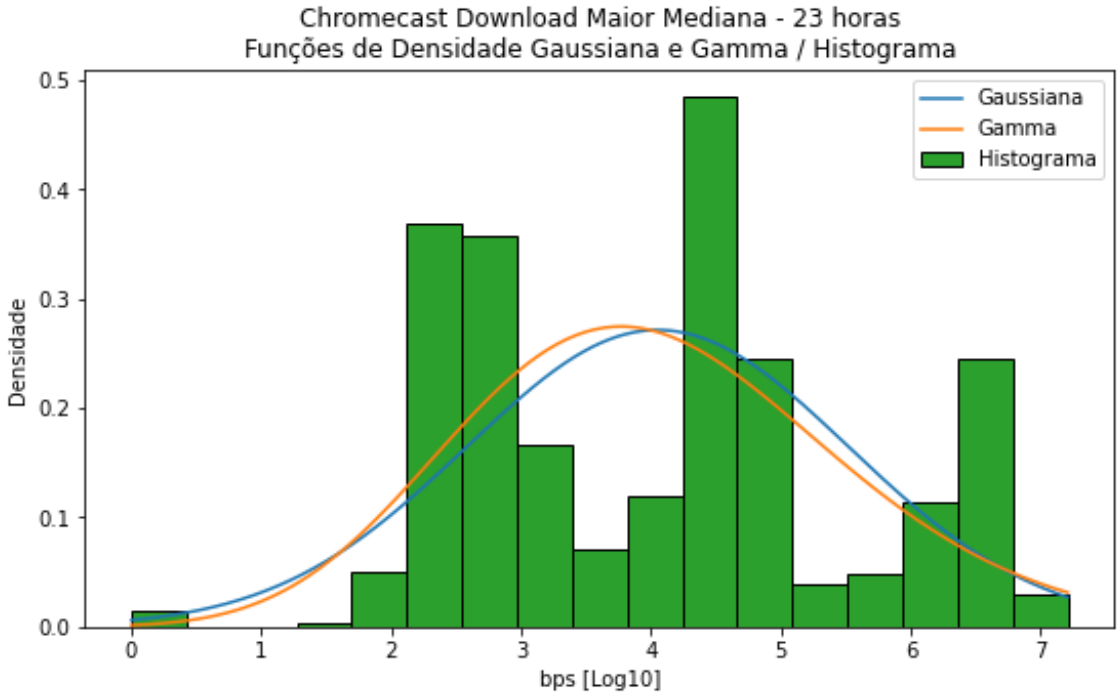

Figura 46 – Histograma e Função Densidade Gamma e Gaussiana – Chromecast, Mediana, Download – 23 Horas

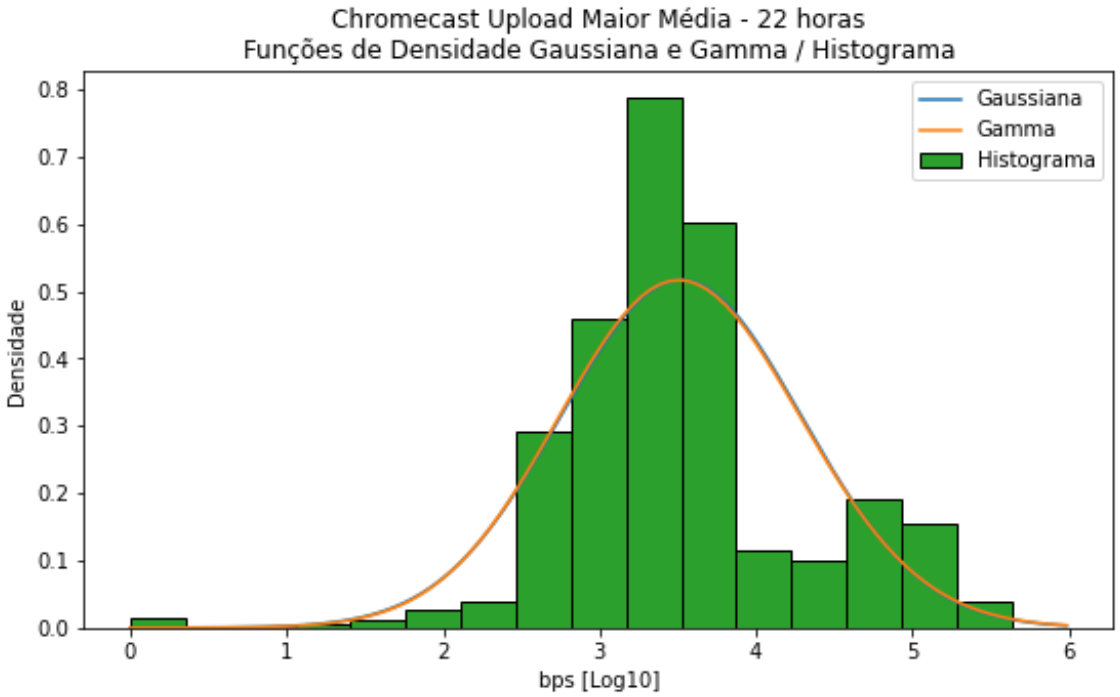

Figura 47 – Histograma e Função Densidade Gamma e Gaussiana – Chromecast, Média, Upload – 22 Horas

