Data Understanding

In this project I will load a dataset regarding to collisions provided by Coursera – IBM Data Science Course.

Load the Collisions data

In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
df = pd.read_csv('https://s3.us.cloud-object-storage.appdomain.cloud/cf-courses-data/CognitiveClass/DP0701EN/version-2/Data-Collisions.csv')
df.head()
```

```
/opt/conda/envs/Python36/lib/python3.6/site-packages/IPython/core/interactiveshell.py:3020: DtypeWarning: Columns (33) have mixed types. Specify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

Out[2]:

	SEVERITYCODE	X	Y	OBJECTID	INCKEY	COLDETKEY	REPORTNO	STATUS	ADDRTYPE	INTKEY	...	ROADCOND	LIGHTCOND	PEDROWNOTGRNT	SDOTCOLNUM	SPEEDING	ST_COLCODE	ST_COLDESC	SEGLANEKEY	CROSSWALKKEY	HITPARKI
0	2	-122.323148	47.703140	1	1307	1307	3502005	Matched	Intersection	37475.0	...	Wet	Daylight	NaN	NaN	NaN	10	Entering at angle	0	0	
1	1	-122.347294	47.647172	2	52200	52200	2607959	Matched	Block	NaN	...	Wet	Dark - Street Lights On	NaN	6354039.0	NaN	11	From same direction - both going straight - bo...	0	0	
2	1	-122.334540	47.607871	3	26700	26700	1482393	Matched	Block	NaN	...	Dry	Daylight	NaN	4323031.0	NaN	32	One parked--one moving	0	0	
3	1	-122.334803	47.604803	4	1144	1144	3503937	Matched	Block	NaN	...	Dry	Daylight	NaN	NaN	NaN	23	From same direction - all others	0	0	
4	2	-122.306426	47.545739	5	17700	17700	1807429	Matched	Intersection	34387.0	...	Wet	Daylight	NaN	4028032.0	NaN	10	Entering at angle	0	0	

5 rows × 38 columns

In [3]:
```
print('This dataset contains ', df.shape[0], 'rows.')
```

```
This dataset contains  194673 rows.
```

In [4]:
```
print('This dataset contains ', df.shape[1], 'columns.')
```

```
This dataset contains  38 columns.
```

In [5]:
```
df.dtypes
```

Out[5]:
```
SEVERITYCODE        int64
X                 float64
Y                 float64
OBJECTID            int64
INCKEY              int64
COLDETKEY           int64
REPORTNO           object
STATUS             object
ADDRTYPE           object
INTKEY            float64
LOCATION           object
EXCEPTRSNCODE      object
EXCEPTRSNDESC      object
SEVERITYCODE.1      int64
SEVERITYDESC       object
COLLISIONTYPE      object
PERSONCOUNT         int64
PEDCOUNT            int64
PEDCYLCOUNT         int64
VEHCOUNT            int64
INCDATE            object
INCDTTM            object
JUNCTIONTYPE       object
SDOT_COLCODE        int64
SDOT_COLDESC       object
INATTENTIONIND     object
UNDERINFL          object
WEATHER            object
ROADCOND           object
LIGHTCOND          object
PEDROWNOTGRNT      object
SDOTCOLNUM        float64
SPEEDING           object
ST_COLCODE         object
ST_COLDESC         object
SEGLANEKEY          int64
CROSSWALKKEY        int64
HITPARKEDCAR       object
dtype: object
```

SEVERITYCODE is target variable for model development.

In [6]:
```
df['SEVERITYCODE'].value_counts()
```

Out[6]:
```
1    136485
2     58188
Name: SEVERITYCODE, dtype: int64
```

In [7]:
```
df.columns
```

Out[7]:
```
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')
```

I will remove other unnecessary variables.

