USP - EACH - Reconhecimento de Padrões - Março de 2020 Tratamento do Arquivo Total SPHARM

- Ajuste do identificador da linha (paciente) - Primeira Coluna
- Ajuste na quantidade de colunas - 714 colunas
- Inclusão da Coluna Target (Variável Dependente) - Última Coluna 0 - Sem Anomalias - até a linha 101 1 - Hipertróficos - da 102 até a linha 284 2 - Dilatados - a partir da linha 285 inclusive

In [52]:

```
arqE = 'TotalSPHARM.txt'
arqS = 'TotalSPHARM_tratado.txt'
lant = ''
lmax = 0

with open(arqS, 'w') as saida:
    with open(arqE) as entrada:
        for l in entrada:
            linha = l.rstrip()

            #Despreza a primeira linha
            if linha[0:4] == ' id0':
                continue

            # Junta o Id com a linha posterior, arrumando a numeração do id
            if linha[0:3] == ' id':
                if (len(linha) == 4):
                    lant = linha[1:3] + '00' + linha[3:4]
                elif (len(linha) == 5):
                    lant = linha[1:3] + '0' + linha[3:5]
                elif (len(linha) == 6):
                    lant = linha[1:3] + linha[3:6]
                continue

            linha = lant + ',' + linha

            # Ajustar o tamanho das linhas com virgulas nos espaços necessários
            if (linha.count(',')) > lmax:
                lmax = linha.count(',')
                print('maximo de virgulas = ', lmax)

            difv = 714 - linha.count(',')

            for x in range(difv):
                linha = linha + ','

            # Inclui a classe na última coluna da linha e mais o comando de quabra de l
inha
            # 0 - Sem Anomalias - até a linha 101
            # 1 - Hipertróficos - da 102 até a linha 284
            # 2 - Dilatados - a partir da linha 285 inclusive
            if (lant < 'id102'):
                linha = linha[0:(len(linha) - 1)] + ',0\n'
            elif (lant < 'id285'):
                linha = linha[0:(len(linha) - 1)] + ',1\n'
            else:
                linha = linha[0:(len(linha) - 1)] + ',2\n'

#             if (lant == 'id192') or (lant > 'id287'): #in ['id192', 'id288']:
#                 print(linha.count(','), lant, linha)


            saida.write(linha)
```

```
maximo de virgulas =  458
maximo de virgulas =  464
maximo de virgulas =  528
maximo de virgulas =  612
maximo de virgulas =  714
```

In [53]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
df = pd.read_csv('TotalSPHARM_tratado.txt', header=None)
df.head()
```

Out[53]:

	0	1	2	3	4	5	
0	id001	-7881.480247	-5759.969698	-24465.608592	-15275.106756	-22974.378070	-22
1	id002	-567.772697	-33.292309	-465.179132	-525.981010	-469.546900	-
2	id003	-135372.767326	-115124.114646	-772665.053883	-292331.423079	-58059.255010	-118
3	id004	-582.939571	-366.425893	-281.022452	-437.739821	-206.814933	-
4	id005	-913.082501	-334.221895	-449.102108	-113.637478	-50.065343	-

5 rows × 715 columns

In [77]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
da = pd.read_csv('Arquivo.csv')
print(da.shape)
da.head()
```

```
(1263, 3)
```

Out[77]:

	Idade	Gênero	Nome_Arquivo
0	29.0	M	P1;
1	31.0	M	P2;
2	27.0	M	P3;
3	52.0	M	P4;
4	56.0	M	P5;

In [78]:

```
# Exclusão dos registros a partir da linha 400
dad = da.iloc[0:400,:]
print(dad.shape)
dad.head()
```

```
(400, 3)
```

Out[78]:

	Idade	Gênero	Nome_Arquivo
0	29.0	M	P1;
1	31.0	M	P2;
2	27.0	M	P3;
3	52.0	M	P4;
4	56.0	M	P5;

In [79]:

```
# Tirando o P do ID e transformando em numérico
for l in range(len(dad)):
    pal = dad.iloc[l, 2]
    dad.iloc[l, 2] = pal[1:-1]

dad.head()
```

Out[79]:

	Idade	Gênero	Nome_Arquivo
0	29.0	M	1
1	31.0	M	2
2	27.0	M	3
3	52.0	M	4
4	56.0	M	5

In [80]:

```
#Transformando a coluna de ID em numérico
dad['Nome_Arquivo'].astype(int)
dad.head()
```

Out[80]:

	Idade	Gênero	Nome_Arquivo
0	29.0	M	1
1	31.0	M	2
2	27.0	M	3
3	52.0	M	4
4	56.0	M	5

In [62]:

```
print(type(dad['Idade']))
```

```
<class 'pandas.core.series.Series'>
```

In [81]:

```
# Vamos transformar o sexo de M/F para 0/1
sexo = pd.get_dummies(dad['Gênero'])

# Reconstruindo o DataFrame
dadx = pd.DataFrame()
dadx[0] = dad['Nome_Arquivo']
dadx[1] = dad['Idade']
dadx[2] = sexo['F']
dadx[3] = sexo['M']

dadx.head()
```

Out[81]:

	0	1	2	3
0	1	29.0	0	1
1	2	31.0	0	1
2	3	27.0	0	1
3	4	52.0	0	1
4	5	56.0	0	1

In [82]:

```
# Ajuste do código da linha de Pxxx para idxxx, com exclusão da virgula
for l in range(len(dadx)):
    id = dadx.iloc[l, 0]

    if (len(id) == 1):
        id = 'id00' + id
    elif (len(id) == 2):
        id = 'id0' + id
    else:
        id = 'id' + id

    dadx.iloc[l, 0] = id

dadx.head()
```

Out[82]:

	0	1	2	3
0	id001	29.0	0	1
1	id002	31.0	0	1
2	id003	27.0	0	1
3	id004	52.0	0	1
4	id005	56.0	0	1

In [103]:

```
# Realiza o cruzamento dos dois datasets pelas colunas de identificação e inclui genero
e idade
dg = df.copy()
dg[715] = 0.00
dg[716] = 0.00
df[717] = 0

for l in range(len(df)):
    if (df.iloc[l, 0] == dadx.iloc[l, 0]):
        dg.iloc[l, 0] = dadx.iloc[l, 0]      #Id do Arquivo
        dg.iloc[l, 1] = dadx.iloc[l, 1]      #Idade
        dg.iloc[l, 2] = dadx.iloc[l, 2]      #Sexo - M
        dg.iloc[l, 3] = dadx.iloc[l, 3]      #Sexo - F
        for c in range(1, 715):
            #print(c)
            dg.iloc[l, c+3] = df.iloc[l, c]
    else:
        print("Deu pau", df.iloc[l, 0], dadx.iloc[l, 0])
        break

print(dg.shape)
dg.head()
```

```
(400, 718)
```

Out[103]:

	0	1	2	3	4	5	6	7	
0	id001	29.0	0.0	1.0	-7881.480247	-5759.969698	-24465.608592	-15275.106756	-229
1	id002	31.0	0.0	1.0	-567.772697	-33.292309	-465.179132	-525.981010	-4
2	id003	27.0	0.0	1.0	-135372.767326	-115124.114646	-772665.053883	-292331.423079	-580
3	id004	52.0	0.0	1.0	-582.939571	-366.425893	-281.022452	-437.739821	-2
4	id005	56.0	0.0	1.0	-913.082501	-334.221895	-449.102108	-113.637478	-

5 rows × 718 columns

In [105]:

```
#Gravando o arquivo de Saída
dg.to_csv (r'Total_SPHARM_20200326.csv', index = False, header = False)
```

In [106]:

```
#Verificando se a gravação foi ok
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
dh = pd.read_csv('Total_SPHARM_20200326.csv', header=None)
dh
```

Out[106]:

	0	1	2	3	4	5	6	7	
0	id001	29.0	0.0	1.0	-7.881480e+03	-5.759970e+03	-24465.608592	-1.527511e+04	-2.29
1	id002	31.0	0.0	1.0	-5.677727e+02	-3.329231e+01	-465.179132	-5.259810e+02	-4.69
2	id003	27.0	0.0	1.0	-1.353728e+05	-1.151241e+05	-772665.053883	-2.923314e+05	-5.80
3	id004	52.0	0.0	1.0	-5.829396e+02	-3.664259e+02	-281.022452	-4.377398e+02	-2.06
4	id005	56.0	0.0	1.0	-9.130825e+02	-3.342219e+02	-449.102108	-1.136375e+02	-5.00
5	id006	35.0	1.0	0.0	-1.366224e+02	-2.199550e+02	-58.230194	-4.328410e+01	-5.36
6	id007	57.0	0.0	1.0	-3.297554e+03	-1.648603e+03	-1826.976716	-1.772262e+03	-3.54
7	id008	38.0	0.0	1.0	-2.442526e+02	-7.486417e+02	-835.703744	-9.245954e+01	-1.05
8	id009	31.0	0.0	1.0	-9.637751e+03	-8.027820e+03	-5467.031364	-1.027926e+04	-5.24
9	id010	52.0	0.0	1.0	-2.487505e+03	-5.893294e+03	-1921.054444	-6.216152e+03	-4.69
10	id011	35.0	0.0	1.0	-4.768730e+02	-1.362988e+03	-1457.880666	-1.111917e+03	-1.83
11	id012	34.0	0.0	1.0	-2.605087e+04	-1.473219e+00	-39260.420820	-1.246688e+05	-7.41
12	id013	31.0	0.0	1.0	-8.408447e+02	-1.007348e+03	-696.233886	-4.659840e+02	-7.41
13	id014	35.0	0.0	1.0	-5.830271e+03	-4.275993e+03	-4695.216087	-2.029802e+03	-3.51
14	id015	67.0	0.0	1.0	-7.373532e+02	-8.577445e+01	-299.292120	-2.558461e+02	-4.90
15	id016	37.0	0.0	1.0	-1.939732e+05	-2.555080e+05	-637349.620003	-6.172288e+05	-6.13
16	id017	24.0	0.0	1.0	-1.234546e+04	-2.466000e+04	-34583.600042	-1.758072e+04	-4.75
17	id018	40.0	1.0	0.0	-1.975407e+03	-3.928906e+03	-519.970806	-1.428224e+03	-2.06
18	id019	59.0	1.0	0.0	-2.965385e+01	-1.005491e+03	-427.524169	-1.541901e+03	-3.37
19	id020	45.0	0.0	1.0	-6.985784e+03	-1.575974e+03	-6226.239818	-3.621642e+03	-2.05
20	id021	23.0	0.0	1.0	-3.165760e+03	-6.637580e+03	-1141.221562	-1.984825e+04	-1.02
21	id022	25.0	0.0	1.0	-1.843351e+04	-6.983452e+03	-12971.478639	-3.375355e+03	-1.15
22	id023	42.0	0.0	1.0	-4.888414e+02	-1.059763e+03	-644.330143	-3.450683e+02	-1.29
23	id024	23.0	0.0	1.0	-1.037149e+02	-8.570231e+01	-38.865049	-1.297267e+02	-1.81
24	id025	35.0	0.0	1.0	-1.471685e+02	-1.210296e+02	-359.786824	-4.481863e+02	-1.13
25	id026	39.0	0.0	1.0	-4.356068e+01	-3.991769e+02	-148.657240	-1.414063e+02	-3.69
26	id027	28.0	0.0	1.0	-1.621457e+05	-2.282778e+05	-37281.031339	-2.472966e+05	-2.29
27	id028	26.0	0.0	1.0	-7.647399e+03	-1.107888e+01	-2739.908928	-2.833898e+03	-7.74
28	id029	31.0	0.0	1.0	-5.576614e+04	-2.898171e+04	-21714.739023	-1.246992e+05	-5.83
29	id030	34.0	1.0	0.0	-7.036986e+03	-5.495268e+03	-6708.475802	-7.235843e+03	-9.03
...	...	...	...	...	...	...	...	...	
370	id371	64.0	0.0	1.0	-2.123185e+02	-2.003174e+02	-1164.322970	-5.212812e+02	-1.25
371	id372	51.0	0.0	1.0	-1.092637e+06	-1.029095e+06	-496288.691490	-1.031039e+06	-1.20
372	id373	59.0	0.0	1.0	-1.595783e+03	-3.129432e+03	-5718.329830	-4.201645e+03	-7.93
373	id374	55.0	0.0	1.0	-8.412583e+02	-1.723350e+03	-2085.425893	-8.785994e+02	-4.01
374	id375	59.0	0.0	1.0	-5.746846e+02	-1.369321e+03	-104.635421	-2.053959e+03	-1.34
375	id376	39.0	0.0	1.0	-2.893650e+03	-2.158697e+03	-267.445740	-1.953194e+03	-8.03

