Capstonedatascience

September 13, 2020

[28]:
```
import pandas as pd
import pylab as pl
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import matplotlib as mpl
```

[3]:
```
#let's import the example dataset from IBM
main_df=pd.read_csv ('https://s3.us.cloud-object-storage.appdomain.cloud/
 ↪cf-courses-data/CognitiveClass/DP0701EN/version-2/Data-Collisions.csv')
main_df.head ()
```

```
/home/jupyterlab/conda/envs/python/lib/python3.6/site-
packages/IPython/core/interactiveshell.py:3072: DtypeWarning: Columns (33) have
mixed types.Specify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

[3]:
```
   SEVERITYCODE           X          Y  OBJECTID  INCKEY  COLDETKEY REPORTNO  \
0             2 -122.323148  47.703140         1    1307       1307  3502005
1             1 -122.347294  47.647172         2   52200      52200  2607959
2             1 -122.334540  47.607871         3   26700      26700  1482393
3             1 -122.334803  47.604803         4    1144       1144  3503937
4             2 -122.306426  47.545739         5   17700      17700  1807429

    STATUS      ADDRTYPE   INTKEY  … ROADCOND                LIGHTCOND  \
0  Matched  Intersection  37475.0  …      Wet                 Daylight
1  Matched         Block      NaN  …      Wet  Dark - Street Lights On
2  Matched         Block      NaN  …      Dry                 Daylight
3  Matched         Block      NaN  …      Dry                 Daylight
4  Matched  Intersection  34387.0  …      Wet                 Daylight

  PEDROWNOTGRNT  SDOTCOLNUM SPEEDING ST_COLCODE  \
0           NaN         NaN      NaN         10
1           NaN   6354039.0      NaN         11
2           NaN   4323031.0      NaN         32
3           NaN         NaN      NaN         23
4           NaN   4028032.0      NaN         10
```

```
                                          ST_COLDESC  SEGLANEKEY  \
0                                  Entering at angle           0
1  From same direction - both going straight - bo…          0
2                             One parked--one moving           0
3                   From same direction - all others           0
4                                  Entering at angle           0

   CROSSWALKKEY  HITPARKEDCAR
0             0             N
1             0             N
2             0             N
3             0             N
4             0             N

[5 rows x 38 columns]
```

[4]:
```
#Now we will have a quick overview of the dataset we are dealing with
main_df.shape
```

[4]:
```
(194673, 38)
```

[5]:
```
#Looking at the columns we can determine what we need and what can be dropped
main_df.columns
```

[5]:
```
Index(['SEVERITYCODE', 'X', 'Y', 'OBJECTID', 'INCKEY', 'COLDETKEY', 'REPORTNO',
       'STATUS', 'ADDRTYPE', 'INTKEY', 'LOCATION', 'EXCEPTRSNCODE',
       'EXCEPTRSNDESC', 'SEVERITYCODE.1', 'SEVERITYDESC', 'COLLISIONTYPE',
       'PERSONCOUNT', 'PEDCOUNT', 'PEDCYLCOUNT', 'VEHCOUNT', 'INCDATE',
       'INCDTTM', 'JUNCTIONTYPE', 'SDOT_COLCODE', 'SDOT_COLDESC',
       'INATTENTIONIND', 'UNDERINFL', 'WEATHER', 'ROADCOND', 'LIGHTCOND',
       'PEDROWNOTGRNT', 'SDOTCOLNUM', 'SPEEDING', 'ST_COLCODE', 'ST_COLDESC',
       'SEGLANEKEY', 'CROSSWALKKEY', 'HITPARKEDCAR'],
      dtype='object')
```

0.1 Data Cleaning

Now that we have imported the data and had a quick verview of it we can begin the process of cleaning it and preparing it for the project

[6]:
```
#Here we drop all columns we do not need
main_df.drop(main_df.columns.difference(['SEVERITYCODE','SEVERITYDESC',␣
 ↪'ADDRTYPE', 'JUNCTIONTYPE', 'SDOT_COLDESC', 'WEATHER', 'LIGHTCOND',␣
 ↪'ROADCOND'])\
, axis=1, inplace=True)
main_df.head()
```

```
[6]:    SEVERITYCODE      ADDRTYPE                    SEVERITYDESC  \
     0             2  Intersection                Injury Collision
     1             1         Block  Property Damage Only Collision
     2             1         Block  Property Damage Only Collision
     3             1         Block  Property Damage Only Collision
     4             2  Intersection                Injury Collision

