DATASET: DRY BEAN

BRENDA BARROS ALVES DA SILVA

FREDERICO DE MORAES BEZERRA

FABIANA SILVA DE OLIVEIRA

WLADIMIR FARIAS TENÓRIO FILHO

YGOR PAES ALCANTARA

BREVE DESCRIÇÃO DO PROBLEMA

Contexto: 13.611 feijões foram coletados e a partir de um sistema visão computacional foram extraídas 16 características e o atributo de tipagem do feijão .

Problema: A partir das características como devemos classificar de modo automático os tipos de feijão.

Benefícios da solução: Garantir o padrão de qualidade do feijão por seu tipo.

BREVE DESCRIÇÃO DO PROBLEMA

Problema: Como classificar e avaliar sementes de feijão seco.

13.611 feijões foram catalogados de acordo com suas características de forma, cor e tipo. Imagens destes grãos foram registradas por meio de câmeras de alta resolução e foram catalogadas 16 características, tais como: área, perímetro, comprimento do eixo maior, comprimento do eixo menor, etc.

A problematização central do artigo detém-se em fornecer um método para a obtenção de variedades de sementes uniformes a fim de evitar que estas não sejam classificadas /certificadas como uma única variedade.

O banco disponibilizado não apresentou valores ausentes nos 17 atributos, no entanto foram verificados dados duplicados (redundância).

PARTES INTERESSADAS

Parte Interessada	Interesse	Influência	Classificação
Fazendeiro (Dono da Empresa)	ALTA	ALTA	PROMOTORES
Biólogos/Agrônomo	ALTA	BAIXA	DEFENSORES
Consumidor	BAIXA	ALTA	LATENTES
Distribuidoras de Fertilizante de Solo	ALTA	BAIXA	DEFENSORES
Empresas de Agrotóxicos e Pesticida	ALTA	BAIXA	DEFENSORES
Empresas de Tecnologias na Área de Plantio	ALTA	BAIXA	DEFENSORES
Empresas de Monitoramento de Pragas na Lavoura	ALTA	BAIXA	DEFENSORES

PARTES INTERESSADAS

- Fazendeiro (Dono da Empresa)
- Biólogos/Agrônomo
- Consumidor
- Distribuidoras de Fertilizante de Solo
- Empresas de Agrotóxicos e Pesticida
- Empresas de Tecnologias na Área de Plantio
- Empresas de Monitoramento de Pragas na Lavoura
- Empresas de Business e Marketing

BASES DE DADOS RELACIONADAS

BASES NACIONAIS	FONTE
Dados Climáticos (Umidade do Ar, precipitação e temperatura)	CPTEC -INPE
Dados de Solo (Concentração de nutrientes no solo)	IBGE
Dados de estrategia de marketing para comercialização de feijão	EPAGRI
Dados relacionados a adubação e fertilização de feijão	EMBRAPA
Dados sobre importação e exportação de feijão	MAPA-GOV

BASE INTERNACIONAL	FONTE
Dados anuais de produção, área plantada e rendimento por área plantada de feijão	Agriculture Organization of the United Nations (FAO)

BASES DE DADOS RELACIONADAS

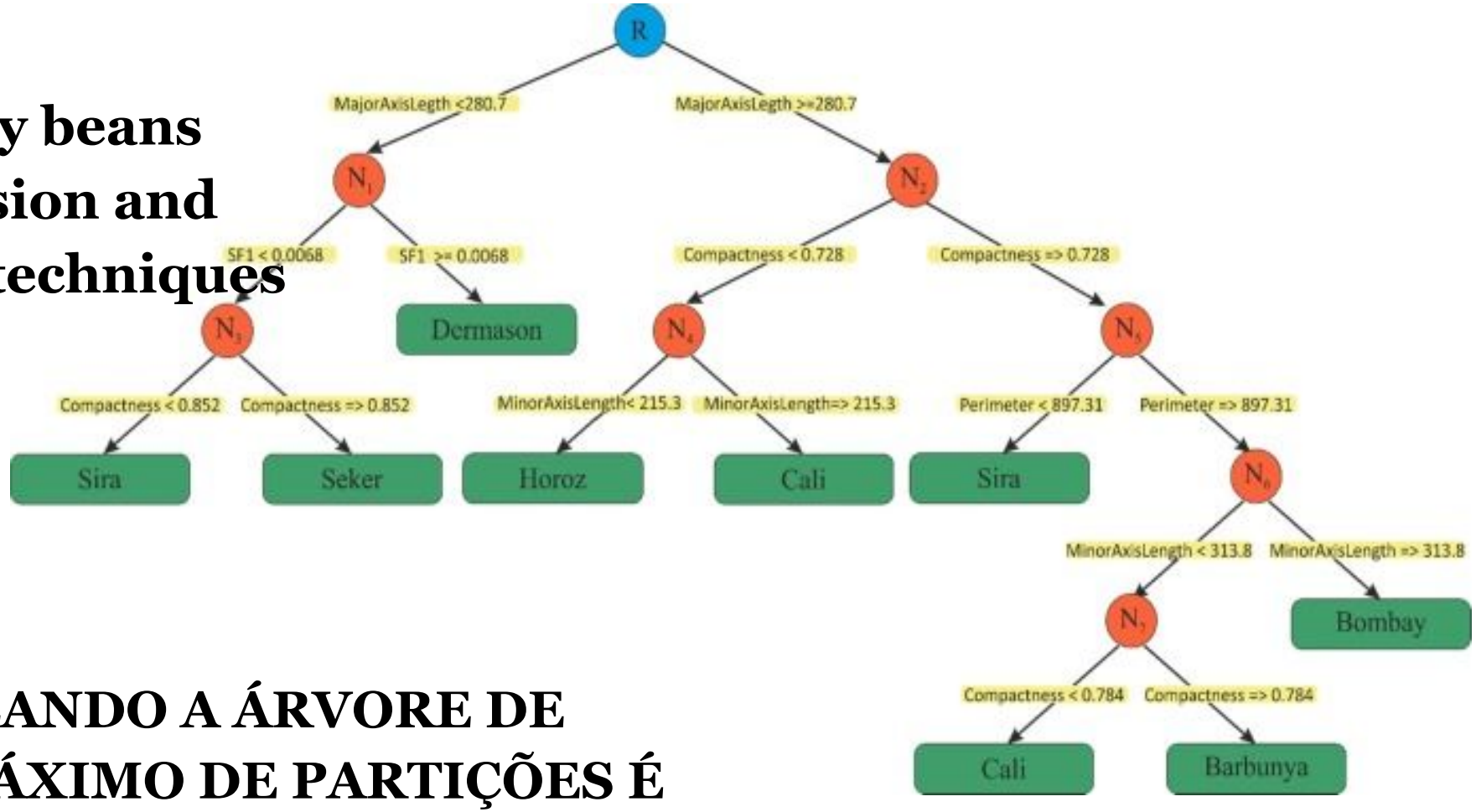
- Dados Climáticos (Umidade do Ar, precipitação e temperatura)

  Fonte: CPTEC - INPE (Instituto Nacional de Pesquisas Espaciais)
- Dados de Solo (Concentração de nutrientes no solo)

  Fonte: IBGE - Dados de área plantada;
- Dados de estratégia de marketing para comercialização de feijão. Fonte: Empresa de Pesquisa Agropecuária e Extensão Rural de Santa Catarina (Epagri);
- Dados relacionados a adubação e fertilização de feijão. Fonte: Agência Emprapa de Informação Tecnológica
- Dados anuais de produção, área plantada e rendimento por área plantada de feijão. Fonte: Agriculture Organization of the United Nations (FAO)
- Dados sobre importação e exportação de feijão. Fonte: Ministério da Agricultura, Pecuária e Abastecimento (Mapa) e Instituto Brasileiro do Feijão e Pulses (Ibrafe);
- Base de Dados da Pesquisa Agropecuária (BDPA);