	0	1	2	3	4	5	6	7	
376	id377	63.0	0.0	1.0	-3.036861e+02	-5.751973e+02	-1161.662218	-9.622598e+02	-1.1
377	id378	54.0	0.0	1.0	-1.540265e+04	-2.169498e+04	-27419.023456	-1.040324e+05	-6.1
378	id379	70.0	1.0	0.0	-1.112629e+04	-1.583282e+04	-15633.258388	-2.926192e+03	-5.0
379	id380	34.0	0.0	1.0	-5.453420e+03	-4.028552e+03	-2555.298376	-1.557641e+03	-6.9
380	id381	54.0	0.0	1.0	-4.458359e+04	-1.173561e+04	-18321.934737	-2.013349e+04	-5.3
381	id382	75.0	0.0	1.0	-7.631249e+02	-4.829211e+02	-464.524393	-7.439314e+01	-2.1
382	id383	44.0	0.0	1.0	-6.033824e+03	-4.249743e+03	-5989.952192	-1.482744e+03	-3.8
383	id384	65.0	1.0	0.0	-1.110345e+03	-1.446968e+02	-905.564362	-1.302779e+03	-7.4
384	id385	42.0	0.0	1.0	-1.963345e+02	-2.764987e+03	-964.060887	-3.276351e+02	-1.9
385	id386	72.0	0.0	1.0	-7.789314e+04	-3.691938e+05	-319525.553558	-1.374414e+05	-4.7
386	id387	50.0	1.0	0.0	-9.773002e+03	-1.825279e+04	-13273.094795	-3.374229e+03	-7.7
387	id388	75.0	0.0	1.0	-9.773002e+03	-1.825279e+04	-13273.094795	-3.374229e+03	-7.7
388	id389	66.0	1.0	0.0	-3.610507e+02	-4.257124e+02	-265.402414	-7.200476e+02	-4.1
389	id390	54.0	0.0	1.0	-1.144933e+03	-4.090116e+02	-776.242840	-1.894642e+02	-2.7
390	id391	0.0	0.0	1.0	-3.112529e+03	-9.422838e+01	-2911.332390	-4.376021e+03	-2.2
391	id392	68.0	1.0	0.0	-7.386287e+00	-9.850329e+02	-362.244176	-9.246079e+02	-1.4
392	id393	36.0	1.0	0.0	-7.386287e+00	-9.850329e+02	-362.244176	-9.246079e+02	-1.4
393	id394	80.0	0.0	1.0	-8.431052e+02	-3.255629e+02	-420.200125	-1.874834e+02	-1.0
394	id395	70.0	0.0	1.0	-5.335541e+04	-2.603940e+04	-30552.558802	-2.401038e+04	-3.6
395	id396	31.0	1.0	0.0	-1.836571e+05	-6.745670e+05	-398795.513057	-4.435872e+05	-1.3
396	id397	70.0	0.0	1.0	-9.806476e+02	-1.872219e+03	-520.334038	-5.216247e+02	-1.3
397	id398	48.0	1.0	0.0	-8.778657e+03	-9.578976e+03	-3980.433446	-5.960246e+03	-8.8
398	id399	0.0	0.0	1.0	-1.271969e+02	-5.049371e+00	-124.475009	-1.609458e+02	-1.8
399	id400	46.0	1.0	0.0	-4.504015e+02	-1.227387e+02	-338.262264	-4.661443e+02	-3.3

400 rows × 718 columns

In [107]:

```
#Diagnóstico da Base
print('linhas = ', dg.shape[0], ' e quantidade de colunas ', dg.shape[1])
```

```
linhas =  400  e quantidade de colunas  718
```

In [ ]: