Aviation Data

My project aims to predict the risk involved in flight

Data Source and Data Exploration

This data comes from a kaggel competition which provides details in Aviation. Source : https://www.kaggle.com/khsamaha/aviation-accident-database-synopses

The target variable shows that there are some outliers in the data, which are injuries that occurred by the type of the airplane used in the dataset

I used 21 columns for my analysis, which included variables about:

- Injuries
- Make and Model
- Purpose of flight and many more

In [148…

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from numbers import Number
import warnings
warnings.filterwarnings('ignore')
```

In [4]:

```
df = pd.read_csv("AviationData.csv",encoding="latin-1")
df
```

Out[4]:

	Event.Id	Investigation.Type	Accident.Number	Event.Date	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
1	20200508X55730	Accident	CEN20CA176	5/8/2020	San Antonio, TX
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
4	20200504X54503	Incident	ENG20IA031	5/3/2020	Clewiston, FL
…	…	…	…	…	…
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH

84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

84983 rows × 31 columns

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 84983 entries, 0 to 84982
Data columns (total 31 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Event.Id                84983 non-null  object 
 1   Investigation.Type      84978 non-null  object 
 2   Accident.Number         84983 non-null  object 
 3   Event.Date              84983 non-null  object 
 4   Location                84905 non-null  object 
 5   Country                 84476 non-null  object 
 6   Latitude                30658 non-null  float64
 7   Longitude               30649 non-null  float64
 8   Airport.Code            48005 non-null  object 
 9   Airport.Name            50746 non-null  object 
 10  Injury.Severity         84983 non-null  object 
 11  Aircraft.Damage         82096 non-null  object 
 12  Aircraft.Category       28257 non-null  object 
 13  Registration.Number     80837 non-null  object 
 14  Make                    84913 non-null  object 
 15  Model                   84884 non-null  object 
 16  Amateur.Built           84303 non-null  object 
 17  Number.of.Engines       79684 non-null  float64
 18  Engine.Type             80505 non-null  object 
 19  FAR.Description         27936 non-null  object 
 20  Schedule                12091 non-null  object 
 21  Purpose.of.Flight       79701 non-null  object 
 22  Air.Carrier             4246 non-null   object 
 23  Total.Fatal.Injuries    57165 non-null  float64
 24  Total.Serious.Injuries  54473 non-null  float64
 25  Total.Minor.Injuries    55702 non-null  float64
 26  Total.Uninjured         70099 non-null  float64
 27  Weather.Condition       81585 non-null  object 
 28  Broad.Phase.of.Flight   78382 non-null  object 
 29  Report.Status           84983 non-null  object 
 30  Publication.Date        70326 non-null  object 
dtypes: float64(7), object(24)
memory usage: 20.1+ MB
```

In [6]:
```
df.describe()
```

Out[6]:

	Latitude	Longitude	Number.of.Engines	Total.Fatal.Injuries	Total.Serious
count	30658.000000	30649.000000	79684.000000	57165.000000	5447
mean	37.429864	-93.251873	1.148060	0.854456	
std	12.575431	39.908402	0.447671	6.331490	
min	-78.016945	-178.676111	0.000000	0.000000	
25%	33.269167	-114.795277	1.000000	0.000000	
50%	38.085416	-94.298611	1.000000	0.000000	
75%	42.500903	-81.605833	1.000000	1.000000	
max	89.218056	435.833334	8.000000	349.000000	11

In [ ]:

```
#check total nulls in each column
df.isna().sum()
```

Out[ ]:

```
Event.Id                    0
Investigation.Type          5
Accident.Number             0
Event.Date                  0
Location                   78
Country                   507
Latitude                54325
Longitude               54334
Airport.Code            36978
Airport.Name            34237
Injury.Severity             0
Aircraft.Damage          2887
Aircraft.Category       56726
Registration.Number      4146
Make                       70
Model                      99
Amateur.Built             680
Number.of.Engines        5299
Engine.Type              4478
FAR.Description         57047
Schedule                72892
Purpose.of.Flight        5282
Air.Carrier             80737
Total.Fatal.Injuries    27818
Total.Serious.Injuries  30510
Total.Minor.Injuries    29281
Total.Uninjured         14884
Weather.Condition        3398
Broad.Phase.of.Flight    6601
Report.Status               0
Publication.Date        14657
dtype: int64
```