DESCOBERTAS ANTERIORES

PAPER: Multiclass classification of dry beans using computer vision and machine learning techniques



ACHADO: UTILIZANDO A ÁRVORE DE DECISÃO O Nº MÁXIMO DE PARTIÇÕES É DEZESSEIS

DESCOBERTAS ANTERIORES

PAPER: Dry Beans Classification Using Machine Learning Grzegorz Słowiński

ACHADOS:

ALGORITMO	PRECISÃO
RANDON FORREST	93,61%
ARVORE DE DECISÃO	88,35%
NAIVE-BAYERS	64,30%

DESCOBERTAS ANTERIORES

- Importância do Uso de Ureia Enriquecida com Selênio em Biofortificação de Dry Beans. NAMORATO, Felipe. 2019. LAVRAS- MG.
- Importância da Classificação de Dry Beans Utilizando Técnicas de Machine Learning e Computer Vision. KOKLU, M. and OZKAN, I.A., (2020)
- THE ECONOMIC IMPACT OF THE SOUTH AFRICAN AGRICULTURAL RESEARCH COUNCIL'S DRY BEANS BREEDING PROGRAM. Cambridge University, 2017.
- Impacto Ambiental do Plantio de Dry Beans, análise do impacto em solo e água. Agencia Embrapa de Informação Tecnologica (AGEITEC).

DICIONÁRIO DOS DADOS

Campo	Descrição	Tipo	Tamanho	Valores permitidos
Area	área geométrica (m2)	Inteiro	5	
Perimeter	perímetro (m)	Real	7	
MajorAxisLength	comprimento (cm)	Real	15	
MinorAxisLength	comprimento (cm)	Real	15	
AspectRation	relação entre comprimento maior e menor	Real	12	
Eccentricity	relação entre distâncias de focos na elipse	Real	11	
ConvexArea	número de pixels no menor polígono	Inteiro	6	
EquivDiameter	diâmetro equivalente em um círculo	Real	20	

DICIONÁRIO DOS DADOS

Campo	Descrição	Tipo	Tamanho	Valores permitidos
Extent	razão de pixels na área	Real	17	
Solidity	proporção dos pixels na concha convexa	Real	17	
roundness	calculado por (4piA)/(P^2)	Real	17	
Compactness	arredodamento do objeto	Real	17	
ShapeFactor1	fator 1 da forma	Real	20	
ShapeFactor2	fator 2 da forma	Real	17	
ShapeFactor3	fator 3 da forma	Real	17	
ShapeFactor4	fator 4 da forma	Real	17	
Class	7 classes de feijão	Caractere	8	SEKER, BARBUNYA, BOMBAY, CALI, HOROZ, SIRA, DERMASON

PRINCIPAIS ATRIBUTOS

Foram 16 atributos: 14 atributos de Dimensão (métricas ou derivações de métricas) e 04 atributos de forma (não especificado com o termo: fator de forma 01 a 04)

Total	Atributos	Informações de atributos:
1	Area	Área (A): zona de feijão e o número de pixels dentro de seus limites.
2	Perimeter	Perímetro (P): circunferência / comprimento de sua borda
3	Majoraxislength	Comprimento do eixo principal (L): A distância / extremidades mais longa do feijão
4	Minoraxislength	Comprimento do eixo menor (l): A linha mais longa extraída/desenhada do feijão em perpendicular ao eixo principal.
5	Aspectration	Proporção (K): define a relação entre L e l. (L/i)
6	Eccentricity	Excentricidade (ce): excentricidade da elipse tendo os mesmos momentos que a região.
7	Convexarea	Área convexa (C): número de pixels no menor polígono convexo que pode conter a área de uma semente de feijão.
8	Equivdiameter	Diâmetro equivalente (ed): diâmetro de um círculo com a mesma área que uma área de semente de feijão.
9	Extent	Extensão (ex): A proporção dos pixels na caixa delimitadora para a área do feijão
10	Solidity	Solidez (S): convexidade. A proporção dos pixels na concha convexa em relação aos encontrados em feijões.
11	Roundness	Arredondamento (R): calculado com a seguinte fórmula: (4PIA)/(P^2)
12	Compactness	Compactação (CO): mede o arredondamento de um objeto: ed/L

PRINCIPAIS ATRIBUTOS

Total	Atributos	Informações de atributos:
13	Shapefactor1	É um atributo com rotulagem de: "fator de forma 1" não especificado em artigo.
14	Shapefactor2	É um atributo com rotulagem de: fator de forma 2" não especificado em artigo
15	Shapefactor3	É um atributo com rotulagem de: fator de forma 3" não especificado em artigo
16	Shapefactor4	É um atributo com rotulagem de: "fator de forma 4" não especificado em artigo
17	Class	É um atributo que classifica o feijão em 07 tipos: seker, barbunya, bombay, cali, dermosan, horoz e sira

PRÉ-PROCESSAMENTO: REDUÇÃO

...

1 - REDUÇÃO: REMOÇÃO DE REDUNDÂNCIA

```
[144] df.shape
0s

(13611, 17)
```

```
[148] df_duplicates.shape
0s

(13543, 17)
```

Diferença de 68 itens.
Menos de 0,05% de diferença.

Mantivemos amostragem inicial.

PRÉ-PROCESSAMENTO: REDUÇÃO

...

2 - REDUÇÃO: EXCLUSÃO / DIMENSIONAMENTO

```
df2 = df.drop(columns=['ShapeFactor1', 'ShapeFactor2', 'ShapeFactor3', 'ShapeFactor4', 'Class'])
df2
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	AspectRation	Eccentricity	ConvexArea	EquivDiameter	Extent	Solidity	roundness	Compactness
0	28395	610.291	208.178117	173.888747	1.197191	0.549812	28715	190.141097	0.763923	0.988856	0.958027	0.913358
1	28734	638.018	200.524796	182.734419	1.097356	0.411785	29172	191.272750	0.783968	0.984986	0.887034	0.953861
2	29380	624.110	212.826130	175.931143	1.209713	0.562727	29690	193.410904	0.778113	0.989559	0.947849	0.908774
3	30008	645.884	210.557999	182.516516	1.153638	0.498616	30724	195.467062	0.782681	0.976696	0.903936	0.928329
4	30140	620.134	201.847882	190.279279	1.060798	0.333680	30417	195.896503	0.773098	0.990893	0.984877	0.970516
...	...	...	...	...	...	...	...	...	...	...	...	...
13606	42097	759.696	288.721612	185.944705	1.552728	0.765002	42508	231.515799	0.714574	0.990331	0.916603	0.801865
13607	42101	757.499	281.576392	190.713136	1.476439	0.735702	42494	231.526798	0.799943	0.990752	0.922015	0.822252
13608	42139	759.321	281.539928	191.187979	1.472582	0.734065	42569	231.631261	0.729932	0.989899	0.918424	0.822730
13609	42147	763.779	283.382636	190.275731	1.489326	0.741055	42667	231.653248	0.705389	0.987813	0.907906	0.817457
13610	42159	772.237	295.142741	182.204716	1.619841	0.786693	42600	231.686223	0.788962	0.989648	0.888380	0.784997

13611 rows × 12 columns

PRÉ-PROCESSAMENTO: REDUÇÃO

1 - REDUÇÃO: REMOÇÃO DE REDUNCIA

