In [9]:
```
import pandas as pd
import seaborn as sns
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
from matplotlib import style
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.mixture import GaussianMixture
from sklearn.metrics import silhouette_score
from plotnine import *
sns.set(style = "darkgrid")
pd.options.mode.chained_assignment = None
```

In [10]:
```
dis0 = pd.read_csv("StormEvents_details-ftp_v1.0_d2000_c20210803.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis1 = pd.read_csv("StormEvents_details-ftp_v1.0_d2001_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis2 = pd.read_csv("StormEvents_details-ftp_v1.0_d2002_c20211102.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis3 = pd.read_csv("StormEvents_details-ftp_v1.0_d2003_c20210803.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis4 = pd.read_csv("StormEvents_details-ftp_v1.0_d2004_c20210803.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis5 = pd.read_csv("StormEvents_details-ftp_v1.0_d2005_c20210803.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis6 = pd.read_csv("StormEvents_details-ftp_v1.0_d2006_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis7 = pd.read_csv("StormEvents_details-ftp_v1.0_d2007_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis8 = pd.read_csv("StormEvents_details-ftp_v1.0_d2008_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis9 = pd.read_csv("StormEvents_details-ftp_v1.0_d2009_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis10 = pd.read_csv("StormEvents_details-ftp_v1.0_d2010_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis11 = pd.read_csv("StormEvents_details-ftp_v1.0_d2011_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis12 = pd.read_csv("StormEvents_details-ftp_v1.0_d2012_c20220107.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis13 = pd.read_csv("StormEvents_details-ftp_v1.0_d2013_c20220124.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis14 = pd.read_csv("StormEvents_details-ftp_v1.0_d2014_c20211217.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis15 = pd.read_csv("StormEvents_details-ftp_v1.0_d2015_c20211217.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis16 = pd.read_csv("StormEvents_details-ftp_v1.0_d2016_c20211217.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis17 = pd.read_csv("StormEvents_details-ftp_v1.0_d2017_c20220124.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis18 = pd.read_csv("StormEvents_details-ftp_v1.0_d2018_c20220124.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis19 = pd.read_csv("StormEvents_details-ftp_v1.0_d2019_c20220124.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis20 = pd.read_csv("StormEvents_details-ftp_v1.0_d2020_c20220124.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
dis21 = pd.read_csv("StormEvents_details-ftp_v1.0_d2021_c20220124.csv.gz", compression="gzip",
                                header=0, sep=',', quotechar='"')
```

```
C:\Users\datre\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3172: DtypeWarning: Columns (29,34,35,3
7) have mixed types.Specify dtype option on import or set low_memory=False.
```