                                   JUNCTIONTYPE  \
     0   At Intersection (intersection related)
     1  Mid-Block (not related to intersection)
     2  Mid-Block (not related to intersection)
     3  Mid-Block (not related to intersection)
     4   At Intersection (intersection related)

                                             SDOT_COLDESC   WEATHER ROADCOND  \
     0  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …  Overcast      Wet
     1  MOTOR VEHICLE STRUCK MOTOR VEHICLE, LEFT SIDE …   Raining      Wet
     2       MOTOR VEHICLE STRUCK MOTOR VEHICLE, REAR END  Overcast      Dry
     3  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …     Clear      Dry
     4  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …   Raining      Wet

                      LIGHTCOND
     0                 Daylight
     1  Dark - Street Lights On
     2                 Daylight
     3                 Daylight
     4                 Daylight
```

```
[7]: #This provides a cleaner view of the columns we are going to use
     main_df.head(0).transpose()
```

```
[7]: Empty DataFrame
     Columns: []
     Index: [SEVERITYCODE, ADDRTYPE, SEVERITYDESC, JUNCTIONTYPE, SDOT_COLDESC,
     WEATHER, ROADCOND, LIGHTCOND]
```

```
[8]: #Now let's check for null values using boolean results
     null_values=main_df.isnull()
     null_values
```

```
[8]:         SEVERITYCODE  ADDRTYPE  SEVERITYDESC  JUNCTIONTYPE  SDOT_COLDESC  \
     0              False     False         False         False         False
     1              False     False         False         False         False
     2              False     False         False         False         False
     3              False     False         False         False         False
     4              False     False         False         False         False
     …                …       …           …           …           …
```

```
194668         False     False         False         False         False
194669         False     False         False         False         False
194670         False     False         False         False         False
194671         False     False         False         False         False
194672         False     False         False         False         False

        WEATHER  ROADCOND  LIGHTCOND
0         False     False      False
1         False     False      False
2         False     False      False
3         False     False      False
4         False     False      False
…         …       …        …
194668    False     False      False
194669    False     False      False
194670    False     False      False
194671    False     False      False
194672    False     False      False

[194673 rows x 8 columns]
```

[9]:

```python
# We will check for null elements
for column in null_values.columns.values.tolist():
    print(column)
    print(null_values[column].value_counts().sort_values(ascending=True))
    print("")
```

```
SEVERITYCODE
False    194673
Name: SEVERITYCODE, dtype: int64

ADDRTYPE
True       1926
False    192747
Name: ADDRTYPE, dtype: int64

SEVERITYDESC
False    194673
Name: SEVERITYDESC, dtype: int64

JUNCTIONTYPE
True       6329
False    188344
Name: JUNCTIONTYPE, dtype: int64

SDOT_COLDESC
False    194673
```

```
Name: SDOT_COLDESC, dtype: int64

WEATHER
True       5081
False    189592
Name: WEATHER, dtype: int64

ROADCOND
True       5012
False    189661
Name: ROADCOND, dtype: int64

LIGHTCOND
True       5170
False    189503
Name: LIGHTCOND, dtype: int64
```

[10]:

```
#gives statistics for categorical variables
main_df.describe(include='O')
```

[10]:

```
       ADDRTYPE                    SEVERITYDESC  \
count    192747                          194673
unique        3                               2
top       Block  Property Damage Only Collision
freq     126926                          136485

                                   JUNCTIONTYPE  \
count                                    188344
unique                                        7
top     Mid-Block (not related to intersection)
freq                                      89800

                                             SDOT_COLDESC WEATHER ROADCOND  \
count                                              194673  189592   189661
unique                                                 39      11        9
top     MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …   Clear      Dry
freq                                                85209  111135   124510

       LIGHTCOND
count     189503
unique         9
top     Daylight
freq      116137
```

[11]:

```
main_df_with_nans=main_df.dropna()
main_df_with_nans.shape
```

[11]:
```
(182914, 8)
```

[12]:
```
a=(1-(182954/194673))*100
print("%.2f" % a,"%")
```

```
6.02 %
```

[13]:
```
# With 6.02% rows with nans, we will drop these rows
main_df=main_df.dropna()
main_df.shape
```

[13]:
```
(182914, 8)
```

[14]:
```
main_df.head()
```

[14]:
```
   SEVERITYCODE      ADDRTYPE                    SEVERITYDESC  \
0             2  Intersection                Injury Collision
1             1         Block  Property Damage Only Collision
2             1         Block  Property Damage Only Collision
3             1         Block  Property Damage Only Collision
4             2  Intersection                Injury Collision

                                JUNCTIONTYPE  \
0   At Intersection (intersection related)
1  Mid-Block (not related to intersection)
2  Mid-Block (not related to intersection)
3  Mid-Block (not related to intersection)
4   At Intersection (intersection related)

                                     SDOT_COLDESC   WEATHER ROADCOND  \
0  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …  Overcast      Wet
1  MOTOR VEHICLE STRUCK MOTOR VEHICLE, LEFT SIDE …   Raining      Wet
2       MOTOR VEHICLE STRUCK MOTOR VEHICLE, REAR END  Overcast      Dry
3  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …     Clear      Dry
4  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …   Raining      Wet

                  LIGHTCOND
0                  Daylight
1  Dark - Street Lights On
2                  Daylight
3                  Daylight
4                  Daylight
```

[15]:
```
# we will reset index to correct rows numbers
main_df=main_df.reset_index(drop=True)
main_df.head()
```

[15]:

```
   SEVERITYCODE      ADDRTYPE                    SEVERITYDESC  \
0             2  Intersection                Injury Collision
1             1         Block  Property Damage Only Collision
2             1         Block  Property Damage Only Collision
3             1         Block  Property Damage Only Collision
4             2  Intersection                Injury Collision

                              JUNCTIONTYPE  \
0   At Intersection (intersection related)
1  Mid-Block (not related to intersection)
2  Mid-Block (not related to intersection)
3  Mid-Block (not related to intersection)
4   At Intersection (intersection related)

                                       SDOT_COLDESC   WEATHER ROADCOND  \
0  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …  Overcast      Wet
1  MOTOR VEHICLE STRUCK MOTOR VEHICLE, LEFT SIDE …   Raining      Wet
2       MOTOR VEHICLE STRUCK MOTOR VEHICLE, REAR END  Overcast      Dry
3  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …     Clear      Dry
4  MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END …   Raining      Wet

                 LIGHTCOND
0                 Daylight
1  Dark - Street Lights On
2                 Daylight
3                 Daylight
4                 Daylight
```

[27]:

```python
#Check numerical values of data
print ("SEVERITYCODE: \n", main_df ['SEVERITYCODE'].value_counts())
print("ADDRTYPE: \n", main_df['ADDRTYPE'].value_counts() )
print("LIGHTCOND: \n",main_df['LIGHTCOND'].value_counts())

print("\n WEATHER: \n",main_df['WEATHER'].value_counts())
print("\n JUNCTIONTYPE: \n",main_df['JUNCTIONTYPE'].value_counts())
print("\n SDOT_COLDESC: \n",main_df['SDOT_COLDESC'].value_counts())
print("\n ROADCOND: \n",main_df['ROADCOND'].value_counts())
```

```
SEVERITYCODE:
 1    126276
2     56638
Name: SEVERITYCODE, dtype: int64
ADDRTYPE:
 Block           119366
Intersection     63313
Alley              235
Name: ADDRTYPE, dtype: int64
```

```
LIGHTCOND:
 Daylight                    113850
Dark - Street Lights On      47550
Unknown                      10448
Dusk                          5772
Dawn                          2454
Dark - No Street Lights       1462
Dark - Street Lights Off      1157
Other                          210
Dark - Unknown Lighting         11
Name: LIGHTCOND, dtype: int64

 WEATHER:
 Clear                       109065
Raining                      32649
Overcast                     27189
Unknown                      11637
Snowing                        881
Other                          746
Fog/Smog/Smoke                 556
Sleet/Hail/Freezing Rain       112
Blowing Sand/Dirt               49
Severe Crosswind                25
Partly Cloudy                    5
Name: WEATHER, dtype: int64

 JUNCTIONTYPE:
 Mid-Block (not related to intersection)              86613
At Intersection (intersection related)               61221
Mid-Block (but intersection related)                 22341
Driveway Junction                                    10519
At Intersection (but not related to intersection)     2055
Ramp Junction                                          160
Unknown                                                  5
Name: JUNCTIONTYPE, dtype: int64

 SDOT_COLDESC:
 MOTOR VEHICLE STRUCK MOTOR VEHICLE, FRONT END AT ANGLE          83027
MOTOR VEHICLE STRUCK MOTOR VEHICLE, REAR END                    52488
MOTOR VEHICLE STRUCK MOTOR VEHICLE, LEFT SIDE SIDESWIPE          9776
MOTOR VEHICLE RAN OFF ROAD - HIT FIXED OBJECT                    8699
MOTOR VEHCILE STRUCK PEDESTRIAN                                  6368
MOTOR VEHICLE STRUCK MOTOR VEHICLE, LEFT SIDE AT ANGLE           5614
MOTOR VEHICLE STRUCK OBJECT IN ROAD                              4581
NOT ENOUGH INFORMATION / NOT APPLICABLE                          3112
MOTOR VEHICLE STRUCK PEDALCYCLIST, FRONT END AT ANGLE            3030
MOTOR VEHICLE STRUCK MOTOR VEHICLE, RIGHT SIDE SIDESWIPE         1567
MOTOR VEHICLE STRUCK MOTOR VEHICLE, RIGHT SIDE AT ANGLE          1370
```

```
PEDALCYCLIST STRUCK MOTOR VEHICLE FRONT END AT ANGLE             1292
MOTOR VEHICLE OVERTURNED IN ROAD                                  472
MOTOR VEHICLE STRUCK PEDALCYCLIST, REAR END                       180
PEDALCYCLIST STRUCK MOTOR VEHICLE LEFT SIDE SIDESWIPE             177
MOTOR VEHICLE RAN OFF ROAD - NO COLLISION                         160
PEDALCYCLIST STRUCK MOTOR VEHICLE REAR END                        134
MOTOR VEHICLE STRUCK PEDALCYCLIST, LEFT SIDE SIDESWIPE            122
DRIVERLESS VEHICLE RAN OFF ROAD - HIT FIXED OBJECT                106
DRIVERLESS VEHICLE STRUCK MOTOR VEHICLE FRONT END AT ANGLE        103
MOTOR VEHICLE STRUCK TRAIN                                        101
DRIVERLESS VEHICLE STRUCK MOTOR VEHICLE REAR END                   92
PEDALCYCLIST STRUCK PEDESTRIAN                                     74
PEDALCYCLIST OVERTURNED IN ROAD                                    67
DRIVERLESS VEHICLE STRUCK MOTOR VEHICLE LEFT SIDE AT ANGLE         53
PEDALCYCLIST STRUCK MOTOR VEHICLE RIGHT SIDE SIDESWIPE             50
PEDALCYCLIST STRUCK OBJECT IN ROAD                                 23
MOTOR VEHICLE STRUCK PEDALCYCLIST, RIGHT SIDE SIDESWIPE            16
DRIVERLESS VEHICLE STRUCK MOTOR VEHICLE RIGHT SIDE AT ANGLE        12
PEDALCYCLIST STRUCK MOTOR VEHICLE LEFT SIDE AT ANGLE                9
DRIVERLESS VEHICLE STRUCK PEDESTRIAN                                8
PEDALCYCLIST STRUCK PEDALCYCLIST REAR END                           7
DRIVERLESS VEHICLE STRUCK MOTOR VEHICLE RIGHT SIDE SIDESWIPE        6
PEDALCYCLIST STRUCK PEDALCYCLIST FRONT END AT ANGLE                 4
PEDALCYCLIST RAN OFF ROAD - HIT FIXED OBJECT                        4
DRIVERLESS VEHICLE STRUCK MOTOR VEHICLE LEFT SIDE SIDESWIPE         4
DRIVERLESS VEHICLE STRUCK OBJECT IN ROADWAY                         3
PEDALCYCLIST STRUCK MOTOR VEHICLE RIGHT SIDE AT ANGLE               2
DRIVERLESS VEHICLE RAN OFF ROAD - NO COLLISION                      1
Name: SDOT_COLDESC, dtype: int64

 ROADCOND:
 Dry               122159
Wet                46720
Unknown            11521
Ice                 1178
Snow/Slush           978
Other                123
Standing Water       108
Sand/Mud/Dirt         67
Oil                   60
Name: ROADCOND, dtype: int64
```

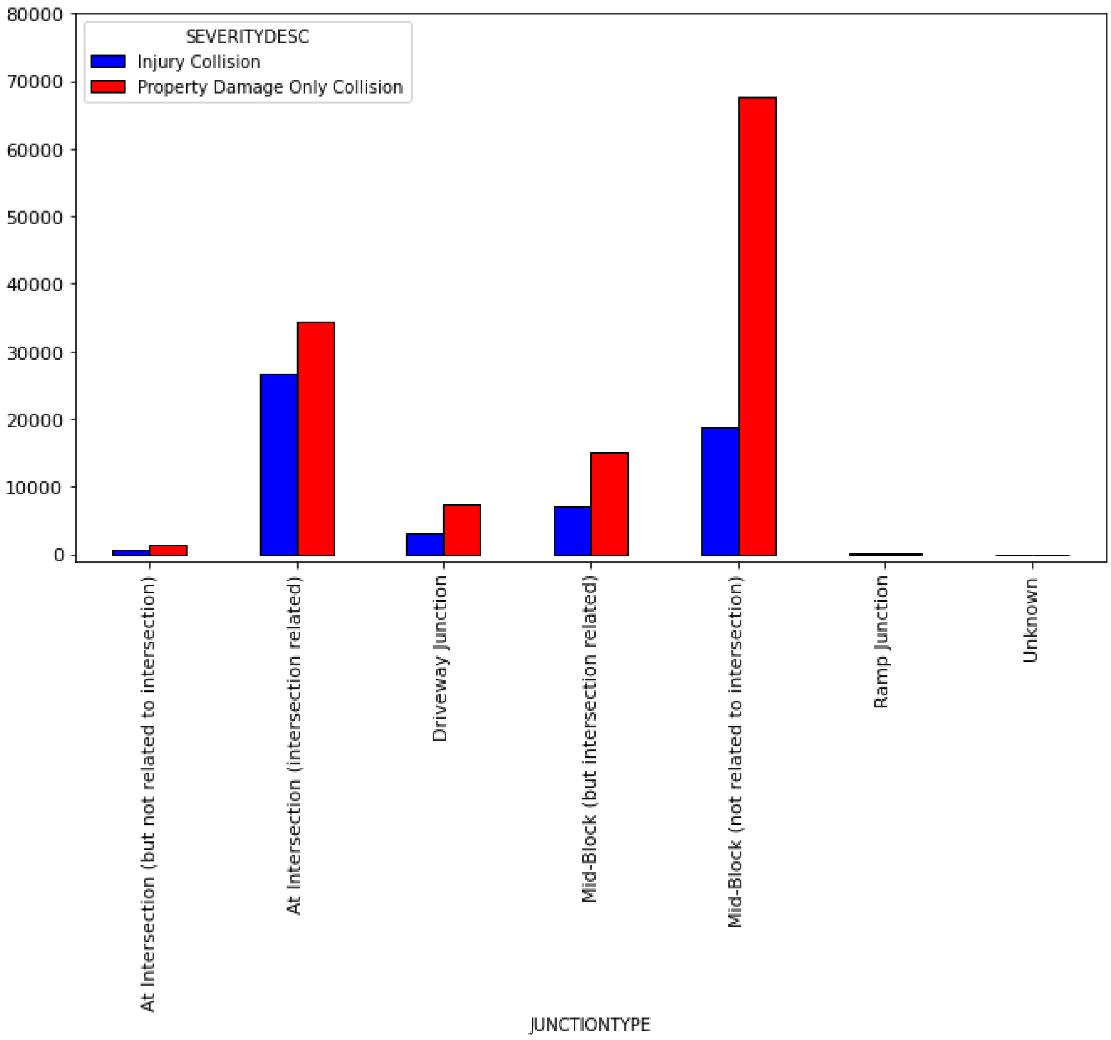
0.2 Exploring the Data

Now we have cleaned the data let’s create some visuals to see what we are dealing with