```
[144] df.shape

(13611, 17)
```

```
[148] df_duplicates.shape

(13543, 17)
```

2 - REDUÇÃO: REDUÇÃO POR SEGMENTAÇÃO: FILTRO TAMANHO*

```
[156] #Segmentação do conjunto de dados pelo atributo "MajorAxisLength"
      df = df[df['MajorAxisLength'] > 183]
      df = df[df['MajorAxisLength'] < 739]

      df.shape

(548, 17)
```

3 - REDUÇÃO: EXCLUSÃO / DIMENSIONAMENTO

```
[157] df2 = df.drop(columns=['AspectRation','ConvexArea','EquivDiameter','Extent','Solidity','rour
      df2
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	Eccentricity	Class
3344	100846.0	1297.770	469.285655	274.423910	0.811200	BARBUNYA
3345	102015.0	1271.970	456.791895	286.894421	0.778162	BARBUNYA
3346	102379.0	1296.377	456.722068	286.557574	0.778679	BARBUNYA
3347	105542.0	1265.623	466.135980	288.999342	0.784610	BARBUNYA
3348	115967.0	1359.763	449.454969	331.305270	0.675755	BARBUNYA
...	...	...	...	...	...	...
5496	106806.0	1263.899	494.727002	276.176469	0.829679	CALI
5497	107911.0	1298.822	498.597779	279.350337	0.828309	CALI
5498	114858.0	1300.819	512.736642	287.561719	0.827926	CALI
5499	115608.0	1298.623	500.298310	296.898826	0.804876	CALI
5500	116272.0	1326.583	534.484404	279.783414	0.852048	CALI

548 rows × 6 columns

* Majoraxislength - extremidades mais longa do feijão

PRÉ-PROCESSAMENTO: REDUÇÃO

4 - REDUÇÃO: *AMOSTRAGEM* DOS DADOS

```
dfsample = df.sample(n=10, replace=False, random_state=123)
dfsample
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	AspectRation
3824	205358.0	1737.704	653.993204	404.064330	1.618537
3787	197245.0	1702.646	642.077245	396.085390	1.621058
3767	191584.0	1634.627	580.635582	421.028013	1.379090
3654	176570.0	1601.037	610.491266	369.668142	1.651458
3788	197598.0	1687.798	641.680640	395.192761	1.623716
3437	151014.0	1456.767	532.740751	362.552358	1.469417
3516	161579.0	1502.815	558.604121	368.791025	1.514690
3499	159369.0	1510.578	570.150632	358.579133	1.590027
3540	164596.0	1550.902	583.920422	360.617230	1.619225
3768	191756.0	1699.140	659.992234	373.001008	1.769411

5 - REDUÇÃO: *AMOSTRAGEM COM FILTRO*

```
[159] df_resample = resample(
          df[df['MajorAxisLength'] > 500],
          replace=True,
          n_samples= 10)

      df_resample
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	Aspec
3861	226806.0	1812.548	679.975387	428.680341	
3399	144083.0	1416.894	508.239887	362.822978	
3848	217182.0	1811.572	704.304209	396.825746	
3828	206702.0	1735.169	663.537080	401.759941	
3422	148325.0	1454.931	549.733694	344.949850	
3649	176276.0	1616.130	621.159964	365.522552	
3622	172941.0	1635.318	643.132308	347.714665	
3822	204635.0	1706.370	635.126287	412.663916	
3816	203677.0	1728.486	662.550897	393.518855	
3692	181877.0	1596.933	596.673548	389.934948	

PRÉ-PROCESSAMENTO: TRANSFORMAÇÃO

1 – TRANSFORMAÇÃO POR CODIFICAÇÃO - LETRA / NÚMERO

	Area	Perimeter	MajorAxisLength	MinorAxisLength	Class
3344	100846.0	1297.770	469.285655	274.423910	BARBUNYA
3345	102015.0	1271.970	456.791895	286.894421	BARBUNYA
3346	102379.0	1296.377	456.722068	286.557574	BARBUNYA
3347	105542.0	1265.623	466.135980	288.999342	BARBUNYA
3348	115967.0	1359.763	449.454969	331.305270	BARBUNYA

```
cleanup_nums = {"Class": {'SEKER':1,'BARBUNYA':2,'BOMBAYER':3,'CALI':4
                          6:'SIRA',7:'DERMASON'}}
df_replaced = df.replace(cleanup_nums)
df_replaced.head()
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	Class
3344	100846.0	1297.770	469.285655	274.423910	2
3345	102015.0	1271.970	456.791895	286.894421	2
3346	102379.0	1296.377	456.722068	286.557574	2
3347	105542.0	1265.623	466.135980	288.999342	2
3348	115967.0	1359.763	449.454969	331.305270	2

PRÉ-PROCESSAMENTO: TRANSFORMAÇÃO

2 – TRANSFORMAÇÃO POR NORMALIZAÇÃO – 1 (VALOR MÁX) 0 (VALOR MÍN)

```
df_normalized.head()
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	AspectRation	Eccentricity	ConvexArea	EquivDiameter	Extent	Solidity	roundness	Compactness	Clas
0	1.000000	1.000000	1.000000	0.962154	0.445781	0.833037	1.000000	1.000000	0.734794	0.635223	0.642855	0.375037	0.33333
1	0.925934	0.956399	0.998711	0.859998	0.542710	0.880821	0.906875	0.951641	0.735976	0.883221	0.634208	0.300075	0.33333
2	0.986405	0.955155	0.964459	0.972542	0.405501	0.808740	0.975942	0.991250	0.680203	0.761928	0.733619	0.421705	0.33333
3	0.943236	0.938773	0.977512	0.902784	0.480112	0.851510	0.921913	0.963092	0.854055	0.907939	0.706537	0.353433	0.33333
4	0.973561	0.930980	0.957124	0.958083	0.411426	0.812510	0.949793	0.982932	0.811315	0.930219	0.777089	0.420938	0.33333

PRÉ-PROCESSAMENTO: LIMPEZA**

1 - LIMPEZA: INSERIR MÉDIA

```
df_fill = df.fillna(df.mean())
df_fill
```

	Area	Perimeter	MajorAxisLength
3344	100846.0	1297.770	469.285655
3345	102015.0	1271.970	456.791895
3346	102379.0	1296.377	456.722068
3347	105542.0	1265.623	466.135980
3348	115967.0	1359.763	449.454969
...	...	...	...
5496	106806.0	1263.899	494.727002
5497	107911.0	1298.822	498.597779
5498	114858.0	1300.819	512.736642
5499	115608.0	1298.623	500.298310
5500	116272.0	1326.583	534.484404

548 rows × 17 columns

2 - LIMPEZA: INSERIR MEDIANA

```
df_fill = df.fillna(df.median())
df_fill
```

	Area	Perimeter	MajorAxisLength
3344	100846.0	1297.770	469.285655
3345	102015.0	1271.970	456.791895
3346	102379.0	1296.377	456.722068
3347	105542.0	1265.623	466.135980
3348	115967.0	1359.763	449.454969
...	...	...	...
5496	106806.0	1263.899	494.727002
5497	107911.0	1298.822	498.597779
5498	114858.0	1300.819	512.736642
5499	115608.0	1298.623	500.298310
5500	116272.0	1326.583	534.484404

548 rows × 17 columns

2 - LIMPEZA: INSERIR MODA

```
df['MajorAxisLength'] = df['MajorAxisLength'].
fillna(df['MajorAxisLength'].mode()[0])
df['MajorAxisLength']
```

```
3344    469.285655
3345    456.791895
3346    456.722068
3347    466.135980
3348    449.454969
           ...
5496    494.727002
5497    498.597779
5498    512.736642
5499    500.298310
5500    534.484404
Name: MajorAxisLength, Length: 548, dtype: float64
```

** **NOTA::** A base não apresenta dados Ausentes para substituição neste processo de limpeza

PRÉ-PROCESSAMENTO: TRANSFORMAÇÃO

3 – TRANSFORMAÇÃO POR PARTIÇÃO: DIVIDIR O DATA SET EM BASE DE TREINO (70%) E BASE DE TESTE (30%

```
X_train
```

	Area	Perimeter	MajorAxisLength	Mi
2440	64036	1028.773	354.632118	
11453	32631	671.953	237.031776	
3505	58238	971.303	397.202654	
10575	34951	694.086	257.090664	
10351	37350	699.848	238.172991	
...	...	...	...	
804	92169	1153.456	398.911407	
633	80217	1194.350	398.653515	
2091	73935	1048.394	397.442148	
8677	37581	741.224	272.164033	
4424	53931	915.794	364.579544	

4083 rows × 16 columns

```
[92] X_test
```

	Area	Perimeter	MajorAxisLength	MinorAxisLength	A
3442	64412	974.540	327.994821	250.790882	
12835	28352	622.517	224.685842	161.792561	
9640	38819	717.192	259.095101	191.207899	
8295	40649	750.717	279.021528	185.998767	
11753	32109	663.582	224.472352	182.364496	
...	...	...	...	...	
2053	77502	1050.945	387.973825	256.218984	
12515	29938	637.143	230.810416	165.463577	
4959	50654	879.938	357.233899	181.407743	
573	88455	1228.134	441.980030	257.483197	
10111	36907	706.067	257.402411	182.887609	

9528 rows × 16 columns

ATRIBUTOS DE ANÁLISE: NUMÉRICO E NOMINAL

TAMANHO POR VARIEDADES

TAMANHO DO GRÃO: MajorAxisLength -distância / extremidades mais longa do feijão

VARIEDADES DO GRÃO: Class – Tipos: Seker, Barbunya, Bombayer, Cali, Horoz, Sira, Dermason

.