In [11]:
```
frames = [dis1, dis2, dis3, dis4, dis5, dis6, dis7, dis8, dis9, dis10, dis11, dis12,
                    dis13, dis14, dis15, dis16, dis17, dis18, dis19, dis20, dis21]
dis_full = pd.concat(frames)
dis_full.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1244206 entries, 0 to 53186
Data columns (total 51 columns):
 #   Column              Non-Null Count    Dtype
---  ------              --------------    -----
 0   BEGIN_YEARMONTH     1244206 non-null  int64
 1   BEGIN_DAY           1244206 non-null  int64
```

```
 2   BEGIN_TIME          1244206 non-null  int64
 3   END_YEARMONTH       1244206 non-null  int64
 4   END_DAY             1244206 non-null  int64
 5   END_TIME            1244206 non-null  int64
 6   EPISODE_ID          1244206 non-null  int64
 7   EVENT_ID            1244206 non-null  int64
 8   STATE               1244205 non-null  object
 9   STATE_FIPS          1244205 non-null  float64
 10  YEAR                1244206 non-null  int64
 11  MONTH_NAME          1244206 non-null  object
 12  EVENT_TYPE          1244206 non-null  object
 13  CZ_TYPE             1244206 non-null  object
 14  CZ_FIPS             1244206 non-null  int64
 15  CZ_NAME             1244206 non-null  object
 16  WFO                 1244206 non-null  object
 17  BEGIN_DATE_TIME     1244206 non-null  object
 18  CZ_TIMEZONE         1244206 non-null  object
 19  END_DATE_TIME       1244206 non-null  object
 20  INJURIES_DIRECT     1244206 non-null  int64
 21  INJURIES_INDIRECT   1244206 non-null  int64
 22  DEATHS_DIRECT       1244206 non-null  int64
 23  DEATHS_INDIRECT     1244206 non-null  int64
 24  DAMAGE_PROPERTY     868745 non-null   object
 25  DAMAGE_CROPS        804161 non-null   object
 26  SOURCE              1244206 non-null  object
 27  MAGNITUDE           666510 non-null   float64
 28  MAGNITUDE_TYPE      418421 non-null   object
 29  FLOOD_CAUSE         98768 non-null    object
 30  CATEGORY            445 non-null      float64
 31  TOR_F_SCALE         29079 non-null    object
 32  TOR_LENGTH          29079 non-null    float64
 33  TOR_WIDTH           29079 non-null    float64
 34  TOR_OTHER_WFO       2577 non-null     object
 35  TOR_OTHER_CZ_STATE  2577 non-null     object
 36  TOR_OTHER_CZ_FIPS   2577 non-null     float64
 37  TOR_OTHER_CZ_NAME   2577 non-null     object
 38  BEGIN_RANGE         670079 non-null   float64
 39  BEGIN_AZIMUTH       670079 non-null   object
 40  BEGIN_LOCATION      793555 non-null   object
 41  END_RANGE           670087 non-null   float64
 42  END_AZIMUTH         670087 non-null   object
 43  END_LOCATION        793569 non-null   object
 44  BEGIN_LAT           759541 non-null   float64
 45  BEGIN_LON           759537 non-null   float64
 46  END_LAT             759541 non-null   float64
 47  END_LON             759537 non-null   float64
 48  EPISODE_NARRATIVE   1088310 non-null  object
 49  EVENT_NARRATIVE     791769 non-null   object
 50  DATA_SOURCE         1244206 non-null  object
dtypes: float64(12), int64(14), object(25)
memory usage: 493.6+ MB
```

In [12]:
```
dis_red = dis_full[["YEAR", "MONTH_NAME", "BEGIN_DAY", "STATE",  "CZ_NAME", "BEGIN_LOCATION","EVENT_TYPE", "CZ_TYPE
dis_red.head()
```

Out[12]:

	YEAR	MONTH_NAME	BEGIN_DAY	STATE	CZ_NAME	BEGIN_LOCATION	EVENT_TYPE	CZ_TYPE	DAMAGE_PROPERTY
0	2001	November	28	TEXAS	BOSQUE	NaN	Ice Storm	Z	NaN
1	2001	October	24	LAKE ERIE	MI WATERS OF LAKE ERIE	STONY POINT	Marine Thunderstorm Wind	M	NaN
2	2001	December	1	NEW JERSEY	SUSSEX	NaN	Heat	Z	0
3	2001	November	29	TEXAS	ELLIS	NaN	Ice Storm	Z	NaN
4	2001	November	15	SOUTH CAROLINA	GEORGETOWN	NaN	Drought	Z	NaN

In [13]:
```
dis_red.info()
dis_red.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1244206 entries, 0 to 53186
Data columns (total 9 columns):
 #   Column           Non-Null Count    Dtype
---  ------           --------------    -----
 0   YEAR             1244206 non-null  int64
 1   MONTH_NAME       1244206 non-null  object
```

```
 2   BEGIN_DAY         1244206 non-null  int64
 3   STATE             1244205 non-null  object
 4   CZ_NAME           1244206 non-null  object
 5   BEGIN_LOCATION    793555 non-null   object
 6   EVENT_TYPE        1244206 non-null  object
 7   CZ_TYPE           1244206 non-null  object
 8   DAMAGE_PROPERTY   868745 non-null   object
dtypes: int64(2), object(7)
memory usage: 94.9+ MB
```

Out[13]:
```
YEAR                   0
MONTH_NAME             0
BEGIN_DAY              0
STATE                  1
CZ_NAME                0
BEGIN_LOCATION    450651
EVENT_TYPE             0
CZ_TYPE                0
DAMAGE_PROPERTY   375461
dtype: int64
```

In [14]:
```
dis_red = dis_red.dropna()
```

In [15]:
```
dis_red["DAMAGE_PROPERTY"].unique()
```

Out[15]:
```
array(['0', '50K', '2K', ..., '139.00M', '885.00K', '3.35M'], dtype=object)
```