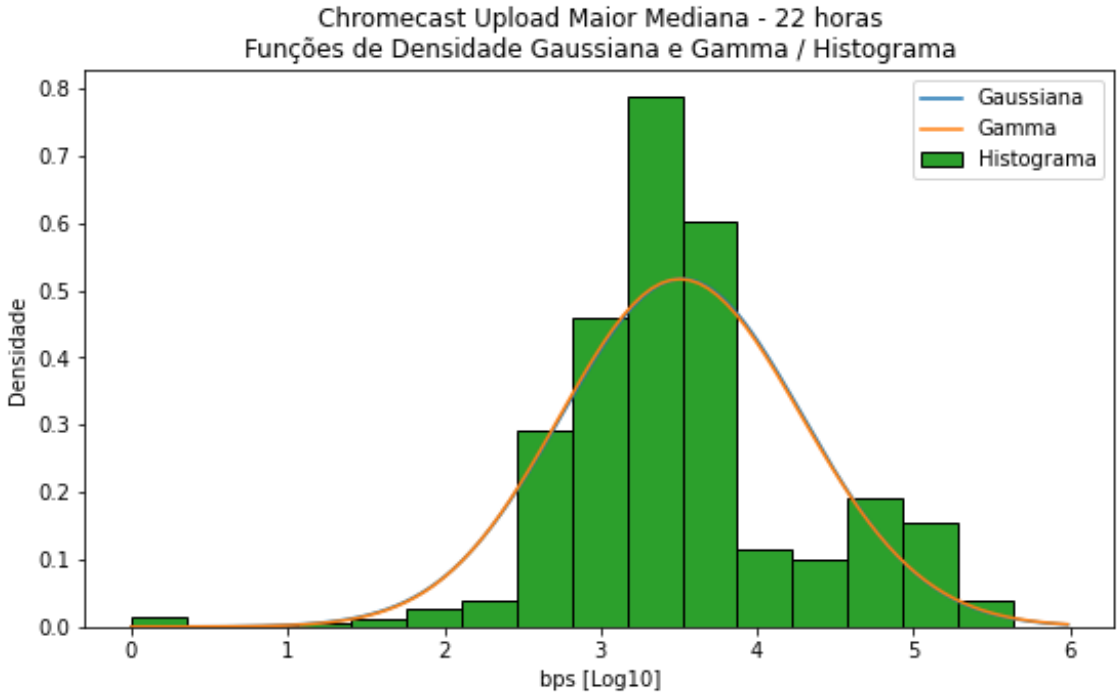

Figura 48 – Histograma e Função Densidade Gamma e Gaussiana – Chromecast, Mediana, Upload – 22 Horas

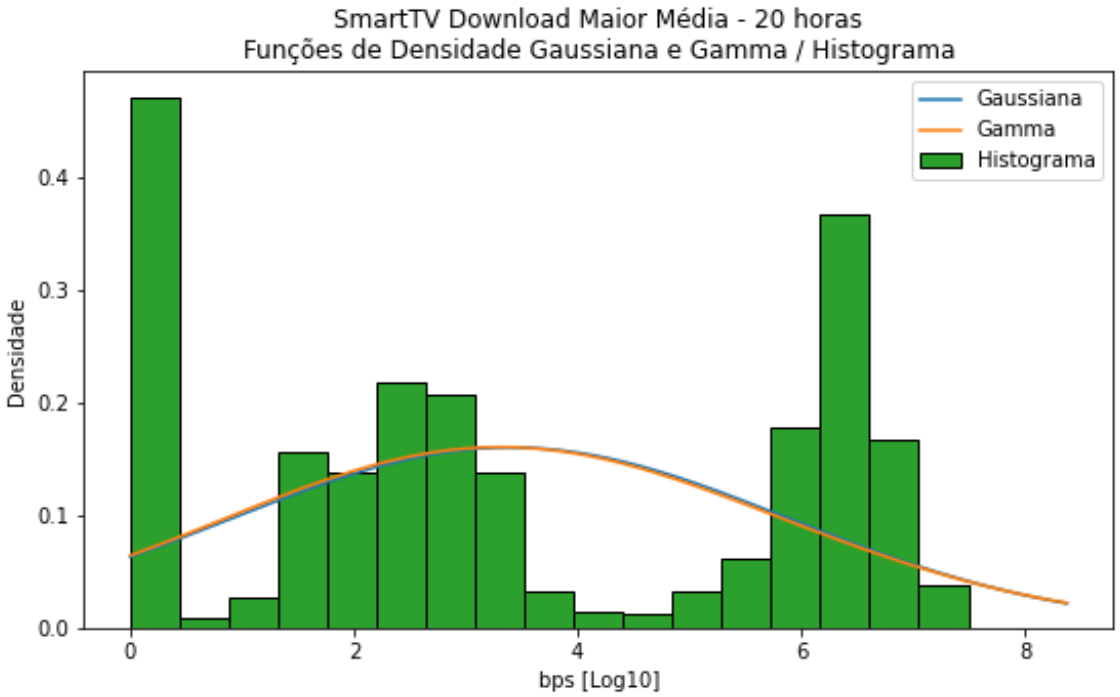

Figura 49 – Histograma e Função Densidade Gamma e Gaussiana – Smart TV, Média, Download – 20 Horas