ATRIBUTO	MAJORAXISLENGTH	CONTRUÇÃO DA ESCALA - TAMANHO DO FEIJÃO	
Valor Mínimo -Vmin	183,6	Amplitude Total (At = Vmax - Vmin)	555,3
Valor Máximo - Vmax	738,9	Amplitude do intervalo - (Ai = At / Tc)	111,06
		ESCALA: total de classes: Tc=05	

CLASSE 1	183,6	294,7
CLASSE 2	294,7	405,7
CLASSE 3	405,7	516,8
CLASSE 4	516,8	627,8
CLASSE 5	627,8	738,9

ATRIBUTOS DE ANÁLISE: NUMÉRICO E NOMINAL

ACHADOS: A Quantidade de feijão, nesta base, é inversamente proporcional ao tamanho da área.

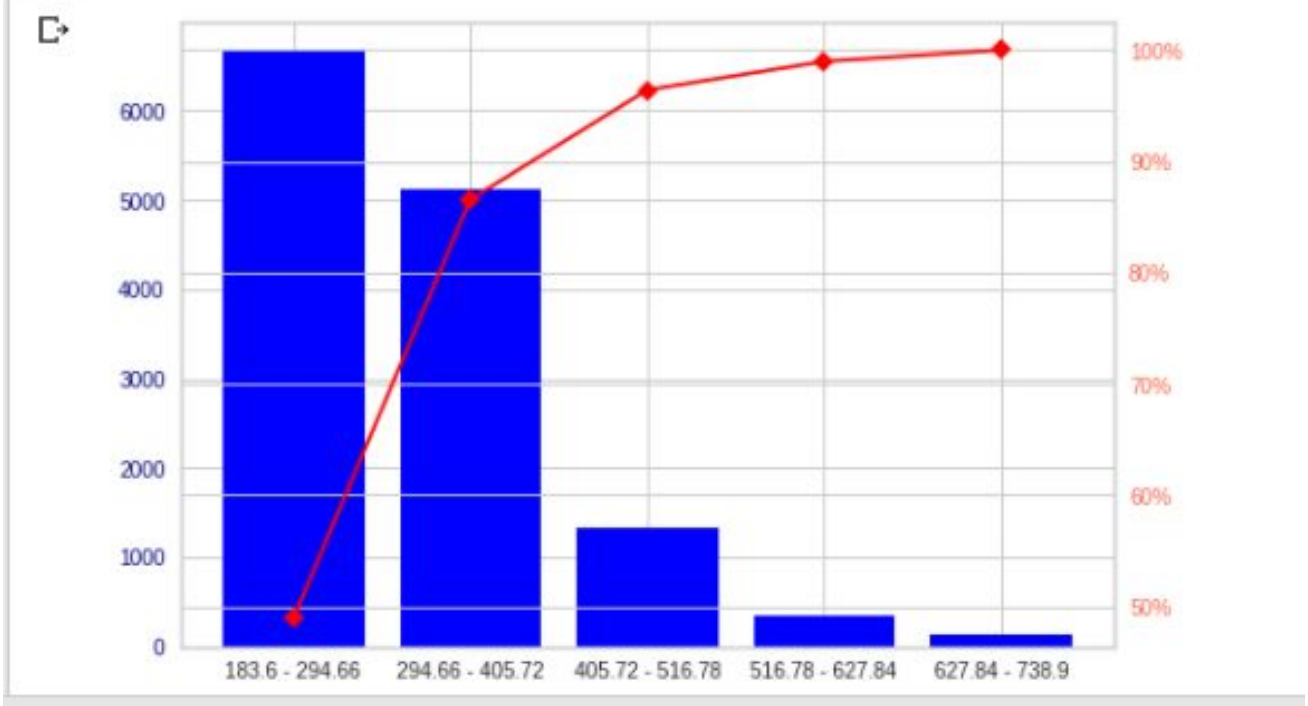
VARIEDADE	TOTAL	TAMANHO MÁX
BOMBAY	522	738,1445017
CALI	1.630	534,4844042
BARBUNYA	1.322	483,6912557
HOROZ	1.928	456,7581544
SIRA	2.636	400,9314668
SEKER	2.027	339,931533
DERMASON	3.546	308,2623358
TOTAL	13.611	

DISTRIBUIÇÃO DE FREQUÊNCIA:

```
[49] df['MajorAxisLength'].min()

[50] df['MajorAxisLength'].max()

     738.8601534818813
```

	frq_absoluta	frq_acumulada
627.84 - 738.9	140	1.028580
516.78 - 627.84	348	3.585335
405.72 - 516.78	1339	13.422967
294.66 - 405.72	5121	51.046947
183.6 - 294.66	6663	100.000000

```
fig, ax = plt.subplots()
ax.bar(df2.index, df2["frq_absoluta"], color="blue")
ax2 = ax.twinx()
ax2.plot(df2.index, df2["frq_acumulada"], color="red", marker="D", ms=7)
ax2.yaxis.set_major_formatter(PercentFormatter())
ax.tick_params(axis="y", colors="navy")
ax2.tick_params(axis="y", colors="tomato")

plt.show()
```



HISTOGRAMA

```
plt.figure(figsize=(10,6))
df[df['Class'] == 'SEKER']['MajorAxisLength'].hist(alpha=0.5, color= 'blue', bins= 30, label='SEKER')
df[df['Class'] == 'BARBUNYA']['MajorAxisLength'].hist(alpha=0.7, color= 'red', bins= 30, label='BARBUNYA')
df[df['Class'] == 'BOMBAY']['MajorAxisLength'].hist(alpha=0.7, color= 'orange', bins= 30, label='BOMBAY')
df[df['Class'] == 'CALI']['MajorAxisLength'].hist(alpha=0.7, color= 'pink', bins= 30, label='CALI')
df[df['Class'] == 'HOROZ']['MajorAxisLength'].hist(alpha=0.7, color= 'yellow', bins= 30, label='HOROZ')
df[df['Class'] == 'SIRA']['MajorAxisLength'].hist(alpha=0.7, color= 'purple', bins= 30, label='SIRA')
df[df['Class'] == 'DERMASON']['MajorAxisLength'].hist(alpha=0.7, color= 'gray', bins= 30, label='DERMASON')
plt.legend()
plt.xlabel('"Tamanho do Feijão Seco"')
plt.ylabel('Quantidade')
plt.show()
```

