Mack_trab1

November 5, 2020

[1]:
```
import pandas as pd
import numpy as np
from sklearn.metrics import mean_squared_error
import tensorflow as tf
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
```

[2]:
```
df=pd.read_csv('cars-uci.csv',delimiter=';')
#eliminando linhas com missings
df=df.dropna()
```

[3]:
```
df.head()
```

[3]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	year	\
0	18.0	8	3070	130.0	3504	120	70	
1	15.0	8	3500	165.0	3693	115	70	
2	18.0	8	3180	150.0	3436	110	70	
3	16.0	8	3040	150.0	3433	120	70	
4	17.0	8	3020	140.0	3449	105	70	

	origin	name
0	1	chevrolet chevelle malibu
1	1	buick skylark 320
2	1	plymouth satellite
3	1	amc rebel sst
4	1	ford torino

[4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 392 entries, 0 to 405
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   mpg           392 non-null    float64
 1   cylinders     392 non-null    int64
 2   displacement  392 non-null    int64
 3   horsepower    392 non-null    float64
```

```
 4   weight        392 non-null    int64
 5   acceleration  392 non-null    int64
 6   year          392 non-null    int64
 7   origin        392 non-null    int64
 8   name          392 non-null    object
dtypes: float64(2), int64(6), object(1)
memory usage: 30.6+ KB
```

[5]:
```
#definindo as dimensões do para clustering
mpg=np.array(df['mpg'])
hp=np.array(df['horsepower'])
w=np.array(df['weight'])
```

[6]:
```
#simples verificação
hp[0]
```

[6]:
```
130.0
```

Standardization das features

[7]:
```
mpgm,mpgdp=mpg.mean(),mpg.std()
```

[8]:
```
hpm,hpdp=hp.mean(),hp.std()
```

[9]:
```
wm,wdp=df['weight'].mean(),df['weight'].std()
```

[10]:
```
XS=np.zeros((len(mpg),3)) # np.zeros vai criar uma array de 'len(mpg)' linhas␣
 ↪por
                          # colunas. A array é retornada para Xs
len(mpg), XS
```

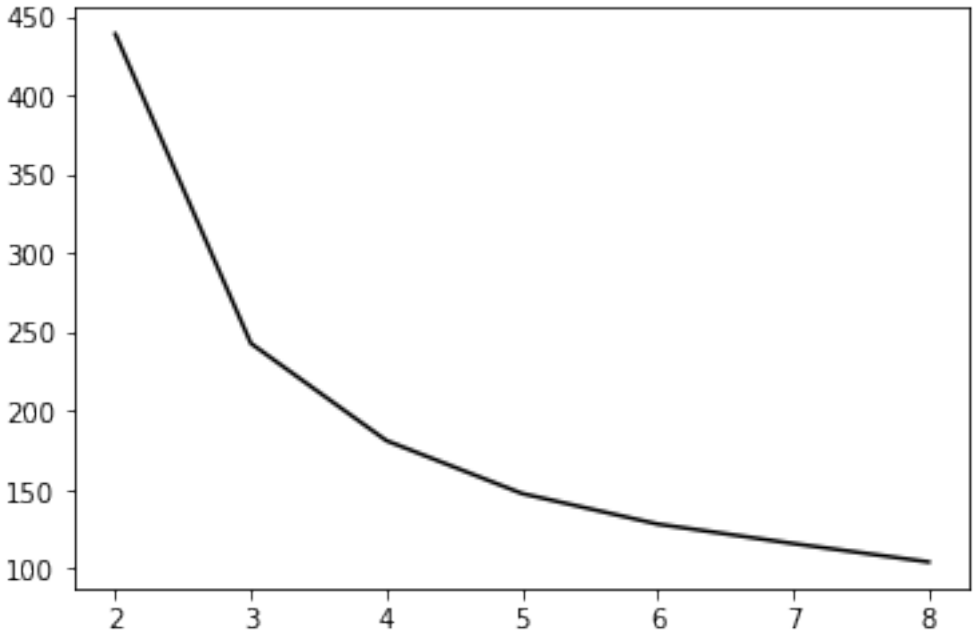
[10]:
```
(392,
 array([[0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.],
        …,
        [0., 0., 0.],
        [0., 0., 0.],
        [0., 0., 0.]]))
```

[11]:
```
# XS recebe os valores standardizados das features mp, hp e w, respectivamente
XS[:,0]=(mpg-mpgm)/mpgdp
XS[:,1]=(hp-hpm)/hpdp
XS[:,2]=(w-wm)/wdp
XS
```

[11]:
```
array([[-0.69863841,  0.66413273,  0.61974833],
       [-1.08349824,  1.57459447,  0.84225766],
       [-0.69863841,  1.18439658,  0.53969206],
       …,
       [ 1.09737414, -0.53247413, -0.80360505],
       [ 0.5842277 , -0.66254009, -0.41509668],
       [ 0.96908753, -0.58450051, -0.30325336]])
```

