Capstone _Week3

October 22, 2020

[6]:
```
import pandas as pd
import os
import numpy as np
from numpy import *
```

[7]:
```
os.listdir()
```

[7]:
```
['.ipynb_checkpoints',
 'ML0101EN-Reg-Simple-Linear-Regression-Co2-py-v1.ipynb',
 'FuelConsumption.csv',
 'ML0101EN-Reg-Mulitple-Linear-Regression-Co2-py-v1.ipynb',
 'ML0101EN-Reg-Polynomial-Regression-Co2-py-v1.ipynb',
 'ML0101EN-Reg-NoneLinearRegression-py-v1.ipynb',
 'china_gdp.csv',
 'ML0101EN-Clas-K-Nearest-neighbors-CustCat-py-v1.ipynb',
 'teleCust1000t.csv',
 'ML0101EN-Clas-Decision-Trees-drug-py-v1.ipynb',
 'drug200.csv',
 'drugtree.png',
 'ML0101EN-Clas-Logistic-Reg-churn-py-v1.ipynb',
 'ChurnData.csv',
 'ML0101EN-Clas-SVM-cancer-py-v1.ipynb',
 'cell_samples.csv',
 'ML0101EN-Clus-K-Means-Customer-Seg-py-v1.ipynb',
 'Cust_Segmentation.csv',
 'ML0101EN-Clus-Hierarchical-Cars-py-v1.ipynb',
 'cars_clus.csv',
 'ML0101EN-Clus-DBSCN-weather-py-v1.ipynb',
 'ML0101EN-RecSys-Content-Based-movies-py-v1.ipynb',
 'moviedataset.zip',
 'ML0101EN-RecSys-Collaborative-Filtering-movies-py-v1.ipynb',
 'links.csv',
 'movies.csv',
 'ratings.csv',
 'README.txt',
 'tags.csv',
 'Untitled Folder',
```

```
 'Crash_Analysis_System__CAS__Data.csv',
 'Capstone _Week3.ipynb',
 'Untitled.ipynb']
```

[8]:
```
os.getcwd()
```

[8]:
```
'/resources/labs/coursera/ML0101EN'
```

[9]:
```
df_test = pd.read_csv('Crash_Analysis_System__CAS__Data.csv')
print (df_test)
```

```
                X          Y  OBJECTID  advisorySpeed  areaUnitID  bicycle  \
0       1552332.0  5174780.0       219            NaN    587904.0        0
1       1249628.0  4839511.0       222            NaN    610074.0        0
2       1921849.0  5595608.0       267            NaN    545851.0        0
3       1737511.0  6000286.0       282            NaN    505010.0        0
4       1885602.0  5721570.0       318            NaN    541344.0        0
…             …          …         …              …           …        …
16333   1522183.0  5193492.0    730154            NaN    587100.0        0
16334   1828769.0  5536756.0    730195            NaN    560301.0        0
16335   1287341.0  5004235.0    730225            NaN    609031.0        0
16336   1351017.0  4897640.0    730238           75.0    607502.0        0
16337   1407137.0  4919068.0    730242            NaN    603601.0        0

       bridge  bus  carStationWagon  cliffBank  … train tree truck  \
0         0.0    0                1        0.0  …   0.0  0.0     0
1         NaN    0                2        NaN  …   NaN  NaN     0
2         0.0    0                1        0.0  …   0.0  1.0     1
3         0.0    0                0        1.0  …   0.0  0.0     0
4         NaN    0                2        NaN  …   NaN  NaN     0
…       …  …              …          …  …    …  …   …
16333     NaN    0                1        NaN  …   NaN  NaN     0
16334     NaN    0                1        NaN  …   NaN  NaN     0
16335     0.0    0                1        1.0  …   0.0  0.0     0
16336     0.0    0                0        0.0  …   0.0  0.0     0
16337     0.0    0                0        0.0  …   0.0  0.0     0

       unknownVehicleType  urban vanOrUtility  vehicle  waterRiver  \
0                       0   Open            0      0.0         0.0
1                       0   Open            0      NaN         NaN
2                       0   Open            0      0.0         0.0
3                       0   Open            0      0.0         0.0
4                       0   Open            0      NaN         NaN
…                   …    …          …       …         …
16333                   0   Open            1      NaN         NaN
16334                   0   Open            0      NaN         NaN
16335                   0   Open            0      0.0         0.0
```

```
16336                   0   Open            1      0.0         0.0
16337                   0  Urban            0      0.0         0.0

          weatherA     weatherB
0             Fine        Frost
1       Light rain  Strong wind
2             Fine  Strong wind
3             Fine  Strong wind
4             Null  Strong wind
…              …           …
16333         Fine  Strong wind
16334         Fine        Frost
16335         Fine        Frost
16336         Fine        Frost
16337         Snow  Strong wind

[16338 rows x 71 columns]
```

[10]:
```
df_test.shape
```

[10]:
```
(16338, 71)
```

[6]:
```
df_test.columns
```

[6]:
```
Index(['X', 'Y', 'OBJECTID', 'advisorySpeed', 'areaUnitID', 'bicycle',
       'bridge', 'bus', 'carStationWagon', 'cliffBank',
       'crashDirectionDescription', 'crashLocation1', 'crashLocation2',
       'crashRoadSideRoad', 'crashSeverity', 'crashSHDescription', 'crashYear',
       'debris', 'directionRoleDescription', 'ditch', 'fatalCount', 'fence',
       'flatHill', 'guardRail', 'holiday', 'houseOrBuilding', 'intersection',
       'kerb', 'light', 'meshblockId', 'minorInjuryCount', 'moped',
       'motorcycle', 'NumberOfLanes', 'objectThrownOrDropped', 'otherObject',
       'otherVehicleType', 'overBank', 'parkedVehicle', 'pedestrian',
       'phoneBoxEtc', 'postOrPole', 'region', 'roadCharacter', 'roadLane',
       'roadSurface', 'roadworks', 'schoolBus', 'seriousInjuryCount',
       'slipOrFlood', 'speedLimit', 'strayAnimal', 'streetLight', 'suv',
       'taxi', 'temporarySpeedLimit', 'tlaId', 'tlaName', 'trafficControl',
       'trafficIsland', 'trafficSign', 'train', 'tree', 'truck',
       'unknownVehicleType', 'urban', 'vanOrUtility', 'vehicle', 'waterRiver',
       'weatherA', 'weatherB'],
      dtype='object')
```

[13]:
```
indexNames = df_test[df_test['crashSeverity'] =='Non-Injury Crash'].index
df_test.drop(indexNames, inplace=True)
```

[16]:
```
sLenght = len(df_test['crashSeverity'])
```

```
df_test = df_test.assign(SeverityCode=pd.Series(np.random.randn(sLenght)).
↪values)
df_test.columns
```

[16]:
```
Index(['X', 'Y', 'OBJECTID', 'advisorySpeed', 'areaUnitID', 'bicycle',
       'bridge', 'bus', 'carStationWagon', 'cliffBank',
       'crashDirectionDescription', 'crashLocation1', 'crashLocation2',
       'crashRoadSideRoad', 'crashSeverity', 'crashSHDescription', 'crashYear',
       'debris', 'directionRoleDescription', 'ditch', 'fatalCount', 'fence',
       'flatHill', 'guardRail', 'holiday', 'houseOrBuilding', 'intersection',
       'kerb', 'light', 'meshblockId', 'minorInjuryCount', 'moped',
       'motorcycle', 'NumberOfLanes', 'objectThrownOrDropped', 'otherObject',
       'otherVehicleType', 'overBank', 'parkedVehicle', 'pedestrian',
       'phoneBoxEtc', 'postOrPole', 'region', 'roadCharacter', 'roadLane',
       'roadSurface', 'roadworks', 'schoolBus', 'seriousInjuryCount',
       'slipOrFlood', 'speedLimit', 'strayAnimal', 'streetLight', 'suv',
       'taxi', 'temporarySpeedLimit', 'tlaId', 'tlaName', 'trafficControl',
       'trafficIsland', 'trafficSign', 'train', 'tree', 'truck',
       'unknownVehicleType', 'urban', 'vanOrUtility', 'vehicle', 'waterRiver',
       'weatherA', 'weatherB', 'SeverityCode'],
      dtype='object')
```

[18]:
```
df_test.loc[df_test['SeverityCode']==1, ['crashSeverity']] = 'Minor Crash'
```

[19]:
```
df_test.loc[df_test['SeverityCode']==2, ['crashSeverity']] = 'Serious Crash'
```

[20]:
```
df_test.loc[df_test['SeverityCode']==3, ['crashSeverity']] = 'Fatal Crash'
```

[21]:
```
df = df_test[['SeverityCode','crashSeverity',␣
↪'fatalCount','minorInjuryCount','seriousInjuryCount','crashYear','region','speedLimit','adv
df_map = df_test[['crashSeverity','X','Y']]
```

[23]:
```
df.shape
```

[23]:
```
(5537, 11)
```

[24]:
```
df.info
```

[24]:
```
<bound method DataFrame.info of        SeverityCode  crashSeverity  fatalCount
minorInjuryCount  \
1         -0.361993  Serious Crash         0.0              1.0
3          0.584170  Serious Crash         0.0              0.0
5          0.424189    Minor Crash         0.0              1.0
7         -3.193104  Serious Crash         0.0              0.0
11        -0.249883    Minor Crash         0.0              1.0
…               …              …         …                …
16329     -0.688833    Minor Crash         0.0              1.0
```

```
16330     -1.196121    Minor Crash         0.0              7.0
16333      0.056705    Minor Crash         0.0              3.0
16334      1.696519    Minor Crash         0.0              1.0
16336     -0.069453  Serious Crash         0.0              0.0

       seriousInjuryCount  crashYear                  region  speedLimit  \
1                     1.0       2017          Southland Region       100.0
3                     1.0       2016          Northland Region       100.0
5                     0.0       2018        Wellington Region        50.0
7                     1.0       2016  Manawatu-Wanganui Region        70.0
11                    0.0       2006        Canterbury Region        50.0
…                     …         …                       …          …
16329                 0.0       2014          Auckland Region        50.0
16330                 0.0       2007             Otago Region       100.0
16333                 0.0       2018        Canterbury Region       100.0
16334                 0.0       2013  Manawatu-Wanganui Region       100.0
16336                 1.0       2016             Otago Region       100.0

       advisorySpeed    weatherA     weatherB
1                NaN  Light rain  Strong wind
3                NaN        Fine  Strong wind
5                NaN        Fine  Strong wind
7                NaN        Fine        Frost
11               NaN        Fine  Strong wind
…                …          …           …
16329            NaN  Heavy rain  Strong wind
16330            NaN        Fine  Strong wind
16333            NaN        Fine  Strong wind
16334            NaN        Fine        Frost
16336           75.0        Fine        Frost

[5537 rows x 11 columns]>
```

[25]:
```
from numpy import *
import warnings
warnings.filterwarnings('ignore')
df.replace("?", np.nan, inplace = True)
```

[26]:
```
missing_data = df.isnull()
missing_data.head(5)
```

[26]:
```
    SeverityCode  crashSeverity  fatalCount  minorInjuryCount  \
1          False          False       False             False
3          False          False       False             False
5          False          False       False             False
7          False          False       False             False
11         False          False       False             False
```

```
    seriousInjuryCount  crashYear  region  speedLimit  advisorySpeed  \
1                False      False   False       False           True
3                False      False   False       False           True
5                False      False   False       False           True
7                False      False   False       False           True
11               False      False   False       False           True

    weatherA  weatherB
1      False     False
3      False     False
5      False     False
7      False     False
11     False     False
```

[27]:
```
for column in missing_data.columns.values.tolist():
    print(column)
    print(missing_data[column].value_counts())
    print("--------------------")
```

```
SeverityCode
False    5537
Name: SeverityCode, dtype: int64
--------------------
crashSeverity
False    5537
Name: crashSeverity, dtype: int64
--------------------
fatalCount
False    5537
Name: fatalCount, dtype: int64
--------------------
minorInjuryCount
False    5537
Name: minorInjuryCount, dtype: int64
--------------------
seriousInjuryCount
False    5537
Name: seriousInjuryCount, dtype: int64
--------------------
crashYear
False    5537
Name: crashYear, dtype: int64
--------------------
region
False    5537
Name: region, dtype: int64
```

```
--------------------
speedLimit
False    5536
True        1
Name: speedLimit, dtype: int64
--------------------
advisorySpeed
True     5059
False     478
Name: advisorySpeed, dtype: int64
--------------------
weatherA
False    5537
Name: weatherA, dtype: int64
--------------------
weatherB
False    5537
Name: weatherB, dtype: int64
--------------------
```

[54]:
```
df.dtypes
```

[54]:
```
SeverityCode          float64
crashSeverity          object
fatalCount            float64
minorInjuryCount      float64
seriousInjuryCount    float64
crashYear               int64
region                 object
speedLimit            float64
advisorySpeed         float64
weatherA               object
weatherB               object
dtype: object
```

[79]:
```
df["fatalCount"] = df['fatalCount'].replace('', np.nan).ffill()
df["fatalCount"].describe()
```

[79]:
```
count    5537.000000
mean        0.032328
std         0.207001
min         0.000000
25%         0.000000
50%         0.000000
75%         0.000000
max         5.000000
Name: fatalCount, dtype: float64
```

[81]:
```
df["minorInjuryCount"] = df['minorInjuryCount'].replace('', np.nan).ffill()
df["minorInjuryCount"].describe()
```

[81]:
```
count    5537.000000
mean        1.067546
std         0.855611
min         0.000000
25%         1.000000
50%         1.000000
75%         1.000000
max        18.000000
Name: minorInjuryCount, dtype: float64
```

[82]:
```
df["seriousInjuryCount"] = df['seriousInjuryCount'].replace('', np.nan).ffill()
df["seriousInjuryCount"].describe()
```

[82]:
```
count    5537.000000
mean        0.254651
std         0.557228
min         0.000000
25%         0.000000
50%         0.000000
75%         0.000000
max         9.000000
Name: seriousInjuryCount, dtype: float64
```

[31]:
```
df['weatherA'].value_counts()
```

[31]:
```
Fine             3508
Light rain        785
Heavy rain        709
Mist or Fog       238
Null              164
Snow              129
Hail or Sleet       4
Name: weatherA, dtype: int64
```

[86]:
```
df['speedLimit'] = df['speedLimit'].replace('', np.nan).ffill()
df['speedLimit'].describe()
```

[86]:
```
count    5537.000000
mean       80.854253
std        23.338810
min        10.000000
25%        50.000000
50%       100.000000
75%       100.000000
```

```
max       100.000000
Name: speedLimit, dtype: float64
```

[34]:
```
df['weatherB'].value_counts()
```

[34]:
```
Strong wind    3317
Frost          2220
Name: weatherB, dtype: int64
```

[35]:
```
df['region'].replace(np.nan, df["region"].value_counts().idxmax(), inplace=True)
df['region'].value_counts()
```

[35]:
```
Otago Region                903
Auckland Region             863
Canterbury Region           663
Wellington Region           609
Waikato Region              540
Southland Region            474
Manawatu-Wanganui Region    425
Bay of Plenty Region        233
Northland Region            189
Hawke's Bay Region          161
Taranaki Region             150
West Coast Region            94
Marlborough Region           70
Tasman Region                65
Gisborne Region              59
Nelson Region                39
Name: region, dtype: int64
```

[75]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 5537 entries, 1 to 16336
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   SeverityCode        5537 non-null   float64
 1   crashSeverity       5537 non-null   object
 2   fatalCount          5537 non-null   float64
 3   minorInjuryCount    5537 non-null   float64
 4   seriousInjuryCount  5537 non-null   float64
 5   crashYear           5537 non-null   int64
 6   region              5537 non-null   object
 7   speedLimit          5536 non-null   float64
 8   advisorySpeed       5537 non-null   float64
 9   weatherA            5537 non-null   object
```

```
 10  weatherB            5537 non-null   object
dtypes: float64(6), int64(1), object(4)
memory usage: 519.1+ KB
```

[76]:
```
df_map.dropna(subset=["X"], axis=0, inplace=True)
df_map.shape
```

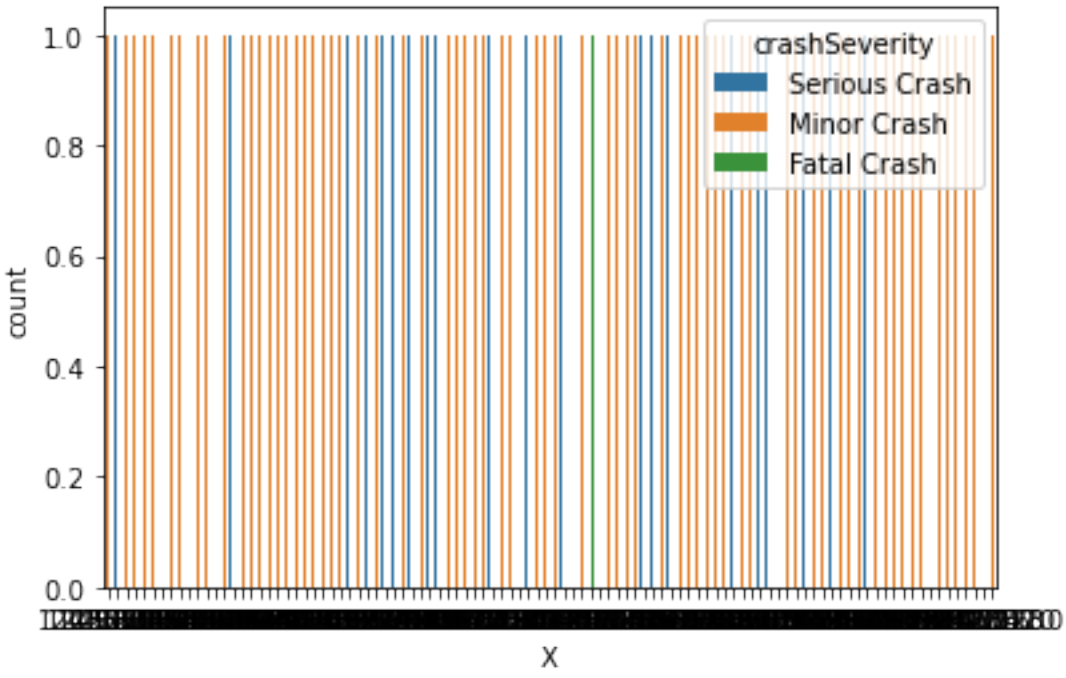
[76]:
```
(5537, 3)
```

[41]:
```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

[42]:
```
"""
seaborn.countplot(x-None,y=None, hue=None,
                  data=None, order=None, hue_order=None,
                  orient=None, color=None, palette=None,
                  saturation=0.75, dodge=True, ax=None,**kwargs)
"""

sns.countplot(x = "X", data= df_map.head(100), hue= "crashSeverity")
```

[42]:
```
<AxesSubplot:xlabel='X', ylabel='count'>
```



[43]:
```
import itertools
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import NullFormatter
import pandas as pd
import numpy as np
import matplotlib.ticker as ticker
from sklearn import preprocessing
%matplotlib inline
```

[61]:
```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import GridSearchCV
from sklearn import metrics
```

[44]:
```
df.columns
```

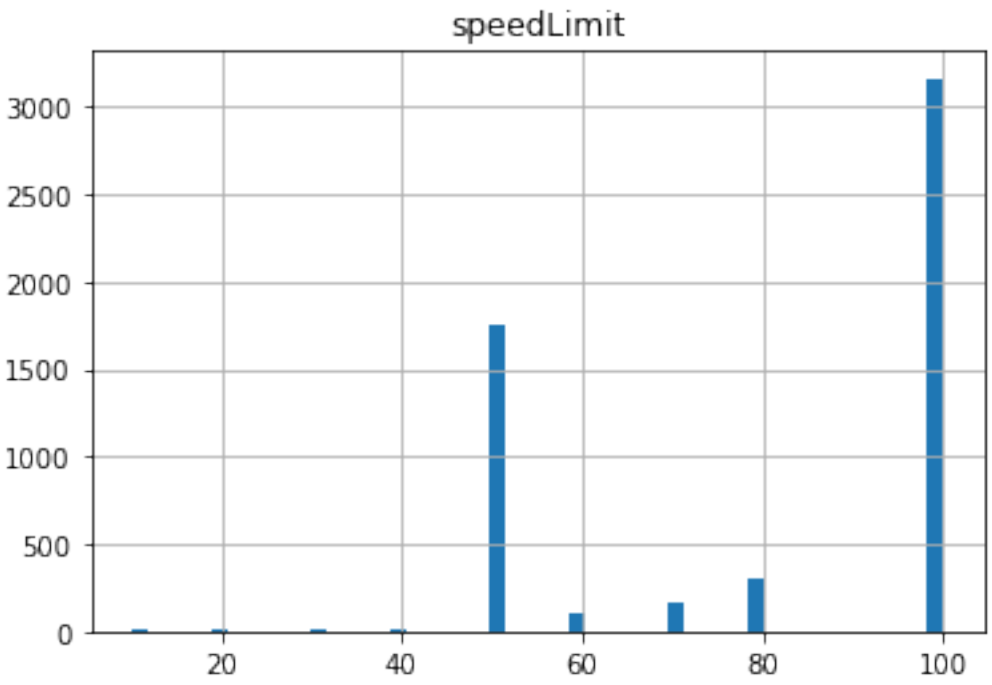
[44]:
```
Index(['SeverityCode', 'crashSeverity', 'fatalCount', 'minorInjuryCount',
       'seriousInjuryCount', 'crashYear', 'region', 'speedLimit',
       'advisorySpeed', 'weatherA', 'weatherB'],
      dtype='object')
```