In [ ]:

```
#drop duplicates
df = df.drop_duplicates()
df
```

Out[ ]:

	Event.Id	Investigation.Type	Accident.Number	Event.Date	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
1	20200508X55730	Accident	CEN20CA176	5/8/2020	San Antonio, TX
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
4	20200504X54503	Incident	ENG20IA031	5/3/2020	Clewiston, FL
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

84983 rows × 31 columns

In [ ]:

```
#check for columns
df.columns
```

Out[ ]:

```
Index(['Event.Id', 'Investigation.Type', 'Accident.Number', 'Event.Date',
       'Location', 'Country', 'Latitude', 'Longitude', 'Airport.Code',
       'Airport.Name', 'Injury.Severity', 'Aircraft.Damage',
       'Aircraft.Category', 'Registration.Number', 'Make', 'Model',
       'Amateur.Built', 'Number.of.Engines', 'Engine.Type', 'FAR.Description',
       'Schedule', 'Purpose.of.Flight', 'Air.Carrier', 'Total.Fatal.Injuries',
       'Total.Serious.Injuries', 'Total.Minor.Injuries', 'Total.Uninjured',
       'Weather.Condition', 'Broad.Phase.of.Flight', 'Report.Status',
       'Publication.Date'],
      dtype='object')
```

In [10]:

```
columns_not_wanted = ['Registration.Number','Amateur.Built', 'Number.of.Engine
       'Schedule','Air.Carrier','Broad.Phase.of.Flight','Publication.Date']
df[columns_not_wanted]
```

Out[10]:

	Registration.Number	Amateur.Built	Number.of.Engines	Engine.Type	FAR.Desc

Out[10]:

	Registration.Number	Amateur.Built	Number.of.Engines	Engine.Type	FAR.Desc
0	N318WH	Yes	NaN	NaN	Part 91: A
1	N3238G	No	NaN	NaN	Part 91: A
2	N25HE	No	1.0	NaN	Part 91: A
3	N11457	No	1.0	NaN	Part 91: A
4	N1WT	No	1.0	Turbo Shaft	Part 91: A
...	...	...	...	...	
84978	N15NY	No	NaN	NaN	
84979	N1168J	No	1.0	Reciprocating	
84980	N5142R	No	1.0	Reciprocating	
84981	N5069P	No	1.0	Reciprocating	
84982	NC6404	No	1.0	Reciprocating	

84983 rows × 9 columns

In [ ]:
```
#drop columns that are not needed for investigations
df=df.drop(columns=columns_not_wanted)
df
```

Out[ ]:

	Event.Id	Investigation.Type	Accident.Number	Event.Date	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
1	20200508X55730	Accident	CEN20CA176	5/8/2020	San Antonio, TX
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
4	20200504X54503	Incident	ENG20IA031	5/3/2020	Clewiston, FL
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH

84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

84983 rows × 22 columns

In [12]:
```
df.columns.value_counts().sum()
```

Out[12]: 22

In [20]:
```
df.columns=df.columns.str.replace('.','')
```

In [ ]:
```
"""
dealing with missing values for floats decided to add zero
The data integrity is import thus report giving it the median,mode or mean can
lead to distortion of information
"""
df['Total.Fatal.Injuries'].fillna(0,inplace=True)
df['Total.Serious.Injuries'].fillna(0,inplace=True)
df['Total.Minor.Injuries'].fillna(0,inplace=True)
df['Total.Uninjured'].fillna(0,inplace=True)
```