In [8]:
```
df_collisions = df.drop(columns=['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', 'PEDROWNOTGRNT',
                                 'SDOTCOLNUM', 'SPEEDING', 'ST_COLCODE', 'ST_COLDESC', 'SEGLANEKEY', 'CROSSWALKKEY', 'HITPARKEDCAR'])
```

In [9]:
```
df_collisions.head()
```

Out[9]:

	SEVERITYCODE	WEATHER	ROADCOND	LIGHTCOND
0	2	Overcast	Wet	Daylight
1	1	Raining	Wet	Dark - Street Lights On
2	1	Overcast	Dry	Daylight
3	1	Clear	Dry	Daylight
4	2	Raining	Wet	Daylight

Replace null value

```
In [10]: df_collisions['WEATHER'] = df_collisions['WEATHER'].fillna(value='Unknown')
         df_collisions['WEATHER'].value_counts()
```

```
Out[10]: Clear                     111135
         Raining                    33145
         Overcast                   27714
         Unknown                    20172
         Snowing                      907
         Other                        832
         Fog/Smog/Smoke               569
         Sleet/Hail/Freezing Rain     113
         Blowing Sand/Dirt             56
         Severe Crosswind              25
         Partly Cloudy                  5
         Name: WEATHER, dtype: int64
```

```
In [11]: df_collisions['ROADCOND'] = df_collisions['ROADCOND'].fillna(value='Other')
         df_collisions['ROADCOND'].value_counts()
```

```
Out[11]: Dry              124510
         Wet               47474
         Unknown           15078
         Other              5144
         Ice                1209
         Snow/Slush         1004
         Standing Water      115
         Sand/Mud/Dirt        75
         Oil                  64
         Name: ROADCOND, dtype: int64
```

```
In [12]: df_collisions['LIGHTCOND'] = df_collisions['LIGHTCOND'].fillna(value='Other')
         df_collisions['LIGHTCOND'].value_counts()
```

```
Out[12]: Daylight                   116137
         Dark - Street Lights On     48507
         Unknown                    13473
         Dusk                        5902
         Other                       5405
         Dawn                        2502
         Dark - No Street Lights     1537
         Dark - Street Lights Off    1199
         Dark - Unknown Lighting       11
         Name: LIGHTCOND, dtype: int64
```

Converting string value to integer

Converting string value to number value for **WEATHER** attribute.

String Value	Integer Value
Clear	0
Raining	1
Overcast	2
Unknown	
Other	
Fog/Smog/Smoke	
Sleet/Hail/Freezing Rain	3
Blowing Sand/Dirt	
Severe Crosswind	
Partly Cloudy	
Snowing	4

```
In [28]: df_collisions['WEATHER'].replace(to_replace=['Clear','Raining','Overcast','Unknown','Snowing','Other','Fog/Smog/Smoke','Sleet/Hail/Freezing Rain','Blowing Sand/Dirt','Severe Crosswind','Partly Cloudy'],
                                          value=[0,1,2,3,4,3,3,3,3,3,3],inplace=True)
         df_collisions['WEATHER'].value_counts()
```

```
Out[28]: 0    111135
         1     33145
         2     27714
         3     21772
         4       907
         Name: WEATHER, dtype: int64
```

Converting string value to number value for **ROADCOND** attribute.

String Value	Integer Value
Dry	0
Wet	1
Unknown	
Other	
Ice	
Snow/Slush	2
Standing Water	
Sand/Mud/Dirt	
Oil	

```
In [29]: df_collisions['ROADCOND'].replace(to_replace=['Dry','Wet','Overcast','Unknown','Other','Ice','Snow/Slush','Standing Water','Sand/Mud/Dirt','Oil'],
                                           value=[0,1,2,2,2,2,2,2,2,2],inplace=True)
         df_collisions['ROADCOND'].value_counts()
```

```
Out[29]: 0    124510
         1     47474
         2     22689
         Name: ROADCOND, dtype: int64
```

Converting string value to number value for **LIGHTCOND** attribute.

String Value	Integer Value
Daylight	0
Dark - Street Lights On	1
Dark - No Street Lights	2
Dark - Street Lights Off	3
Dusk	4
Dawn	5
Unknown	
Other	6
Dark - Unknown Lighting	

```
In [31]: df_collisions['LIGHTCOND'].replace(to_replace=['Daylight','Dark - Street Lights On','Dark - No Street Lights','Dark - Street Lights Off','Dusk','Dawn','Unknown','Other','Dark - Unknown Lighting'],
                                           value=[0,1,2,3,4,5,6,6,6],inplace=True)
         df_collisions['LIGHTCOND'].value_counts()
```

```
Out[31]: 0    116137
         1     48507
         6     18889
         4      5902
         5      2502
         2      1537
         3      1199
         Name: LIGHTCOND, dtype: int64
```