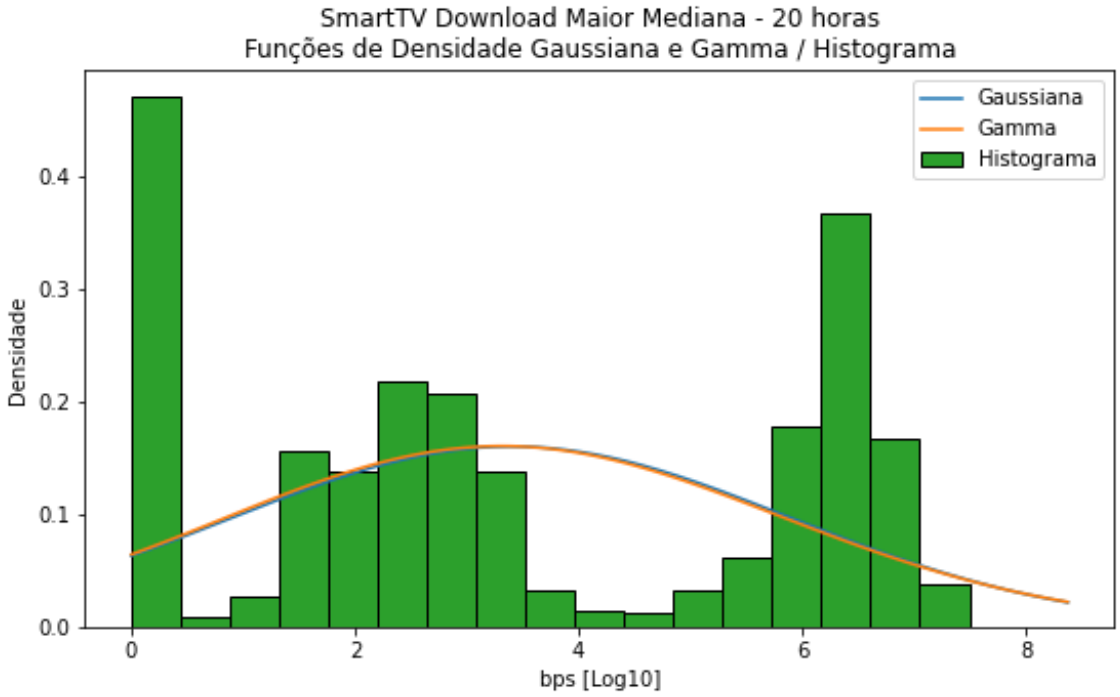

Figura 50 – Histograma e Função Densidade Gamma e Gaussiana – Smart TV, Mediana, Download – 20 Horas

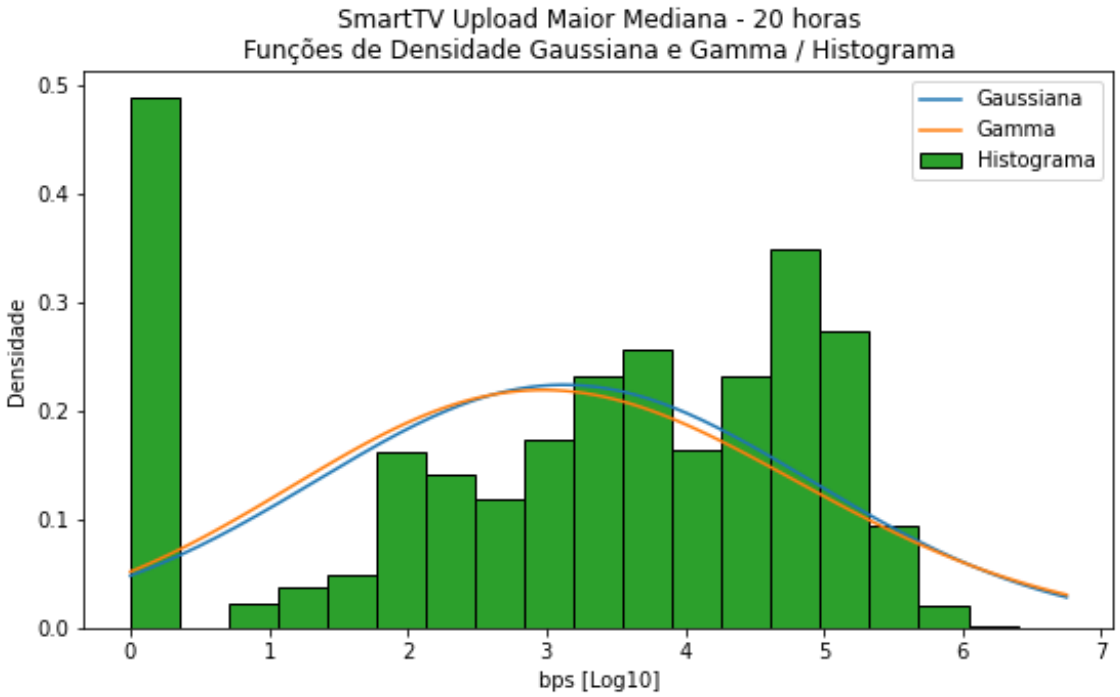

Figura 51 – Histograma e Função Densidade Gamma e Gaussiana – Smart TV, Mediana, Upload – 20 Horas

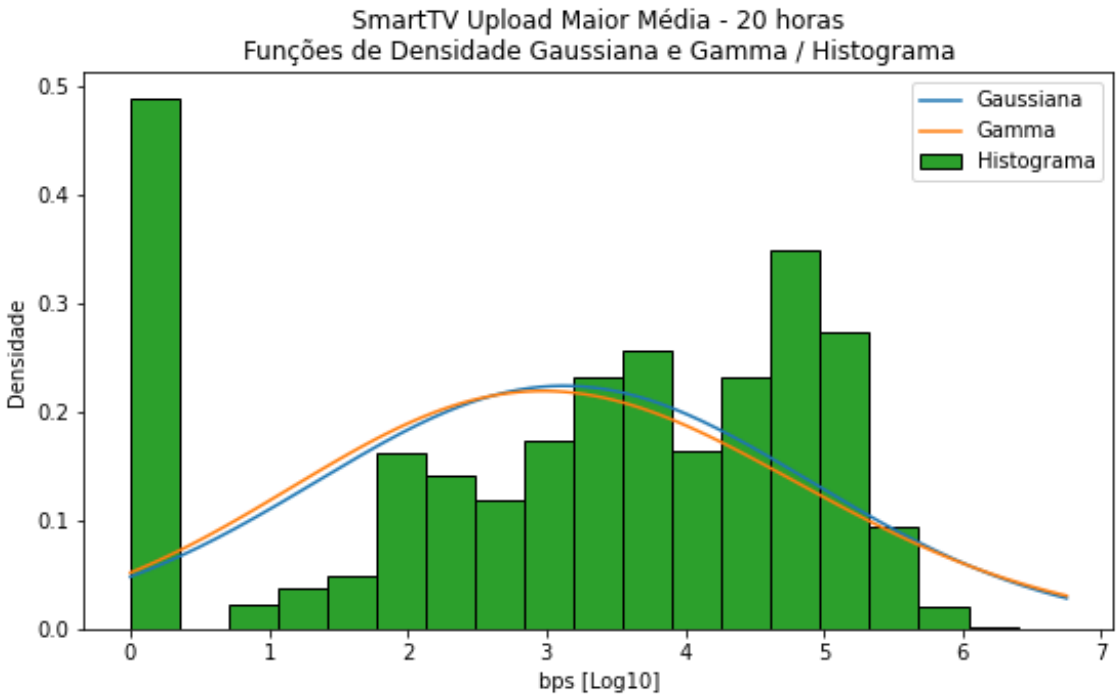

Figura 52 – Histograma e Função Densidade Gamma e Gaussiana – Smart TV, Média, Upload – 20 Horas

4.4- Gráfico de Probabilidade

Para a criação dos gráficos de probabilidade, foi utilizada a função ‘probplot’ do ‘scipy’, que recebe os parâmetros obtidos do MLE e qual a função que ele construirá.