In [ ]:
```
"""
for strings or object decided to give it missing values when it comes to clean
for floats decided to give zero for longitude and latitude since we have to fo
given location
"""
df['Country'].fillna('missing',inplace=True)
df["AirportCode"].fillna("missing",inplace=True)
df["Make"].fillna('missing',inplace=True)
df["Model"].fillna('missing',inplace=True)
df["AirportName"].fillna('missing',inplace=True)
df["WeatherCondition"].fillna('missing',inplace=True)
df["AircraftCategory"].fillna('missing',inplace=True)
df["AircraftDamage"].fillna('missing',inplace=True)
df["PurposeofFlight"].fillna('missing',inplace=True)
df["Latitude"].fillna(0,inplace=True)
df["Longitude"].fillna(0,inplace=True)
```

In [142…
```
#check if the columns are well aligned
df
```

Out[142 | EventId | InvestigationType | AccidentNumber | EventDate | Location

Out[142...

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	05-09-2020	Haskell, OK
1	20200507X60215	Accident	CEN20CA174	05-06-2020	Gonazales, TX
2	20200509X85739	Accident	ERA20CA175	05-03-2020	Shirley, NY
3	20200502X81549	Accident	CEN20LA168	05-02-2020	PALMYRA, IL
4	20200502X73540	Accident	CEN20LA167	05-02-2020	HOUSTON, TX
...	...	...	...	...	...
77236	20041105X01764	Accident	CHI79FA064	08-02-1979	Canton, OH
77237	20001218X45448	Accident	LAX96LA321	19-06-1977	EUREKA, CA
77238	20061025X01555	Accident	NYC07LA005	30-08-1974	Saltville, VA
77239	20001218X45447	Accident	LAX94LA336	19-07-1962	BRIDGEPORT, CA
77240	20001218X45444	Accident	SEA87LA080	24-10-1948	MOOSE CREEK, ID

77241 rows × 21 columns

Remove rows with missing values

In [ ]:

```
#removed missing rows from weather condition
for x in df.index:
 if df.loc[x,"WeatherCondition"] == 'missing':
     df.drop(x,inplace=True)
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY

6	20200502X81549	Accident	CEN20LA168	5/2/2020	PALMYRA, IL
7	20200502X73540	Accident	CEN20LA167	5/2/2020	HOUSTON, TX
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

81585 rows × 22 columns

In [ ]:
```
#removed missing rows from make
for x in df.index:
 if df.loc[x,"Make"] == 'missing':
     df.drop(x,inplace=True)
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
6	20200502X81549	Accident	CEN20LA168	5/2/2020	PALMYRA, IL
7	20200502X73540	Accident	CEN20LA167	5/2/2020	HOUSTON, TX
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA

84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

81558 rows × 22 columns

In [ ]:

```
#removed missing rows from model
for x in df.index:
 if df.loc[x,"Model"] == 'missing':
     df.drop(x,inplace=True)
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
6	20200502X81549	Accident	CEN20LA168	5/2/2020	PALMYRA, IL
7	20200502X73540	Accident	CEN20LA167	5/2/2020	HOUSTON, TX
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

81524 rows × 22 columns

In [ ]:
```
#removed missing rows from aircraft damage
for x in df.index:
 if df.loc[x,"AircraftDamage"] == 'missing':
     df.drop(x,inplace=True)
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
6	20200502X81549	Accident	CEN20LA168	5/2/2020	PALMYRA, IL
7	20200502X73540	Accident	CEN20LA167	5/2/2020	HOUSTON, TX
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

79501 rows × 22 columns

In [ ]:
```
#removed missing rows from purpose of flight
for x in df.index:
 if df.loc[x,"PurposeofFlight"] == 'missing':
     df.drop(x,inplace=True)
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK

2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
6	20200502X81549	Accident	CEN20LA168	5/2/2020	PALMYRA, IL
7	20200502X73540	Accident	CEN20LA167	5/2/2020	HOUSTON, TX
...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

77286 rows × 22 columns

In [ ]:
```
#removed missing rows from country
for x in df.index:
 if df.loc[x,"Country"] == 'missing':
     df.drop(x,inplace=True)
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	5/9/2020	Haskell, OK
1	20200508X55730	Accident	CEN20CA176	5/8/2020	San Antonio, TX
2	20200507X60215	Accident	CEN20CA174	5/6/2020	Gonazales, TX
3	20200509X85739	Accident	ERA20CA175	5/3/2020	Shirley, NY
4	20200504X54503	Incident	ENG20IA031	5/3/2020	Clewiston, FL