Fazer kmeans para k de 2 a 8 Traçar a curva do cotovelo Escolher um K Pegar 20 amostras de cada cluster e "explicá-las" (Storytelling)

[12]:
```
lk = [k for k in range(2,9)]
# for k in range(2,9):lk.append(k)
li = []
for i in range(2,9):
    km = KMeans(n_clusters=i)
    km.fit(XS)
    li.append(km.inertia_)
plt.plot(lk, li, color='black')
plt.show()
```



À partir da curva do cotovelo, entendo que o melhor K seja 3

[13]:
```
km = KMeans(n_clusters=3)
km.fit(XS)
```

[13]:
```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
       n_clusters=3, n_init=10, n_jobs=None, precompute_distances='auto',
       random_state=None, tol=0.0001, verbose=0)
```

[14]:
```
km.labels_
```

[14]:
```
array([1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 0, 0, 1, 1,
       1, 1, 1, 2, 2, 2, 2, 0, 0, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2,
       1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 1, 2, 2, 2, 2, 1, 1, 1, 0, 1, 0, 1, 0, 0, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 1, 1, 1, 1, 1, 0, 2, 2, 2, 2, 1, 1, 1, 1,
       1, 1, 1, 0, 2, 2, 0, 0, 1, 1, 2, 1, 1, 2, 1, 1, 1, 0, 0, 0, 1, 1,
       1, 1, 2, 2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1,
       2, 2, 2, 2, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1,
       1, 1, 1, 0, 0, 0, 1, 0, 0, 2, 2, 2, 2, 1, 1, 1, 1, 0, 0, 0, 0, 1,
       1, 1, 1, 0, 0, 0, 0, 1, 2, 1, 1, 2, 2, 2, 2, 2, 0, 0, 0, 1, 0, 2,
       2, 2, 2, 1, 1, 1, 1, 2, 2, 2, 2, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1,
       0, 0, 0, 0, 0, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2,
       0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 2, 0, 0, 1, 1, 1, 1, 1, 2, 2, 2,
       2, 2, 2, 1, 2, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0,
       0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
       1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0], dtype=int32)
```

[15]:
```
df['Cluster'] = km.labels_
```

[16]:
```
df.head(10)
```

[16]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	year	\
0	18.0	8	3070	130.0	3504	120	70	
1	15.0	8	3500	165.0	3693	115	70	
2	18.0	8	3180	150.0	3436	110	70	
3	16.0	8	3040	150.0	3433	120	70	
4	17.0	8	3020	140.0	3449	105	70	
5	15.0	8	4290	198.0	4341	100	70	
6	14.0	8	4540	220.0	4354	90	70	
7	14.0	8	4400	215.0	4312	85	70	
8	14.0	8	4550	225.0	4425	100	70	
9	15.0	8	3900	190.0	3850	85	70	

	origin	name	Cluster
0	1	chevrolet chevelle malibu	1
1	1	buick skylark 320	2
2	1	plymouth satellite	2
3	1	amc rebel sst	2
4	1	ford torino	2