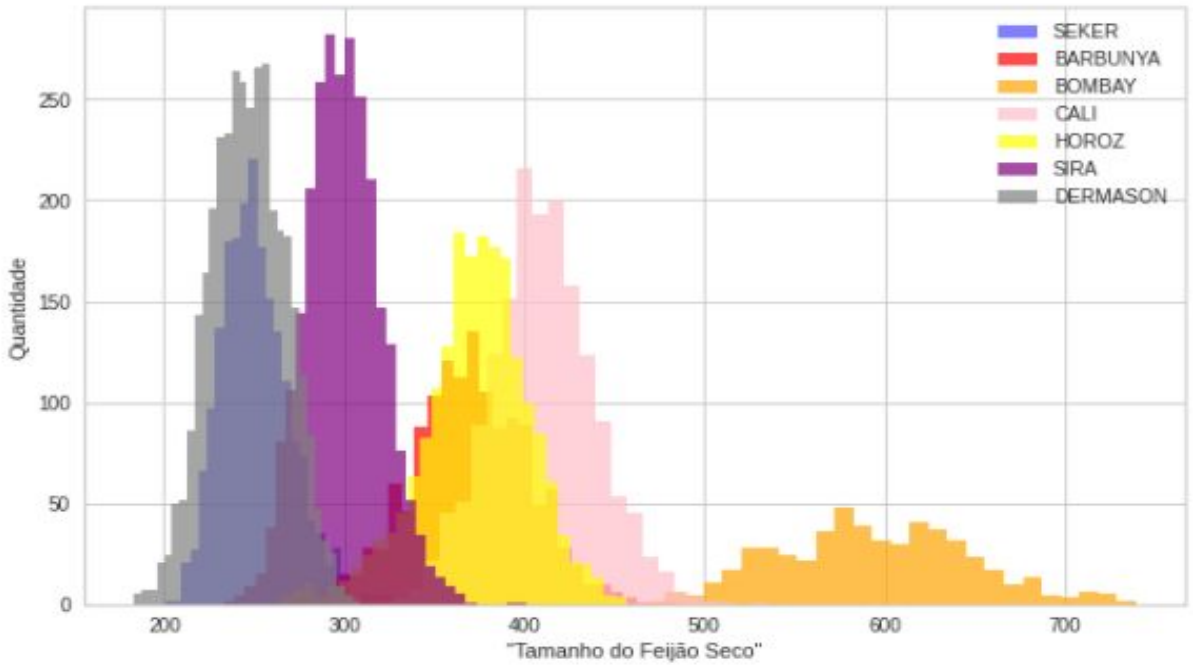
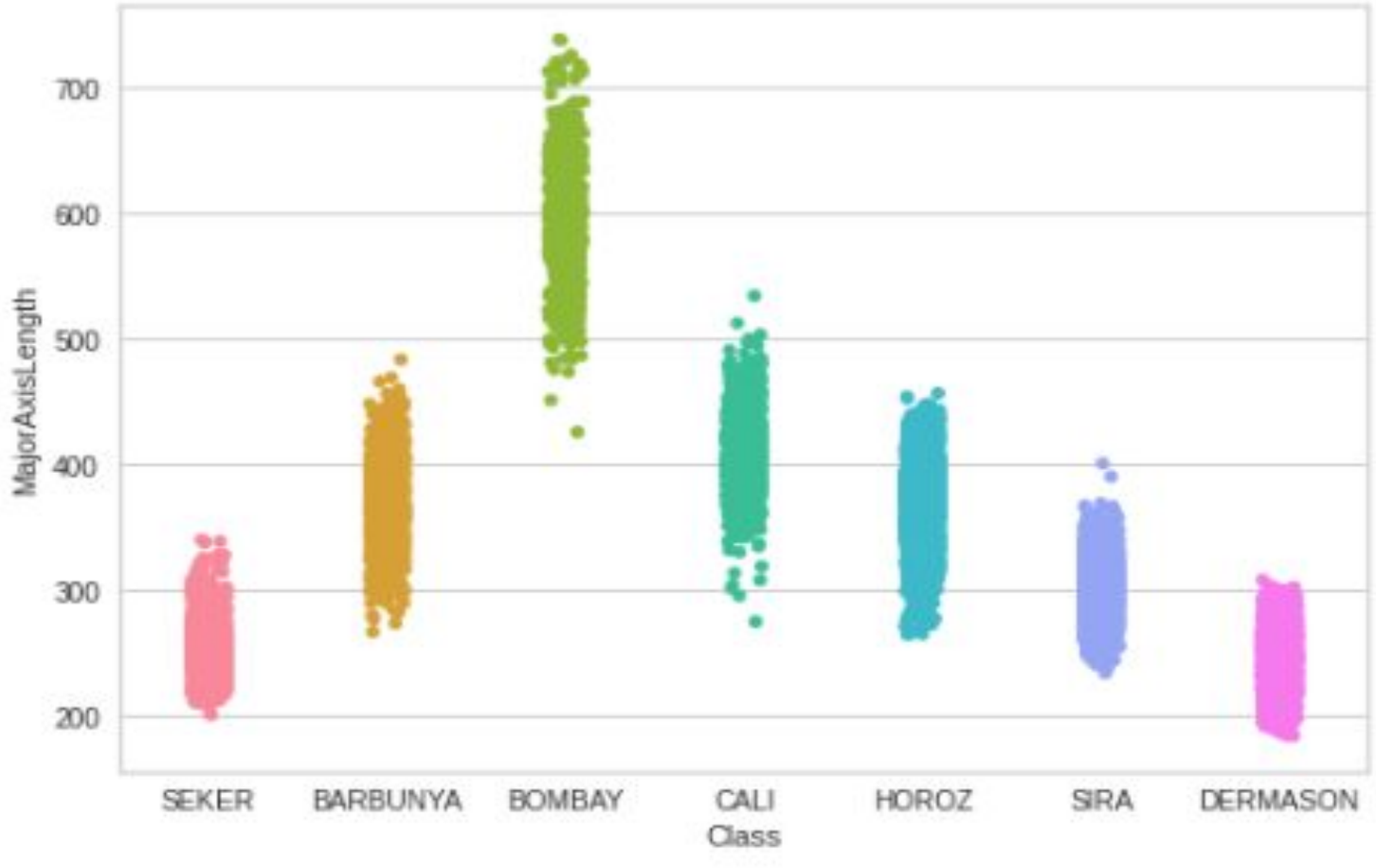


GRÁFICO DE DISPERSÃO

```
sns.stripplot (x='Class',y='MajorAxisLength',data=df_filter)
```

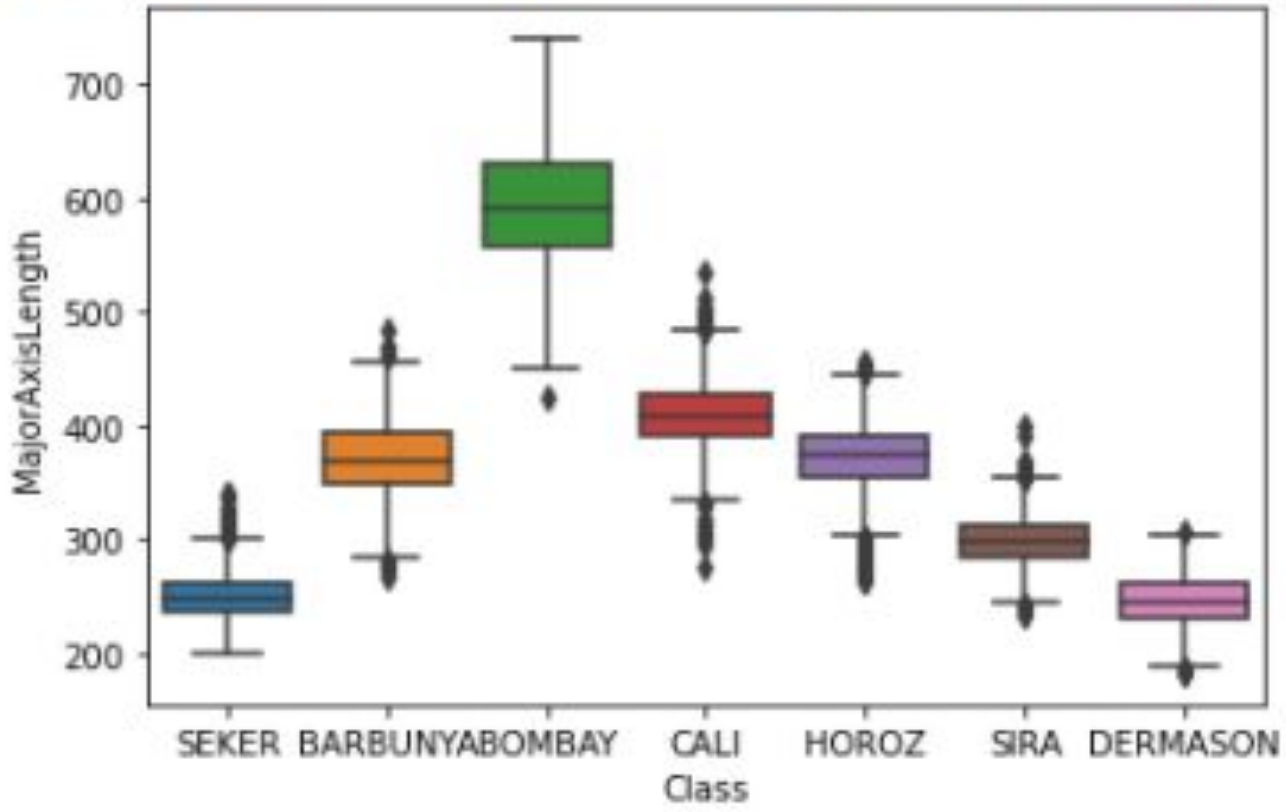
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fabaf6ee190>
```



BOX-PLOT