Além disso, recebe os dados e os compara com a distribuição teórica encontrada. A reta vermelha representa o quão próximo os valores obtidos (em azul) estão da distribuição teórica que foi parametrizada.

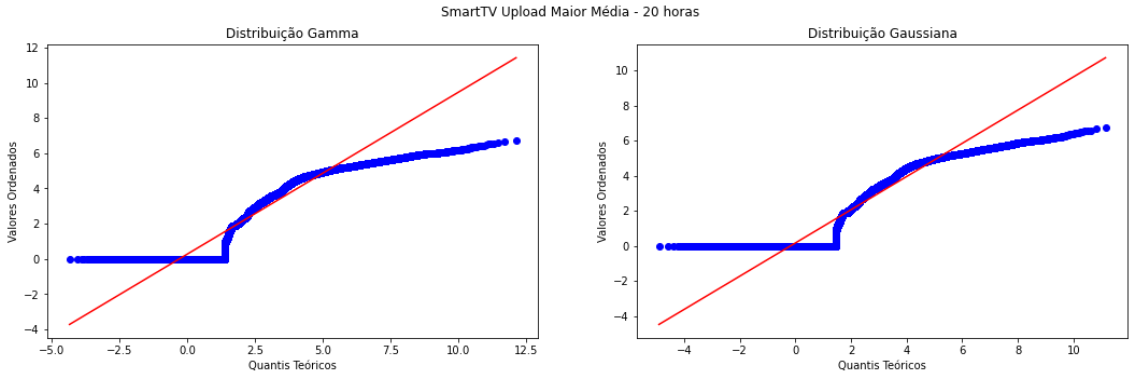

Figura 53 – Gráfico de Probabilidade – Smart TV, Média, Upload – 20 Horas

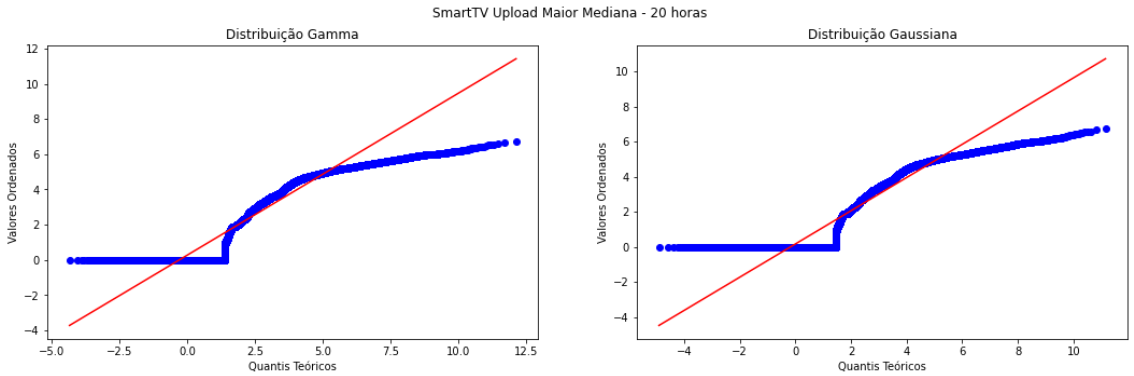

Figura 54 – Gráfico de Probabilidade – Smart TV, Mediana, Upload – 20 Horas

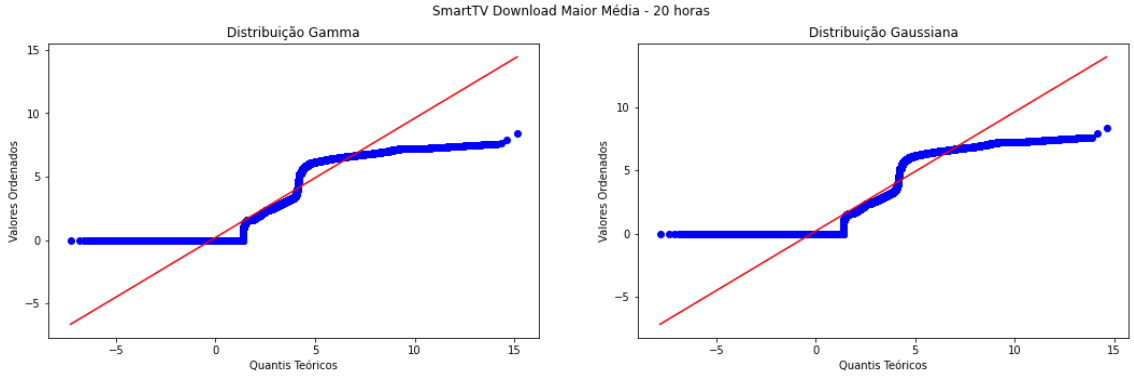

Figura 55 – Gráfico de Probabilidade – Smart TV, Média, Download – 20 Horas

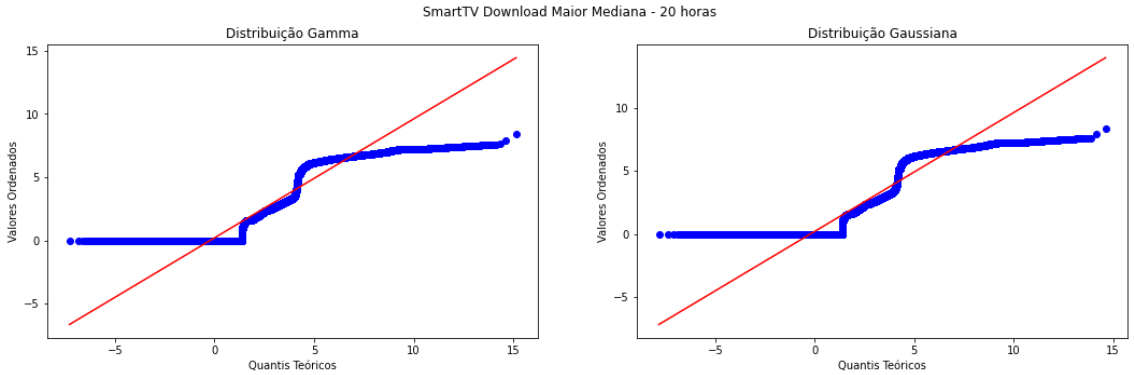

Figura 56 – Gráfico de Probabilidade – Smart TV, Mediana, Download – 20 Horas

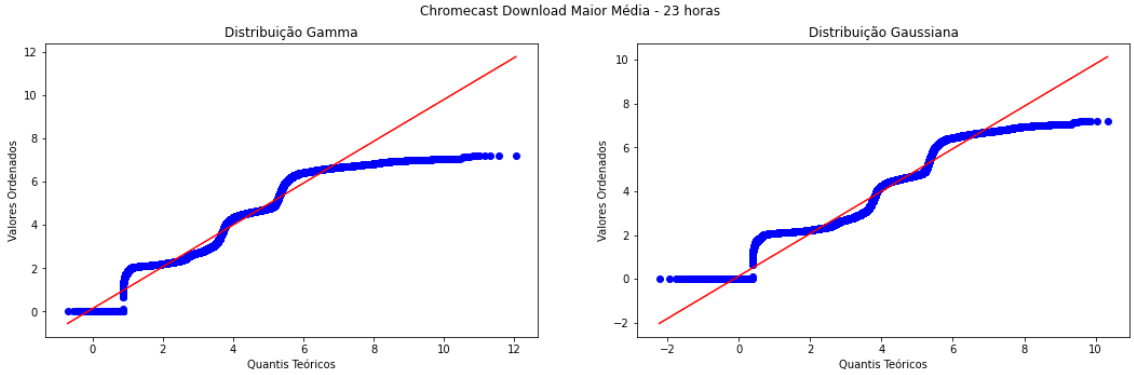

Figura 57 – Gráfico de Probabilidade – Chromecast, Média, Download – 23 Horas

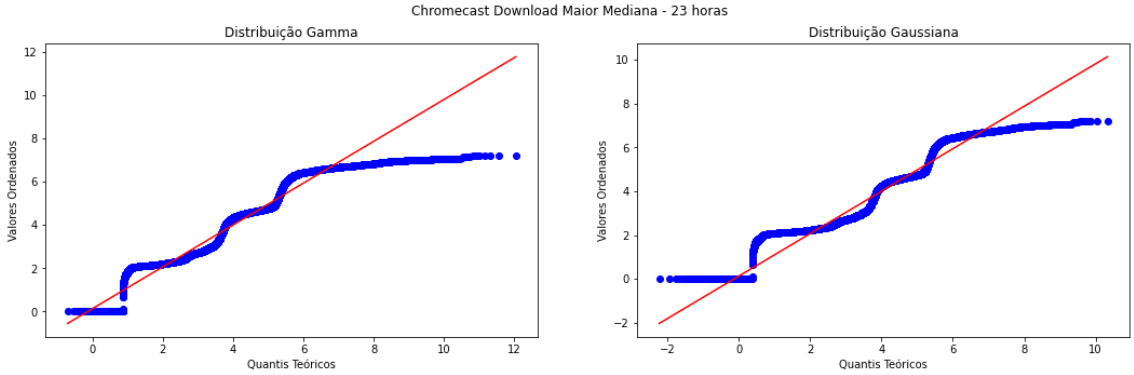

Figura 58 – Gráfico de Probabilidade – Chromecast, Mediana, Download – 23 Horas

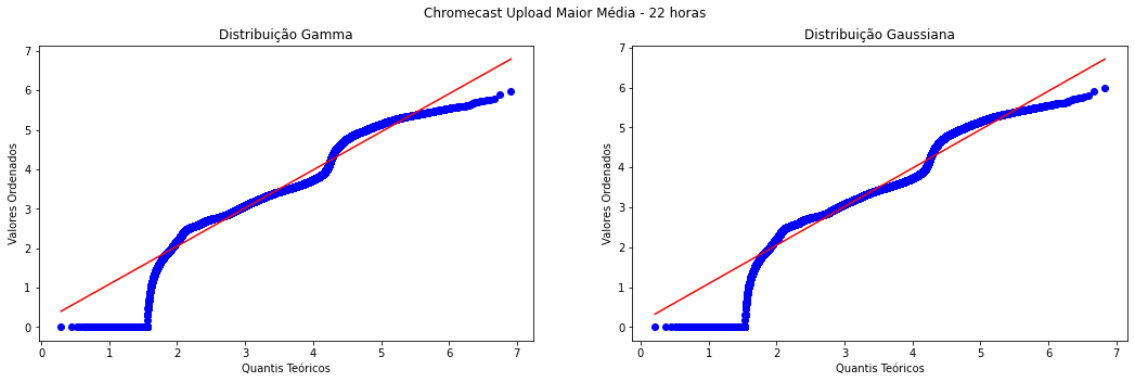

Figura 59 – Gráfico de Probabilidade – Chromecast, Média, Upload – 22 Horas

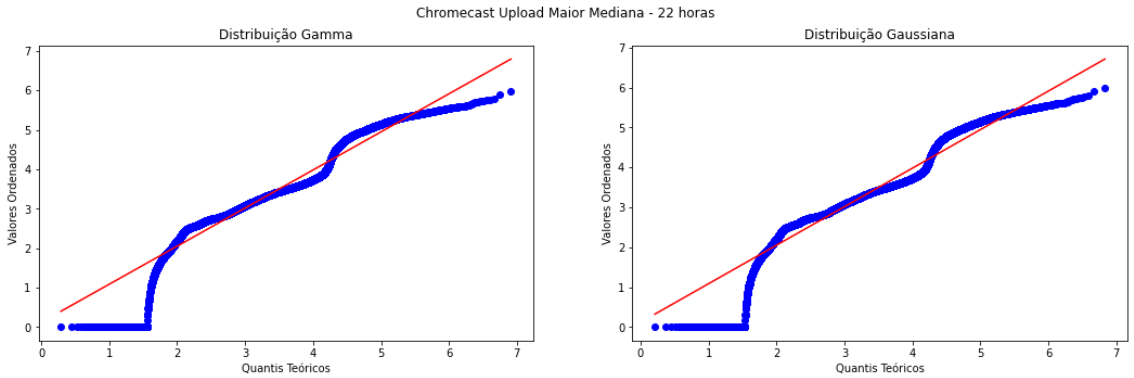

Figura 60 – Gráfico de Probabilidade – Chromecast, Mediana, Upload – 22 Horas

4.5 – Resultados