```
5       1           ford galaxie 500        2
6       1           chevrolet impala        2
7       1          plymouth fury iii        2
8       1           pontiac catalina        2
9       1         amc ambassador dpl        2
```

[17]:
```
df[df['Cluster']==0].head(20)
```

[17]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	year	origin	name	Cluster
24	27.0	4	9700	88.0	2130	145	70	3	datsun pl510	0
25	26.0	4	9700	46.0	1835	205	70	2	volkswagen 1131 deluxe sedan	0
35	27.0	4	9700	88.0	2130	145	71	3	datsun pl510	0
36	28.0	4	1400	90.0	2264	155	71	1	chevrolet vega 2300	0
57	28.0	4	1160	90.0	2123	140	71	2	opel 1900	0
58	30.0	4	7900	70.0	2074	195	71	2	peugeot 304	0
59	30.0	4	8800	76.0	2065	145	71	2	fiat 124b	0
60	31.0	4	7100	65.0	1773	190	71	3	toyota corolla 1200	0
61	35.0	4	7200	69.0	1613	180	71	3	datsun 1200	0
62	27.0	4	9700	60.0	1834	190	71	2	volkswagen model 111	0
63	26.0	4	9100	70.0	1955	205	71	1	plymouth cricket	0
65	25.0	4	9750	80.0	2126	170	72	1	dodge colt hardtop	0
66	23.0	4	9700	54.0	2254	235	72	2	volkswagen type 3	0
86	26.0	4	9600	69.0	2189	180	72	2	renault 12 (sw)	0
88	28.0	4	9700	92.0	2288	170	72	3	datsun 510 (sw)	0
90	28.0	4	9800	80.0	2164	150	72	1	dodge colt (sw)	0
91	27.0	4	9700	88.0	2100	165	72			
109	26.0	4	9700	46.0	1950	210	73			
121	26.0	4	9800	90.0	2265	155	73			
124	29.0	4	6800	49.0	1867	195	73			

```
91        3      toyota corolla 1600 (sw)        0
109       2       volkswagen super beetle        0
121       2          fiat 124 sport coupe        0
124       2                       fiat 128        0
```

[18]:
```
df[df['Cluster']==1].head(20)
```

[18]:
```
      mpg  cylinders  displacement  horsepower  weight  acceleration  year  \
0    18.0          8          3070       130.0    3504           120    70
20   24.0          4          1130        95.0    2372           150    70
21   22.0          6          1980        95.0    2833           155    70
22   18.0          6          1990        97.0    2774           155    70
23   21.0          6          2000        85.0    2587           160    70
26   25.0          4          1100        87.0    2672           175    70
27   24.0          4          1070        90.0    2430           145    70
28   25.0          4          1040        95.0    2375           175    70
29   26.0          4          1210       113.0    2234           125    70
30   21.0          6          1990        90.0    2648           150    70
37   25.0          4          1130        95.0    2228           140    71
40   19.0          6          2320       100.0    2634           130    71
41   16.0          6          2250       105.0    3439           155    71
42   17.0          6          2500       100.0    3329           155    71
43   19.0          6          2500        88.0    3302           155    71
44   18.0          6          2320       100.0    3288           155    71
52   18.0          6          2580       110.0    2962           135    71
53   22.0          4          1400        72.0    2408           190    71
54   19.0          6          2500       100.0    3282           150    71
55   18.0          6          2500        88.0    3139           145    71

    origin                       name  Cluster
0        1  chevrolet chevelle malibu        1
20       3      toyota corona mark ii        1
21       1           plymouth duster        1
22       1                amc hornet        1
23       1             ford maverick        1
26       2               peugeot 504        1
27       2               audi 100 ls        1
28       2                  saab 99e        1
29       2                  bmw 2002        1
30       1              amc gremlin         1
37       3             toyota corona        1
40       1              amc gremlin         1
41       1  plymouth satellite custom        1
42       1  chevrolet chevelle malibu        1
43       1           ford torino 500        1
44       1              amc matador         1
52       1  amc hornet sportabout (sw)        1
```