```
sns.boxplot(x='Class', y='MajorAxisLength', data=df_filter)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fabcfe68550>
```

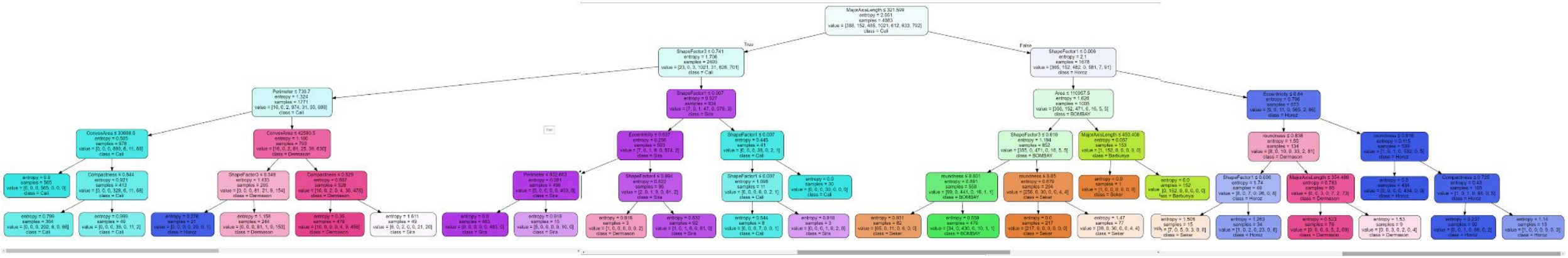


ÁRVORE DA DECISÃO

```
[35] # Fazendo a predição nos dados de treino
     resultado_dtc = dtc.predict(X_test)
     print(classification_report(y_test, resultado_dtc))
```

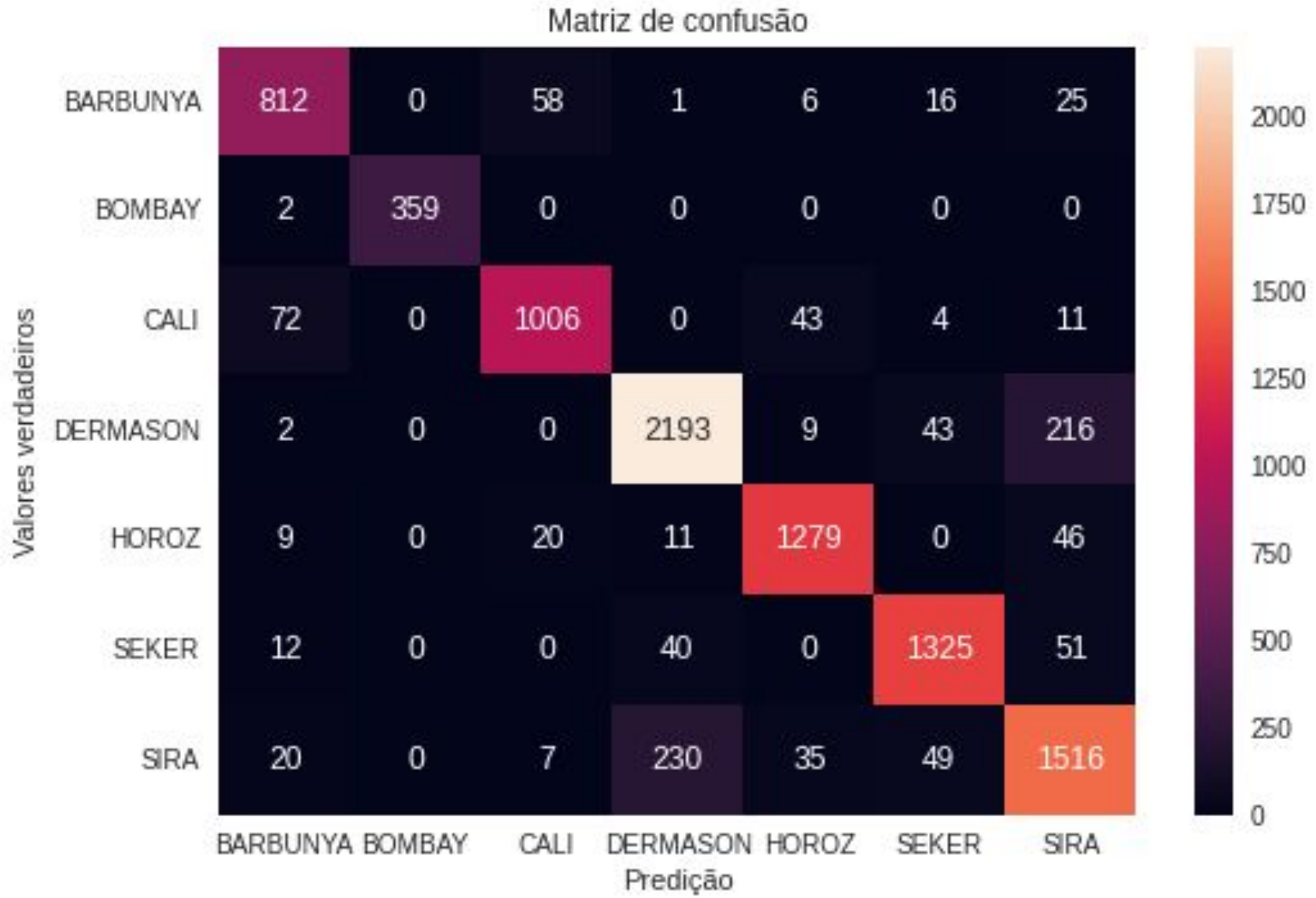
```
              precision    recall  f1-score   support

    BARBUNYA       0.87      0.88      0.88       918
      BOMBAY       1.00      0.99      1.00       361
        CALI       0.92      0.89      0.90      1136
    DERMASON       0.89      0.89      0.89      2463
       HOROZ       0.93      0.94      0.93      1365
       SEKER       0.92      0.93      0.92      1428
        SIRA       0.81      0.82      0.81      1857

    accuracy                           0.89      9528
   macro avg       0.91      0.91      0.91      9528
weighted avg       0.89      0.89      0.89      9528
```

ÁRVORE DA DECISÃO

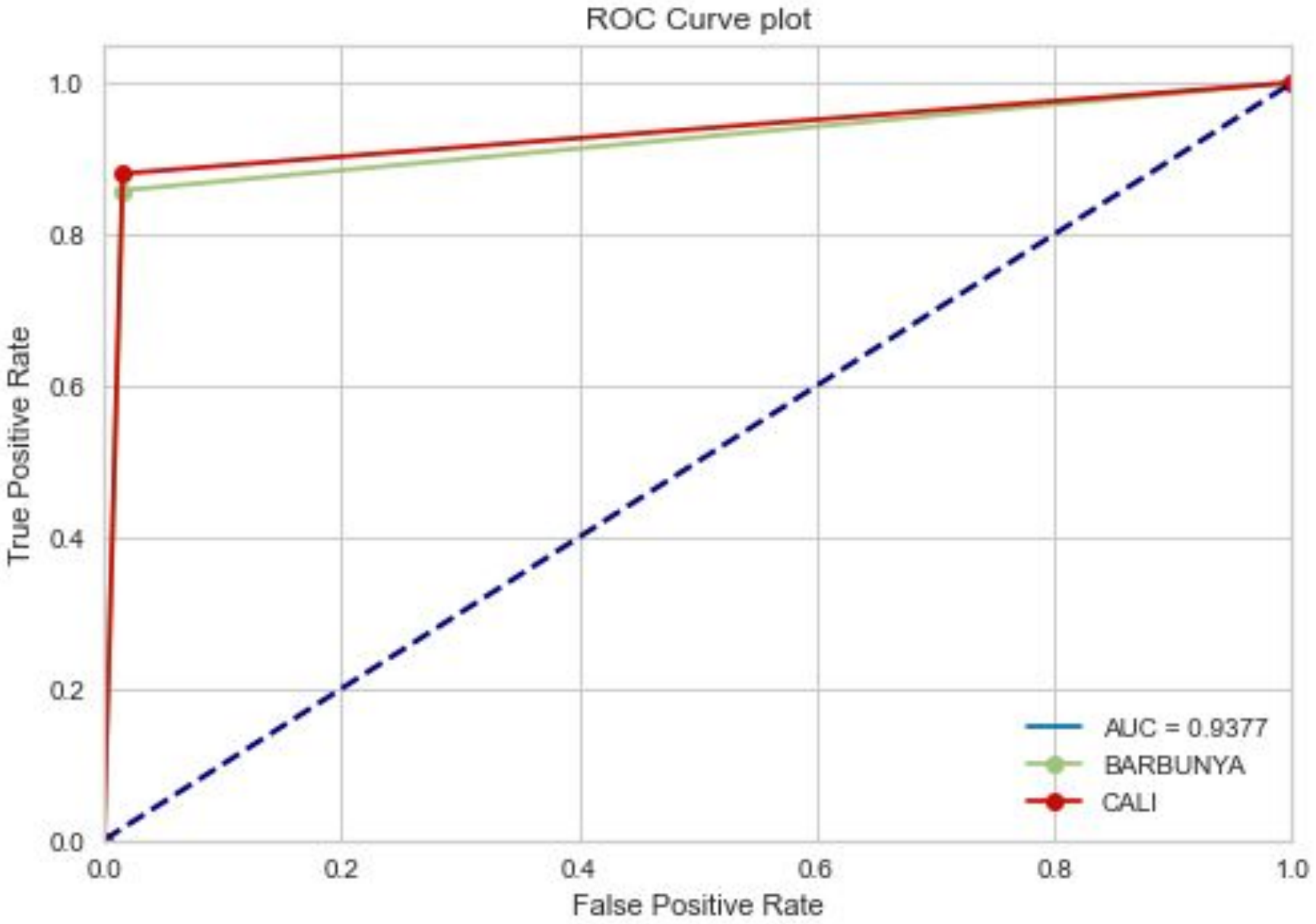


ÁRVORE DA DECISÃO:
https://colab.research.google.com/drive/1CFVkdD2yIA6nh_1edNCV0nsvfVToVqbe#scrollTo=lsnDwUtMF_n1&line=9&uniqifier=1

Matriz de Confusão/ Distribuição



CURVA ROC



SENDO UM PROBLEMA DE CLASSIFICAÇÃO MULTICLASSE,

A CURVA ROC PRECISA SER PLOTADA INDIVIDUALMENTE PARA CADA CLASSE

Rates:

- True Positive Rate(TPR): True Positive/positive
- False Positive Rate(FPR): False Positive /Negative
- False Negative Rate(FNR): False Negative/Positive
- True Negative Rate(TNR): True Negative/Negative

K-MEANS

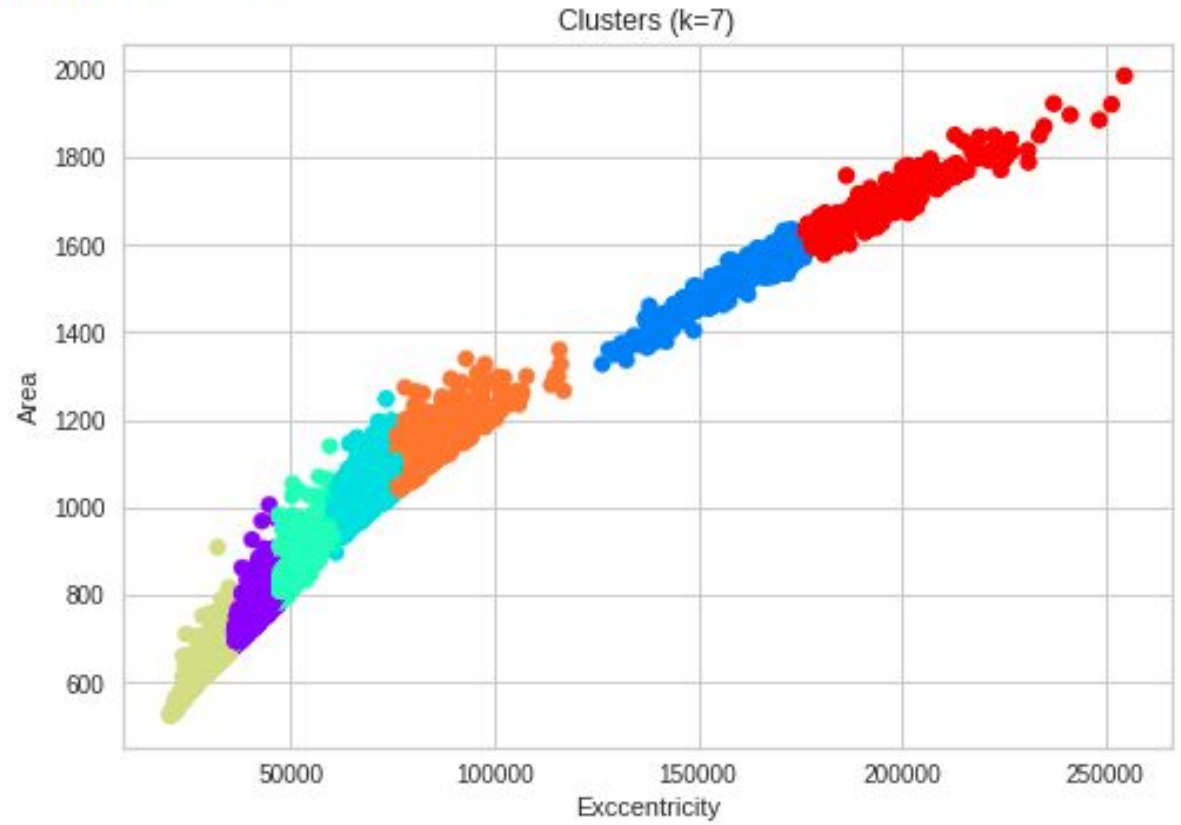
```
[19] kmeans7 = KMeans(n_clusters=7)
     y_kmeans7 = kmeans7.fit_predict(x)