```
[19]:
```

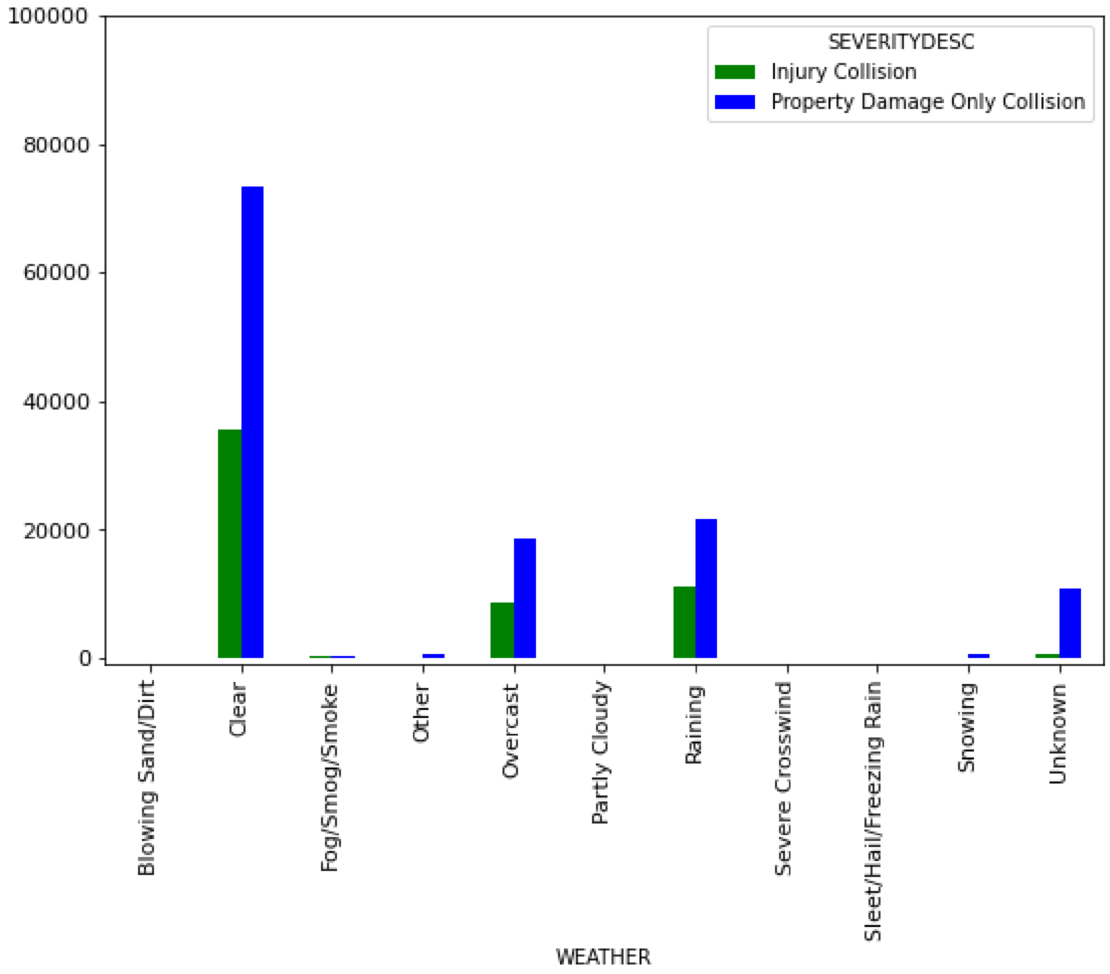
```
#This will look at the effects of the junction type
main_df.groupby(['JUNCTIONTYPE', 'SEVERITYDESC']).agg('size').unstack().
 ↪plot(kind = 'bar', legend=True, figsize=(11, 6), fontsize=11,␣
 ↪edgecolor='black',color=['blue', 'red',])
plt.ylim((-1000,80000))
```

```
[19]: (-1000.0, 80000.0)
```



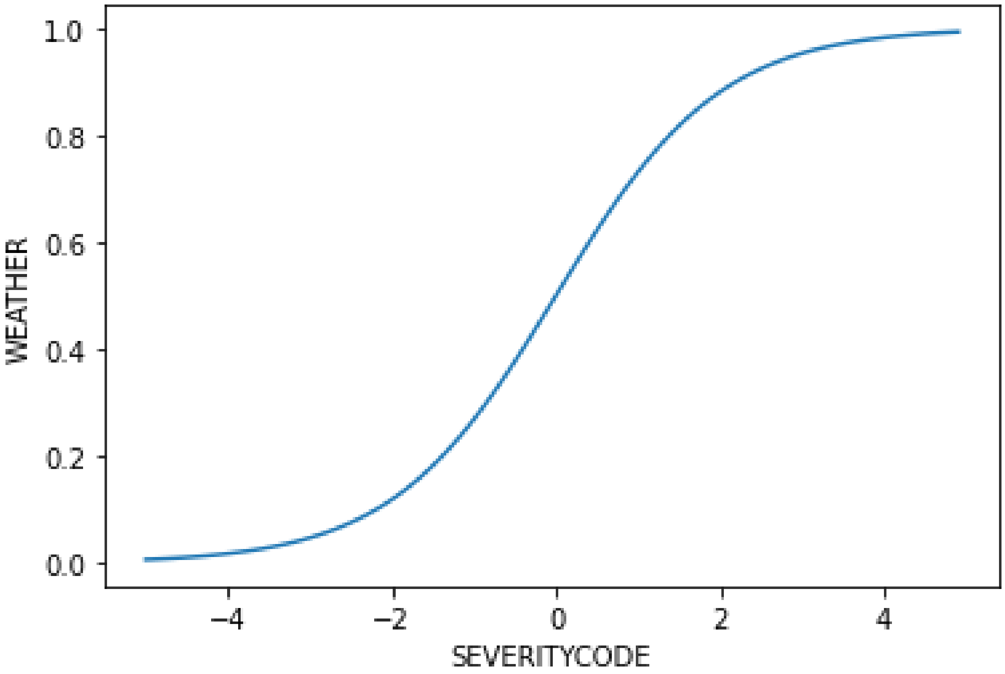
```
[20]: #This looks at the effects of weather
main_df.groupby(['WEATHER', 'SEVERITYDESC']).agg('size').unstack().plot(kind =␣
 ↪'bar', figsize=(9,6), legend=True, fontsize=11, color=['green', 'blue'])
plt.ylim((-1000, 100000))
```