```
53       1          chevrolet vega (sw)        1
54       1             pontiac firebird        1
55       1                 ford mustang        1
```

[19]:
```
df[df['Cluster']==2].head(20)
```

[19]:
```
     mpg  cylinders  displacement  horsepower  weight  acceleration  year  \
1   15.0          8          3500       165.0    3693           115    70
2   18.0          8          3180       150.0    3436           110    70
3   16.0          8          3040       150.0    3433           120    70
4   17.0          8          3020       140.0    3449           105    70
5   15.0          8          4290       198.0    4341           100    70
6   14.0          8          4540       220.0    4354            90    70
7   14.0          8          4400       215.0    4312            85    70
8   14.0          8          4550       225.0    4425           100    70
9   15.0          8          3900       190.0    3850            85    70
15  15.0          8          3830       170.0    3563           100    70
16  14.0          8          3400       160.0    3609            80    70
18  15.0          8          4000       150.0    3761            95    70
19  14.0          8          4550       225.0    3086           100    70
31  10.0          8          3600       215.0    4615           140    70
32  10.0          8          3070       200.0    4376           150    70
33  11.0          8          3180       210.0    4382           135    70
34   9.0          8          3040       193.0    4732           185    70
45  14.0          8          3500       165.0    4209           120    71
46  14.0          8          4000       175.0    4464           115    71
47  14.0          8          3510       153.0    4154           135    71

    origin                       name  Cluster
1        1          buick skylark 320        2
2        1         plymouth satellite        2
3        1              amc rebel sst        2
4        1                ford torino        2
5        1           ford galaxie 500        2
6        1           chevrolet impala        2
7        1          plymouth fury iii        2
8        1           pontiac catalina        2
9        1         amc ambassador dpl        2
15       1        dodge challenger se        2
16       1         plymouth 'cuda 340        2
18       1      chevrolet monte carlo        2
19       1    buick estate wagon (sw)        2
31       1                 ford f250        2
32       1                 chevy c20        2
33       1                dodge d200        2
34       1                  hi 1200d        2
45       1           chevrolet impala        2
```

```
46       1  pontiac catalina brougham        2
47       1           ford galaxie 500        2
```

Partindo da análise dos Clusters podemos observar que os carros ficam bem divididos de acordo com algumas características preponderantes. Para a análise aqui proposta, trago duas destas características:

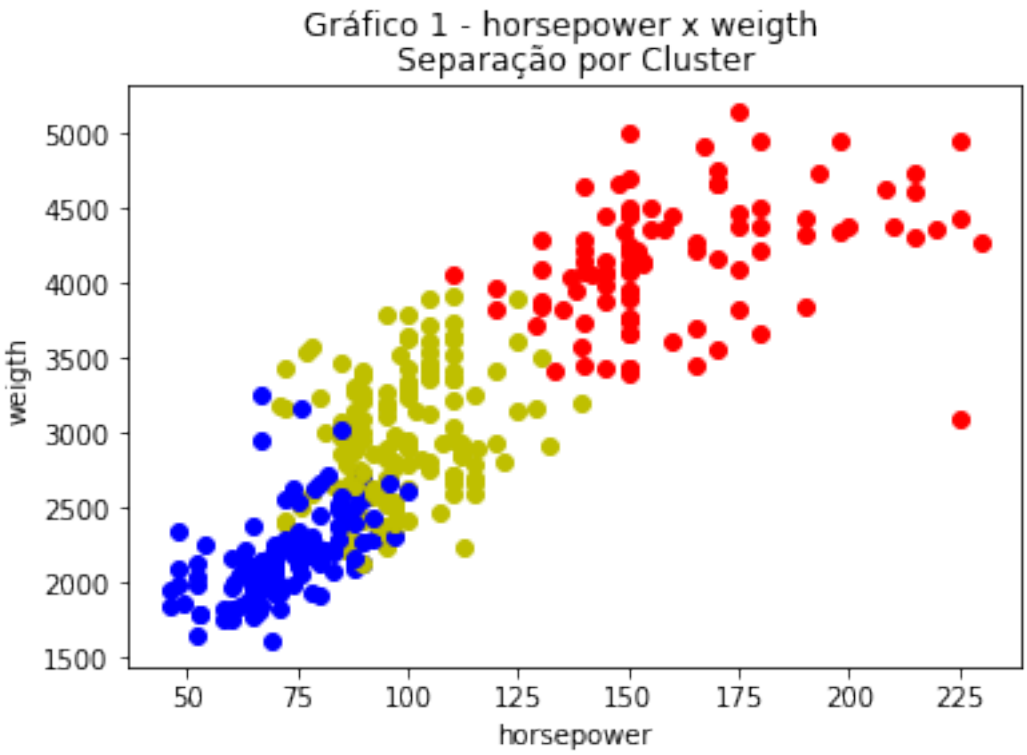
* Gráfico 1 (horsepower x weigth): No primeiro gráfico pode-se observar como estão bem definidos os clusters, onde quão maior a potência (horsepower) maior o peso do carro (weigth)
* Gráfico 2 (mpg x weigth): No gráfico 2 é demonstrado quanto o consumo (miles per galon) está diretamente ligado com o peso do carro. Mais uma vez, os cluster são bem definidos e mostram de maneira exato a relação peso x consumo.

As listas de clusters acima também demonstram tal separação por clusters e servem como referencial para os gráficos aqui apresentados.

[42]:
```
cor = ['bo', 'yo', 'ro']