     kmeans7.cluster_centers_
```

```
array([[4.15570952e+04, 7.61283100e+02, 2.79874388e+02, 1.90443634e+02,
        1.48216596e+00, 7.11944507e-01, 4.20311027e+04, 2.29867264e+02,
        7.54582116e-01, 9.88761020e-01, 9.02347101e-01, 8.26315711e-01,
        6.74175728e-03, 1.96447684e-03, 6.86268051e-01, 9.96459105e-01],
       [1.58100865e+05, 1.51061199e+03, 5.60876066e+02, 3.61467658e+02,
        1.55381552e+00, 7.60222957e-01, 1.60104604e+05, 4.48313187e+02,
        7.74875450e-01, 9.87531935e-01, 8.70088000e-01, 8.00794597e-01,
        3.55500985e-03, 9.06147431e-04, 6.42120054e-01, 9.92639035e-01],
       [6.87912919e+04, 1.02778059e+03, 3.86775741e+02, 2.28648939e+02,
        1.70374056e+00, 7.99149376e-01, 6.98985037e+04, 2.95803281e+02,
        7.47615068e-01, 9.84181433e-01, 8.19131931e-01, 7.67204793e-01,
        5.63706586e-03, 1.21176116e-03, 5.90613367e-01, 9.92712898e-01],
       [5.30644814e+04, 8.97870134e+02, 3.50255008e+02, 1.95049036e+02,
        1.81399124e+00, 8.17325544e-01, 5.38060571e+04, 2.59755781e+02,
        7.25142220e-01, 9.86305895e-01, 8.29396384e-01, 7.46709024e-01,
        6.60813491e-03, 1.28792897e-03, 5.61094190e-01, 9.93589683e-01],
       [3.11087556e+04, 6.53246146e+02, 2.38915361e+02, 1.66112622e+02,
        1.44696415e+00, 7.07957435e-01, 3.14769148e+04, 1.98674684e+02,
        7.56656441e-01, 9.88234586e-01, 9.13756823e-01, 8.33169509e-01,
        7.74352460e-03, 2.30787745e-03, 6.95961660e-01, 9.97080409e-01],
       [8.37442648e+04, 1.12862881e+03, 4.24804140e+02, 2.53110534e+02,
        1.68377023e+00, 7.98498739e-01, 8.50762987e+04, 3.26303794e+02,
        7.55360680e-01, 9.84354483e-01, 8.26630305e-01, 7.69610221e-01,
        5.08600340e-03, 1.10538992e-03, 5.93441555e-01, 9.92025950e-01],
       [1.95500465e+05, 1.69323707e+03, 6.39643548e+02, 3.92683714e+02,
```

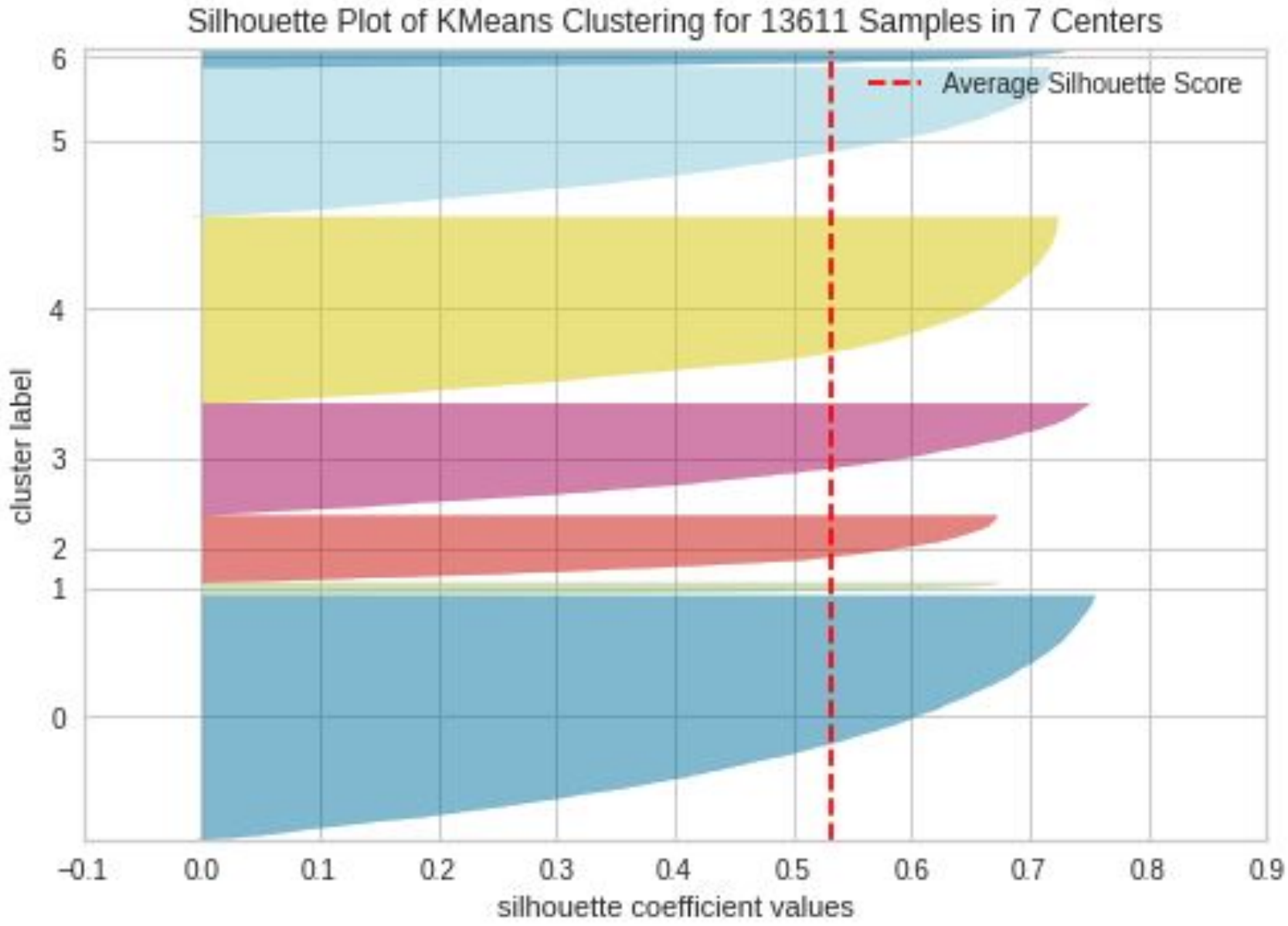
K-MEANS

```
[21] plt.scatter(x[:, 0], x[:, 1], c=y_kmeans7, cmap='rainbow')
     plt.title('Clusters (k=7)')
     plt.xlabel('Exccentricity')
     plt.ylabel('Area')
```

```
Text(0, 0.5, 'Area')
```



Silhouette K-MEANS



REFERÊNCIAS

Distributed Learning on Image Classification of Beans in TensorFlow. Disponível em: <https://towardsdatascience.com/distributed-learning-on-image-classification-of-beans-in-tensorflow-5a85e6c3eb71> Acesso: 24/03/2022.

Download do conjunto de dados do Dry Bean. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Dry+Bean+Dataset> Acesso em: 24/03/2022.

KOKLU, Murat; OZKAN, Ilker Ali. Multiclass classification of dry beans using computer vision and machine learning techniques. Computers and Electronics in Agriculture, v. 174, p. 105507, 2020.

LONG, Yunfei et al. Bean split ratio for dry bean canning quality and variety analysis. In: arxiv.org. Disponível em: < https://arxiv.org/pdf/1905.00336.pdf > Acesso em: 24/03/2022.

M. M. Hasan, M. U. Islam and M. J. Sadeq, "A Deep Neural Network for Multi-class Dry Beans Classification," 2021 24th International Conference on Computer and Information Technology (ICCIT), 2021, pp. 1-5, doi: 10.1109/ICCIT54785.2021.9689905.

SLOWIŃSKI, Grzegorz .Dry Beans Classification Using Machine Learning .In:CEUR Workshop Proceedings. Disponível em: < http://ceur-ws.org/Vol-2951/paper3.pdf > Acesso em: 24/03/2022.