...	...	...	...	...	...
84978	20041105X01764	Accident	CHI79FA064	8/2/1979	Canton, OH
84979	20001218X45448	Accident	LAX96LA321	6/19/1977	EUREKA, CA
84980	20061025X01555	Accident	NYC07LA005	8/30/1974	Saltville, VA
84981	20001218X45447	Accident	LAX94LA336	7/19/1962	BRIDGEPORT, CA
84982	20001218X45444	Accident	SEA87LA080	10/24/1948	MOOSE CREEK, ID

84983 rows × 22 columns

In [ ]:
```
#removed missing rows from location
df['Location'].fillna('missing',inplace=True)
for x in df.index:
    if df.loc[x,"Location"] == 'missing':
     df.drop(x,inplace=True)
```

In [77]:
```
#Check info
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 77241 entries, 0 to 77285
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   index                 77241 non-null  int64  
 1   EventId               77241 non-null  object 
 2   InvestigationType     77241 non-null  object 
 3   AccidentNumber        77241 non-null  object 
 4   EventDate             77241 non-null  object 
 5   Location              77241 non-null  object 
 6   Country               77241 non-null  object 
 7   Latitude              77241 non-null  float64
 8   Longitude             77241 non-null  float64
 9   InjurySeverity        77241 non-null  object 
 10  AircraftDamage        77241 non-null  object 
 11  AircraftCategory      77241 non-null  object 
 12  Make                  77241 non-null  object 
 13  Model                 77241 non-null  object 
 14  PurposeofFlight       77241 non-null  object 
 15  TotalFatalInjuries    77241 non-null  float64
 16  TotalSeriousInjuries  77241 non-null  float64
 17  TotalMinorInjuries    77241 non-null  float64
 18  TotalUninjured        77241 non-null  float64
 19  WeatherCondition      77241 non-null  object 
 20  ReportStatus          77241 non-null  object 
dtypes: float64(6), int64(1), object(14)
```

```
memory usage: 15.5+ MB
```

In [78]:
```
#reset index of the data frame
df.reset_index(inplace=True)
```

In [ ]:
```
#drop column due to more than 30,000 missing values which is not sustainable f
df.drop(columns="AirportCode",inplace=True)
df.drop(columns="AirportName",inplace=True)
```

In [ ]:
```
#check the number of columns left
df.columns.value_counts().sum()
```

Out[ ]:
```
21
```

In [ ]:
```
#changed the date to start with the day first instead of month
df['EventDate'] = pd.to_datetime(df['EventDate'],dayfirst=True)
```

In [ ]:
```
#change the formart of the date in order to reverse it
df['EventDate']=df['EventDate'].dt.strftime('%d-%m-%Y')
```

In [141…
```
df['Year'] = pd.to_datetime(df['EventDate'], errors='coerce').dt.year
```

In [80]:
```
df.drop(columns="level_0",inplace=True)
df
```

Out[80]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	05-09-2020	Haskell, OK
1	20200507X60215	Accident	CEN20CA174	05-06-2020	Gonazales, TX
2	20200509X85739	Accident	ERA20CA175	05-03-2020	Shirley, NY
3	20200502X81549	Accident	CEN20LA168	05-02-2020	PALMYRA, IL
4	20200502X73540	Accident	CEN20LA167	05-02-2020	HOUSTON, TX
...	...	...	...	...	...
77236	20041105X01764	Accident	CHI79FA064	08-02-1979	Canton, OH
77237	20001218X45448	Accident	LAX96LA321	19-06-1977	EUREKA, CA