for i in range(len(df)):
    plt.plot(df.iloc[i,3], df.iloc[i,4], cor[df.iloc[i,9]])

plt.title("Separação por Cluster")
plt.suptitle("Gráfico 1 - horsepower x weigth")
plt.xlabel("horsepower")
plt.ylabel("weigth")
plt.show()
```



[43]:
```
for i in range(len(df)):
    plt.plot(df.iloc[i,0], df.iloc[i,4], cor[df.iloc[i,9]])

plt.title("Separação por Cluster")
plt.suptitle("Gráfico 2 - mpg x weigth")
plt.xlabel("horsepower")
plt.ylabel("weigth")
plt.show()
```

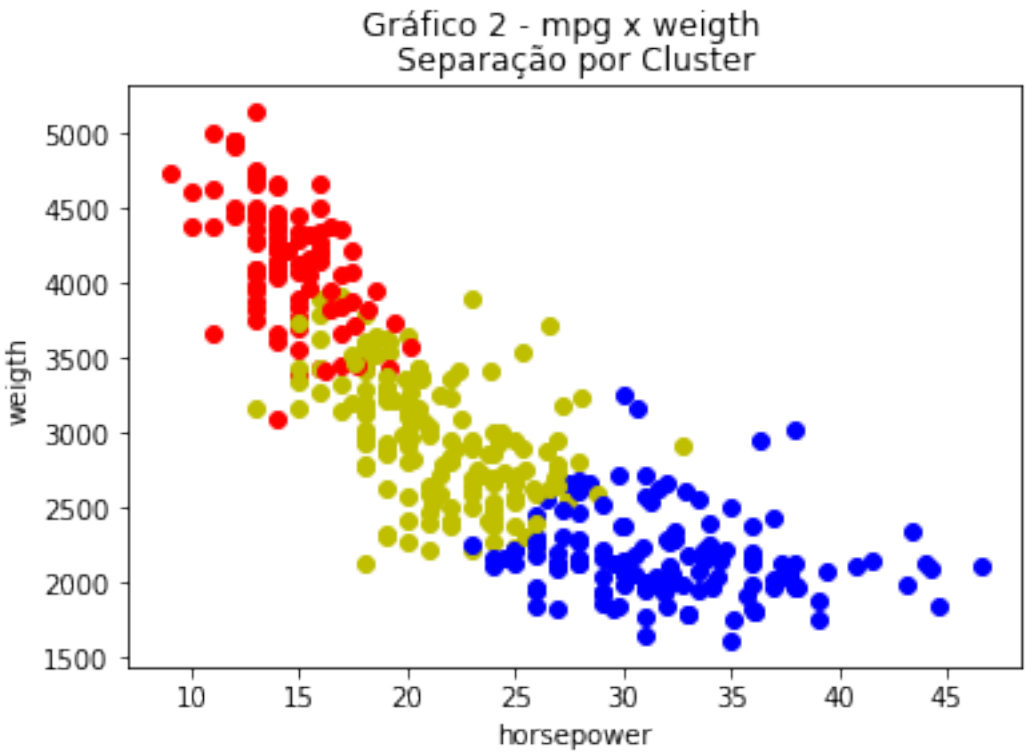


Gráfico 2 - mpg x weigth
Separação por Cluster
weigth
horsepower
5000
4500
4000
3500
3000
2500
2000
1500
10
15
20
25
30
35
40
45