Como explicitado na seção 4.1, os horários encontrados para os datasets foram: Dataset 1: 20 horas; Dataset 2: 20 horas; Dataset 3: 20 horas; Dataset 4: 20 horas; Dataset 5: 22 horas; Dataset 6: 22 horas; Dataset 7: 23 horas; Dataset 8: 23 horas.

Observando os histogramas, um fato interessantíssimo é que as Smart Tvs têm uma quantidade de marcações de download e upload iguais a zero altíssima se comparadas com os Chromecasts,

isso se dá possivelmente por marcações realizadas quando o volume de tráfego é zero, algo que não ocorre no Chromecast.

Outro ponto é que nos histogramas de download, é possível perceber um padrão em que existem ‘picos’, as maiores densidades estão espaçadas em duas densidades para as Smart Tvs e três para os Chromecasts.

Já nos histogramas de upload, é possível perceber que nas Smart Tvs a densidade das medições cresce junto do volume de tráfego, enquanto no Chromecast existe um ponto central com densidade maior.

Os valores para os datasets 1 e 2, 3 e 4, 5 e 6 e 7 e 8, são os mesmos visto que se trata do mesmo horário e do mesmo aparelho, resultando no mesmo dataset.

Quanto a modelagens da literatura, observando as distribuições obtidas com o MLE e seus histogramas, os datasets5 e 6 parecem ter resultados satisfatórios com a gamma e com a gaussiana. Apesar de não ter sido testado dentro deste trabalho, é possível que com duas ou três variáveis aleatórias definidas dentro de dois ou três intervalos os outros datasets possam ser modelados também.

Através dos gráficos de probabilidade pode-se observar que para as variáveis aleatórias testadas, a Smart Tv não apresentou um bom resultado, enquanto o Chromecast poderia ser modelado com algumas perdas pela gaussiana e pela gamma -com resultados melhores na gaussiana.

Talvez o comportamento que os zeros possuem nas Smart TVs sejam um problema para modelagem.

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

5.1- Correlação

Para realizar a análise da correlação se faz necessário que o dispositivo e o horário sejam os mesmos, para que a quantidade de dados seja a mesma e a comparação seja coerente.

Por este motivo, a comparação entre os datasets 3 e 7 e os datasets 4 e 8 seriam inviáveis. Porém, os datasets 7 e 8 foram mantidos e comparados consigo mesmos, utilizando as colunas de upload e download.

A correlação foi calculada a partir do número de Pearson, fornecido pela função ‘pearsonr’ da biblioteca ‘scipy’.

Um número de Pearson igual a 1 representa correlação perfeita positiva, enquanto um número de Pearson igual a -1 representa uma correlação perfeita negativa e por fim, um número de Pearson igual a zero representa ausência de correlação linear.

As correlações obtidas foram:

Correlação amostral entre o dataset1 e o dataset3: 0.9156089964784074

Correlação amostral entre o dataset2 e o dataset4: 0.9156089964784074

Correlação amostral entre o dataset7 e o dataset7: 0.7925043015217014

Correlação amostral entre o dataset8 e o dataset8: 0.7925043015217014

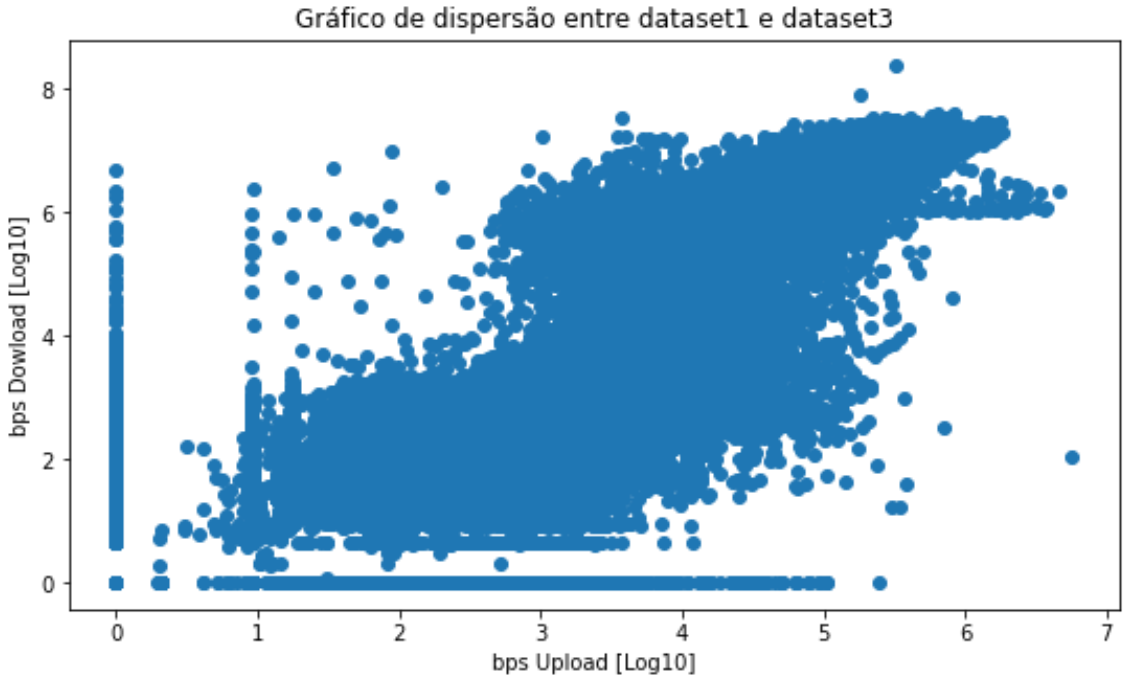

Figura 61 – Gráfico de Dispersão – Dataset1 e Dataset3

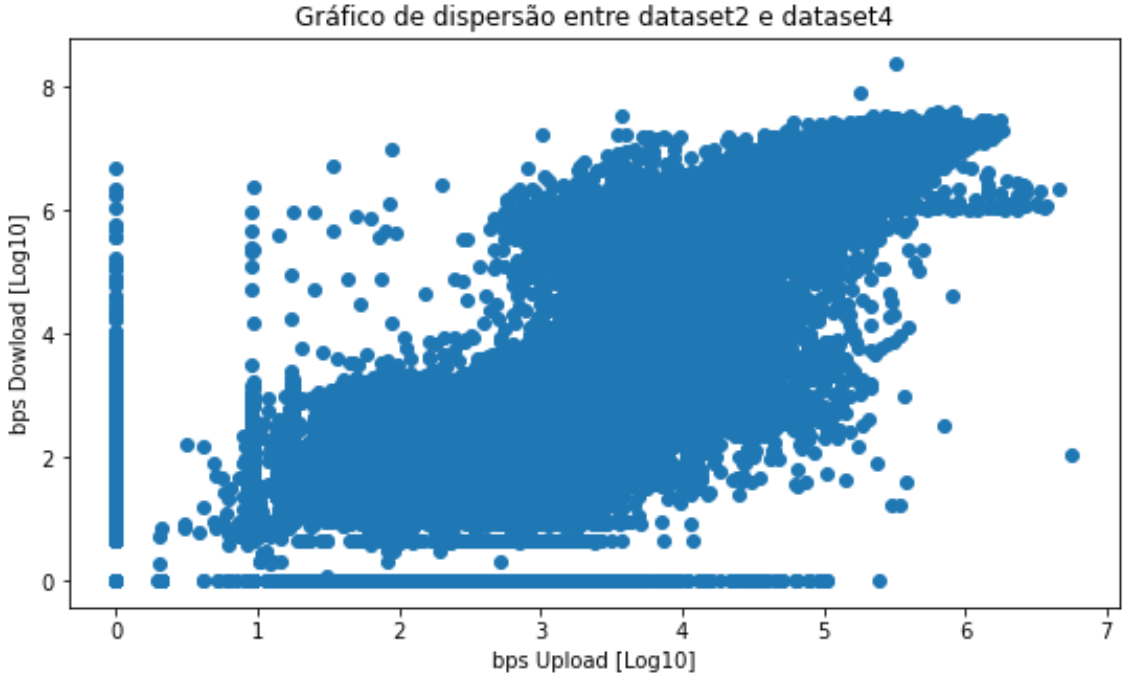

Figura 62 – Gráfico de Dispersão – Dataset2 e Dataset4

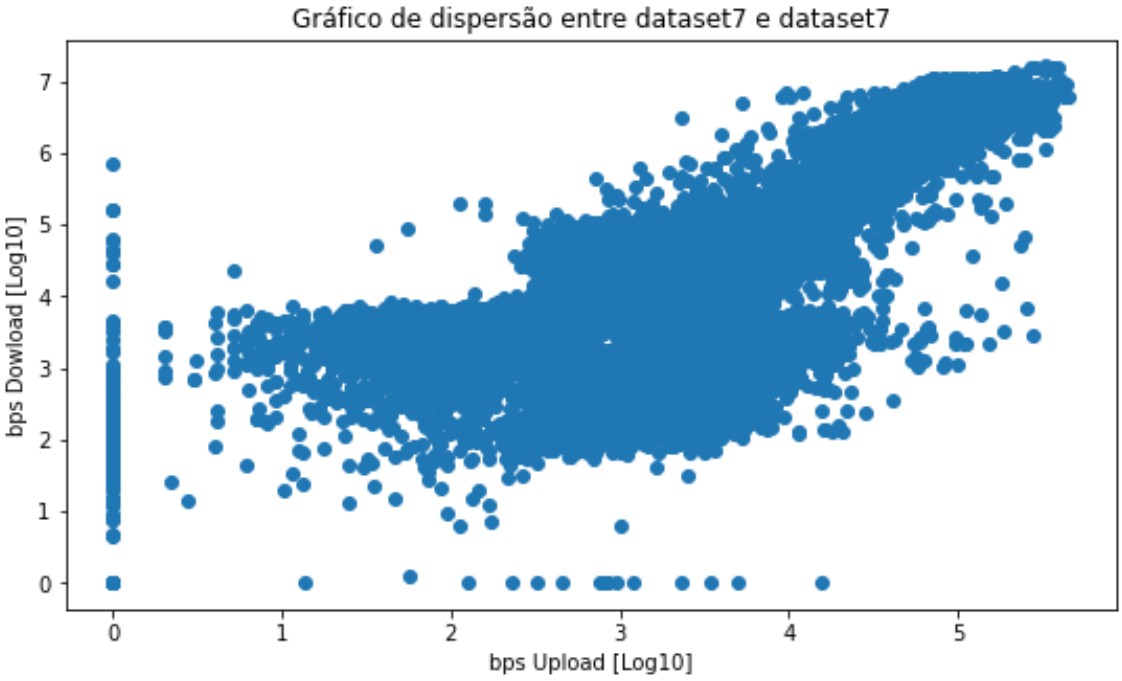

Figura 63 – Gráfico de Dispersão – Dataset7(upload) e Dataset7(download)