77238	20061025X01555	Accident	NYC07LA005	30-08-1974	Saltville, VA
77239	20001218X45447	Accident	LAX94LA336	19-07-1962	BRIDGEPORT, CA
77240	20001218X45444	Accident	SEA87LA080	24-10-1948	MOOSE CREEK, ID

77241 rows × 20 columns

In [ ]:
```
#remove any spaces from rows and align them
df['Model'].str.strip()
```

Out[ ]:
```
0        TITAN TORNADO S
1                   T240
2                    QCF
3                 YAK 52
4                    369
               ...
77281                501
77282                112
77283               172M
77284          PA24-180
77285             108-3
Name: Model, Length: 77286, dtype: object
```

In [144…
```
df.to_csv("Cleaned_AviationData.csv",index=False)
```

In [169…
```
df.to_excel("Cleaned_AviationData.xlsx",index=False)
```

Data Exploratory

- check for outliers
- compare data

In [81]:
```
df["InvestigationType"].value_counts()
```

Out[81]:
```
Accident    75957
Incident     1284
Name: InvestigationType, dtype: int64
```

In [ ]:
```
#use a pychart
# showing the names of Investigations in the dataset
df['InvestigationType'].unique()
```

Out[ ]:
```
array(['Accident', 'Incident'], dtype=object)
```

In [84]:
```
#showing all the Investigation frequency
```

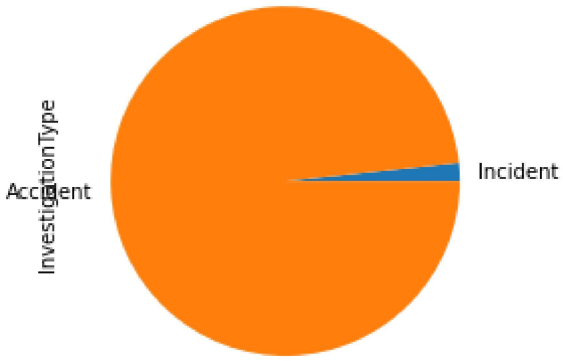
```
df['InvestigationType'].value_counts()
```

Out[84]:
```
Accident    75957
Incident     1284
Name: InvestigationType, dtype: int64
```

In [97]:
```
#showing all the Investigation Type with their number of frquency in pie Graph

df['InvestigationType'].value_counts(ascending=True).plot.pie()
```

Out[97]:
```
<AxesSubplot:ylabel='InvestigationType'>
```



In [99]:
```
df['AircraftCategory'].unique()
```

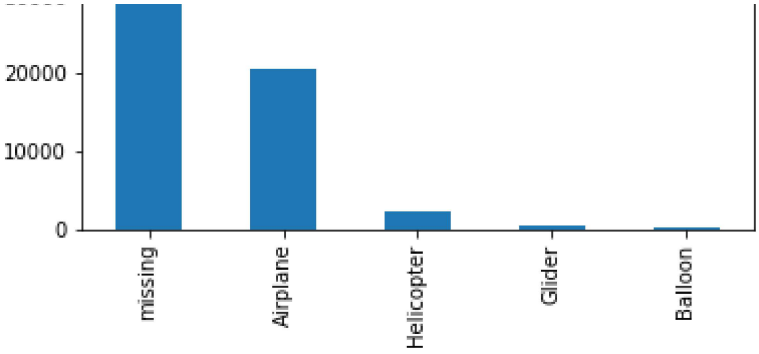
Out[99]:
```
array(['Airplane', 'Helicopter', 'Glider', 'Balloon', 'Gyroplane',
       'Weight-Shift', 'Powered Parachute', 'Ultralight', 'Blimp',
       'Gyrocraft', 'missing', 'Powered-Lift', 'Unknown'], dtype=object)
```

In [ ]:
```
#showing all the aircraft category frequency
df['AircraftCategory'].value_counts().head(8)
```