[51]:
```
avg_speedLimit = df['speedLimit'].astype('float').mean(axis=0)
print("Average Speed LImit:", avg_speedLimit)
```

```
Average Speed LImit: 80.85079479768787
```

[52]:
```
df.hist(column='speedLimit', bins=50)
```

[52]:
```
array([[<AxesSubplot:title={'center':'speedLimit'}>]], dtype=object)
```



[53]:
```
df['advisorySpeed'].replace(np.nan, avg_speedLimit, inplace=True)
```

[88]:
```
X = df [['fatalCount',␣
 ↪'minorInjuryCount','seriousInjuryCount','crashYear','speedLimit']].values
X[0:5]
```

[88]:
```
array([[0.000e+00, 1.000e+00, 1.000e+00, 2.017e+03, 1.000e+02],
       [0.000e+00, 0.000e+00, 1.000e+00, 2.016e+03, 1.000e+02],
       [0.000e+00, 1.000e+00, 0.000e+00, 2.018e+03, 5.000e+01],
       [0.000e+00, 0.000e+00, 1.000e+00, 2.016e+03, 7.000e+01],
       [0.000e+00, 1.000e+00, 0.000e+00, 2.006e+03, 5.000e+01]])
```

[89]:
```
y = df['SeverityCode'].values
y[0:5]
```

[89]:
```
array([-0.36199316,  0.58416976,  0.42418926, -3.1931036 , -0.24988329])
```

[90]:
```
X = preprocessing.StandardScaler().fit(X).transform(X.astype(float))
X[0:5]
```

[90]:
```
array([[-0.15618749, -0.07895144,  1.33772395,  0.88590045,  0.82041356],
       [-0.15618749, -1.2478128 ,  1.33772395,  0.68412214,  0.82041356],
       [-0.15618749, -0.07895144, -0.45703677,  1.08767876, -1.32213426],
       [-0.15618749, -1.2478128 ,  1.33772395,  0.68412214, -0.46511513],
```

```
       [-0.15618749, -0.07895144, -0.45703677, -1.333661   , -1.32213426]])
```

[67]:
```
def readLines():
    def conv(s):
        try:
            s=float(s)
        except ValueError:
            pass
        return s

    with open ('testdata.csv','rU') as data :
        for cell in row:
            y=conv(cell)
```

[68]:
```
n1 = float("nan")
print(n1)
```

```
nan
```

[48]:
```
import math

n1 = math.nan
print(n1)
print(math.isnan(n1))
```

```
nan
True
```

[56]:
```
df['advisorySpeed'].replace(np.nan, avg_speedLimit, inplace=True)
```

[91]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=4)

print ('Train set:', X_train.shape, y_train.shape)
print ('Test set:', X_test.shape, y_test.shape)
```

```
Train set: (4429, 5) (4429,)
Test set: (1108, 5) (1108,)
```

[70]:
```
import scipy.sparse
scipy.sparse.issparse(scipy.sparse.bsr_matrix([[1,0],[0,1]]))
```

[70]:
```
True
```

[58]:
```
import math
```

```
x1 = float("nan")

print(f"It's pd.isna  : {pd.isna(x1)}")
print(f"It's np.isnan : {np.isnan(x1)}")
print(f"It's math.isnan :{math.isnan(x1)}")
```

```
It's pd.isna  : True
It's np.isnan : True
It's math.isnan :True
```

[59]:
```
math.isnan(avg_speedLimit)
```

[59]:
```
False
```

[ ]:
```

```