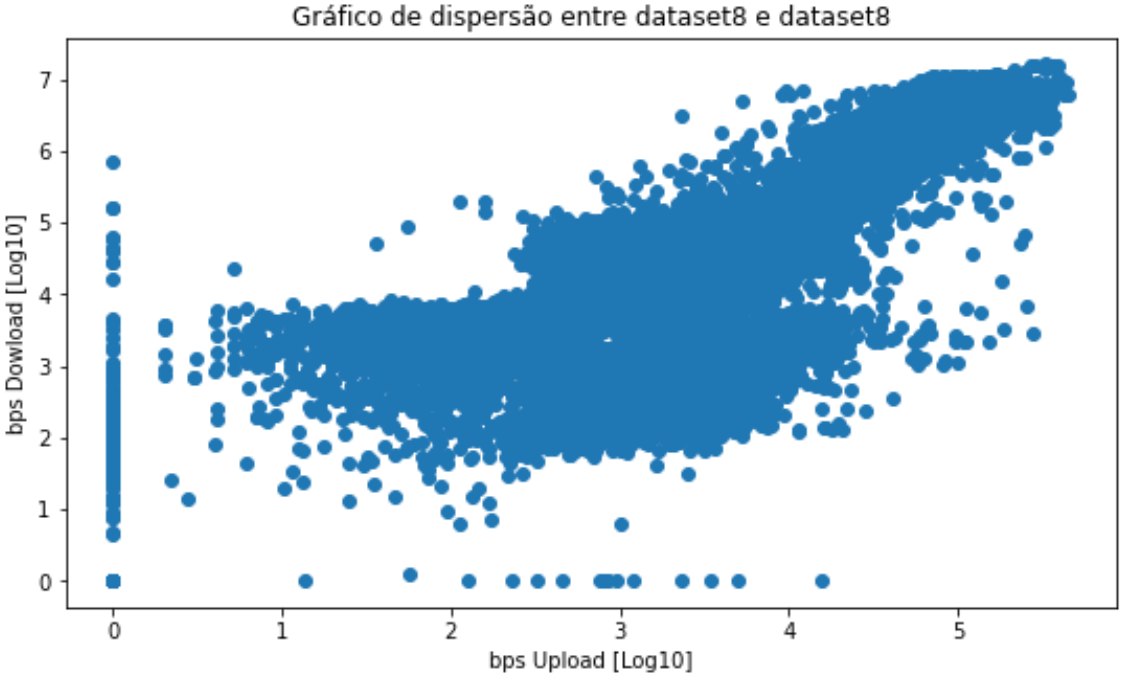

Figura 64 – Gráfico de Dispersão – Dataset8(upload) e Dataset8(download)

5.2- Resultados

É possível observar que os coeficientes de correlação entre os datasets de 1 a 4 – Correspondentes as Smart Tvs -, possuem uma alta correlação positiva. Com os datasets de 5 a 8 – Correspondentes ao Chromecast -, os valores de correlação ainda são suficientemente altos para dizer que há certa correlação positiva.

Isso reforça a ideia na seção 3.4, onde é dito que os fluxos de upload e download parecem estar conectados. Agora, isso é numericamente comprovado, quanto maior o fluxo de download, maior o fluxo de upload.

6- Comparação dos dados gerados pelos dispositivos Smart-TV e Chromecast

6.1- Teste-G

Para realização das comparações, a estatística utilizada será o teste-g, que compara uma distribuição observada com uma distribuição esperada.

No caso deste projeto, as duas distribuições serão ‘observadas’, visto que todas foram obtidas experimentalmente.

Antes da realização deste teste, se faz necessário que os dois parâmetros comparados tenham o mesmo tamanho. Isso será realizado de maneira semelhante ao que foi feito com os histogramas, categorias (bins) serão definidas a partir do método de sturges. Após isso, os valores de cada uma das bins será normalizada com a seguinte fórmula:

Valor de uma das bins / soma dos valores de todas as bins

Passaremos então estes dois ‘arrays’ para a função ‘power_divergence’ do ‘scipy’, com o parâmetro ‘lambda=log_likelihood’ que caracteriza o teste-g para esta função.

Com isso, os valores obtidos são:

Teste G do dataset 1 e do dataset 5: 1.7406537727017426

Teste G do dataset2 e do dataset6: 1.7406537727017426

Teste G do dataset3 e do dataset7: 2.3592793746997165

Teste G do dataset4 e do dataset8: 2.3592793746997165

6.2- Resultados

Com o resultado destes testes, poderíamos dizer que há pouca semelhança entre as distribuições. Entretanto, como o p-valor de todos os testes foi aproximadamente 1, podemos concluir pouca coisa destes testes já que possuem baixa significância.

7- Link para o código utilizado

https://github.com/GARCl-A/Modelos-Probabilisticos-Projeto-Final

8- Referências

Além dos materiais fornecidos em aula, foram consultadas as documentações das bibliotecas e alguns fóruns -utilizados principalmente para observar as argumentações dos usuários sobre o uso de determinadas técnicas. Todas as referências estavam funcionando até o dia 06/01/2023

https://stats.stackexchange.com/questions/1444/how-should-i-transform-non-negative-data-including-zeros

https://en.wikipedia.org/wiki/G-test#Distribution_and_use

https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.pearsonr.html

http://labtrop.ib.usp.br/doku.php?id=dicas_mat_apoio:analises_dados:anal_cat#teste_g

https://pt.wikipedia.org/wiki/Coeficiente_de_correla%C3%A7%C3%A3o_de_Pearson

https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.probplot.html

https://docs.analytica.com/index.php/Skewness

https://docs.analytica.com/index.php/Gamma_distribution#:~:text=To%20estimate%20the%20parameters%20of,)%2FMean(X%2C%20I)

https://en.wikipedia.org/wiki/Gamma_distribution#Maximum_likelihood_estimation]

https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.power_divergence.html#rf6c2a1ea428c-3

https://www.statstest.com/g-test/#G-Test_Example

https://numpy.org/doc/stable/reference/generated/numpy.linspace.html

https://stats.stackexchange.com/questions/362860/kl-divergence-between-which-distributions-could-be-infinity

https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.scatter.html

https://pandas.pydata.org/docs/reference/api/pandas.cut.html

https://cursos.alura.com.br/forum/topico-funcao-cut-e-categorizacao-intervalar-125799