Out[ ]:
```
missing         53587
Airplane        20469
Helicopter       2288
Glider            446
Balloon           116
Weight-Shift       98
Gyrocraft          88
Gyroplane          60
Name: AircraftCategory, dtype: int64
```

In [ ]:
```
#df['AircraftCategory'].value_counts().head(5).plot.bar()
```

Out[ ]:
```
<AxesSubplot:>
```

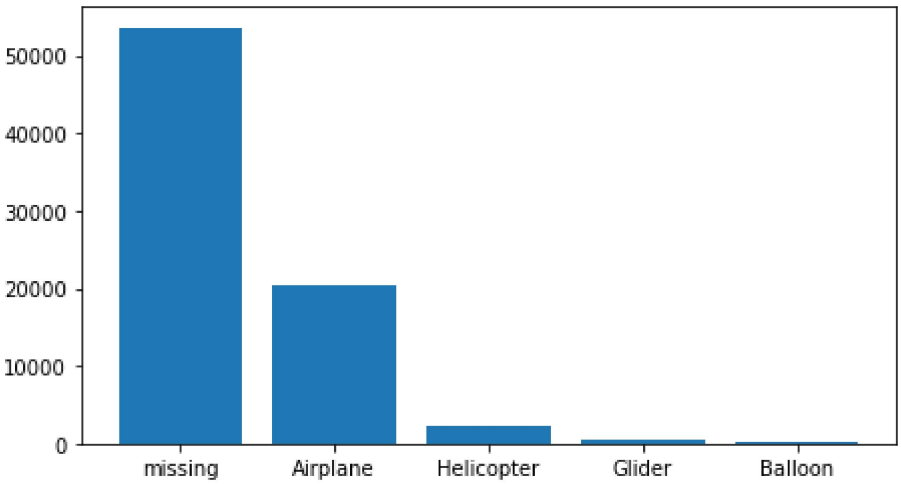


In [168…

```
#showing airplane category and  investigation type with their counts
plt.figure(figsize = (16,4))
plt.subplot(1,2,1)

plt.bar(df['AircraftCategory'].value_counts().head(5).index,df['AircraftCatego
#plt.pie(df['InvestigationType'].value_counts().values)
#plt.pie(df['InvestigationType'].value_counts().index,df['InvestigationType'].
```

Out[168… <BarContainer object of 5 artists>



In [ ]:

```
df
```

Out[ ]:

	EventId	InvestigationType	AccidentNumber	EventDate	Location
0	20200509X11853	Accident	CEN20LA173	05-09-2020	Haskell, OK
1	20200507X60215	Accident	CEN20CA174	05-06-2020	Gonazales, TX
2	20200509X85739	Accident	ERA20CA175	05-03-2020	Shirley, NY
3	20200502X81549	Accident	CEN20LA168	05-02-2020	PALMYRA, IL
4	20200502X73540	Accident	CEN20LA167	05-02-2020	HOUSTON, TX

...	...	...	...	...	...	...
77236	20041105X01764	Accident	CHI79FA064	08-02-1979	Canton, OH
77237	20001218X45448	Accident	LAX96LA321	19-06-1977	EUREKA, CA
77238	20061025X01555	Accident	NYC07LA005	30-08-1974	Saltville, VA
77239	20001218X45447	Accident	LAX94LA336	19-07-1962	BRIDGEPORT, CA
77240	20001218X45444	Accident	SEA87LA080	24-10-1948	MOOSE CREEK, ID

77241 rows × 20 columns

In [140…
```
df['AircraftDamage'].value_counts()
```

Out[140…
```
Substantial    58540
Destroyed      16817
Minor           1884
Name: AircraftDamage, dtype: int64
```

In [170…
```
df.columns
```

Out[170…
```
Index(['EventId', 'InvestigationType', 'AccidentNumber', 'EventDate',
       'Location', 'Country', 'Latitude', 'Longitude', 'InjurySeverity',
       'AircraftDamage', 'AircraftCategory', 'Make', 'Model',
```