```
[20]: (-1000.0, 100000.0)
```



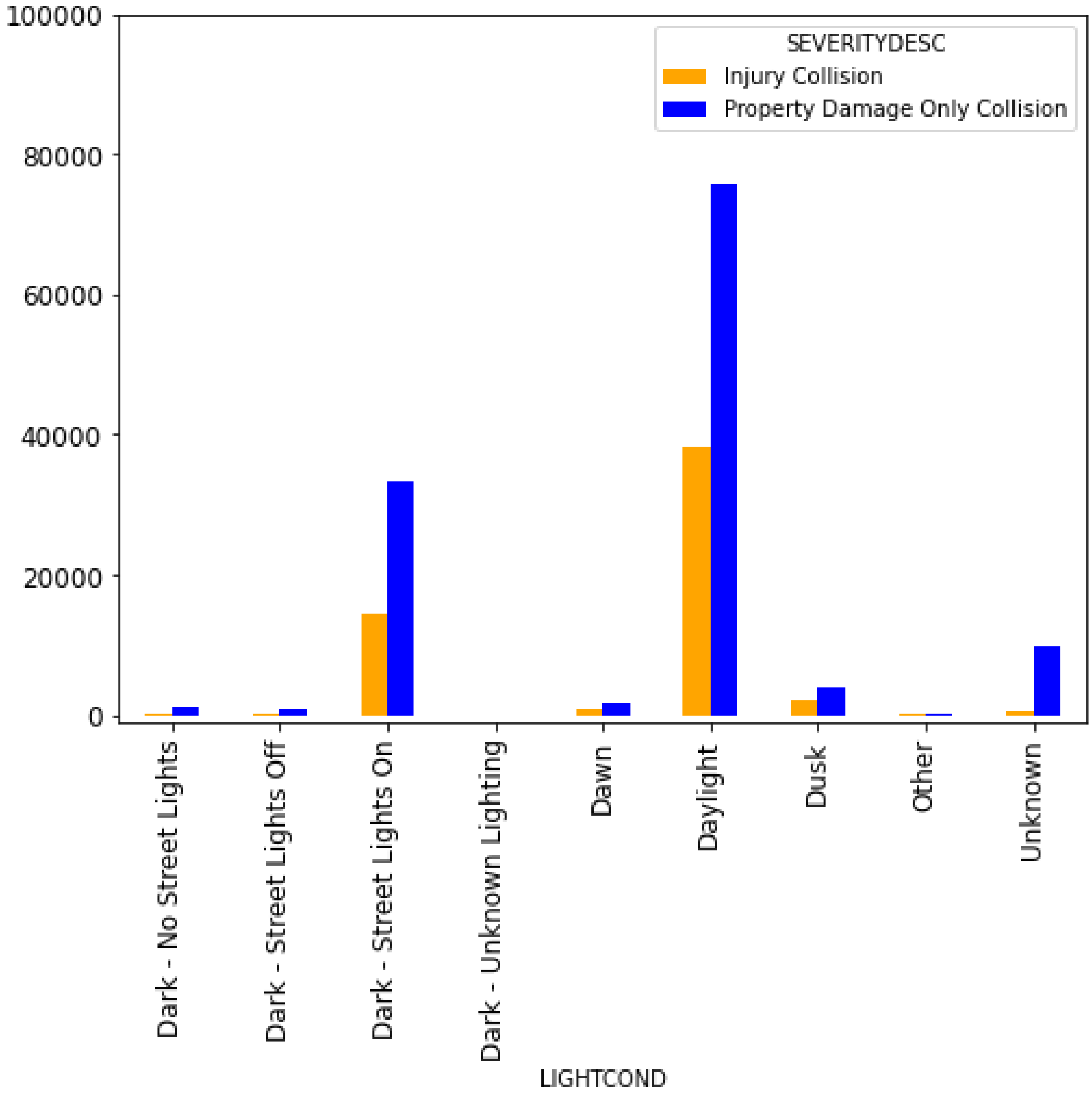
[21]:

```
X = np.arange(-5.0, 5.0, 0.1)
Y = 1.0 / (1.0 + np.exp(-X))

plt.plot(X,Y)
plt.ylabel('WEATHER')
plt.xlabel('SEVERITYCODE')
plt.show()
```

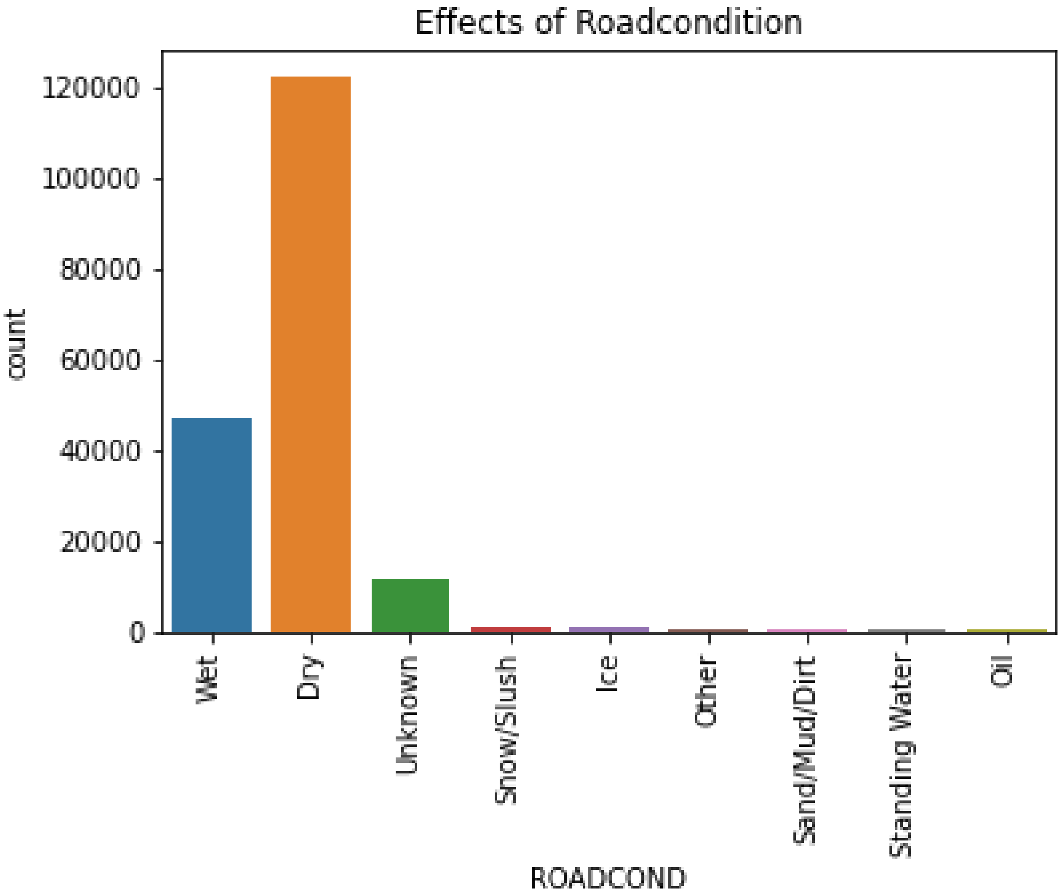


```
[22]: #This looks at the effect of light condition
      main_df.groupby(['LIGHTCOND', 'SEVERITYDESC']).agg('size').unstack().plot(kind␣
       ↪= 'bar', figsize=(8,6), legend=True, fontsize=12, color=['orange', 'blue'])

      plt.ylim((-1000, 100000))
```

```
[22]: (-1000.0, 100000.0)
```



[23]:
```
#This looks at the effect of road conditions
sns.countplot(x = "ROADCOND" , data = main_df, )
plt.title("Effects of Roadcondition")
plt.xticks(rotation='vertical')
plt.show()
```



```
[24]: #A breakdown in roadconditions numerically
      main_df ['ROADCOND'].value_counts()
```

```
[24]: Dry                122159
      Wet                 46720
      Unknown             11521
      Ice                  1178
      Snow/Slush            978
      Other                 123
      Standing Water        108
      Sand/Mud/Dirt          67
      Oil                    60
      Name: ROADCOND, dtype: int64
```

```
[26]: from sklearn.model_selection import train_test_split

      train, test = train_test_split(main_df, test_size=0.2)
```

```
[ ]:
```