In [16]:
```
def value_to_float(x):
    if type(x) == float or type(x) == int:
        return x
    if 'K' in x:
        if len(x) > 1:
            return float(x.replace('K', '')) * 1000
        return 1000.0
    if 'M' in x:
        if len(x) > 1:
            return float(x.replace('M', '')) * 1000000
        return 1000000.0
    if 'B' in x:
        return float(x.replace('B', '')) * 1000000000
    return 0.0

dis_red["DAMAGE_PROPERTY"] = dis_red["DAMAGE_PROPERTY"].apply(value_to_float)
```

In [17]:
```
dis_red["DAMAGE_PROPERTY"].unique()
```

Out[17]:
```
array([0.00e+00, 5.00e+04, 2.00e+03, ..., 1.39e+08, 8.85e+05, 3.35e+06])
```

In [18]:
```
cst = ["MAINE", "CONNECTICUT", "DELAWARE", "GEORGIA",
       "FLORIDA", "VIRGINA", "MARYLAND", "MASSACHUSETTS",
      "NEW HAMPSHIRE", "NEW JERSEY", "NEW YORK", "RHODE ISLAND",
      "NORTH CAROLINA", "SOUTH CAROLINA", "ALASKA", "HAWAII",
      "CALIFORNIA", "OREGON", "WASHINGTON", "ALABAMA", "LOUSISIANA",
      "TEXAS","MISSISSIPPI"]
```

In [19]:
```
ccty = ["BALDWIN", "MOBILE", "ANCHORAGE", "BETHEL", "BRISTOL BAY",
       "DILLINGHAM", "HAINES", "HOONAH_ANGOON", "JUNEAU", "KENAI PENINSULA",
       "KETCHIKAN GATEWAY", "KODIAK ISLAND", "KUSILVAK", "NOME",
       "NORTH SLOPE", "PETERSBURG", "SITKA", "SKAGWAY", "VALDEZ-CORDOVA",
       "WRANGELL CITY", "YAKUTAT", "ALAMEDA", "CONTRA COSTA", "DEL NORTE",
       "HUMBOLDT", "LOS ANGELES", "MARIN", "MENDOCINO", "ORANGE", "SAN DIEGO",
       "SAN FRANCISO", "SAN LUIS OBISPO", "SAN MATEO", "SANTA BARBARA",
       "SANTA CLARA", "SANTA CRUZ", "SOALNO", "SONOMA", "VENTURA", "FAIRFIELD",
       "MIDDLESEX", "NEW HAVEN", "NEW LONDON", "KENT", "NEW CASTLE", "SUSSEX",
       "BAY", "BREVARD", "BROWARD", "CHARLOTTE", "CITRUS", "COLLIER", "DIXIE",
       "DUVAL", "ESCAMBIA", "FLAGLER", "FRANKLIN", "GULF", "HERNANDO",
       "HILLSBOROUGH", "INDIAN RIVER", "JEFFERSON", "LEE", "LEVY", "MANATEE",
       "MARTIN", "MIAMI-DADE", "MONROE", "NASSAU", "OKALOOSA", "PALM BEACH",
       "PASCO", "ST. JOHNS", "PINELLAS", "ST. LUCIE", "SARASOTA",
       "SANTA ROSA", "TAYLOR", "VOLUSIA", "WAKULLA", "WALTON", "BRYAN", "CAMDEN",
       "CHATHAM", "GLYNN", "LIBERTY", "MCINTOSH", "HAWAII", "HONOLULU", "KALAWAO",
       "KAUAI", "MAUI", "CAMERON", "IBERIA", "LAFOURCHE", "ORLEANS", "PLAQUEMINES",
       "ST. BERNARD", "ST. MARY", "ST. TAMMANY", "TERREBONNE", "CUMBERLAND", "HANCOCK",
       "KNOX", "LINCOLN", "SAGADAHOC", "WALDO", "WASHINGTON", "YORK", "ANNE ARUNDEL",
       "BALTIMORE", "CALVERT", "CECIL", "CHARLES", "DORCHESTER", "HARFORD", "KENT",
       "QUEEN ANNE'S", "ST. MARY'S", "SOMERSET", "TALBOT", "WICOMICO", "WORCESTER",
```

```
"BALTIMORE CITY", "BRISTOL", "DUKES", "BARNSTABLE", "ESSEX", "NANTUCKET",
"NORFOLK", "PLYMOUTH", "SUFFOLK", "HANCOCK", "HARRISON", "JACKSON", "ROCKINGHAM",
"ATLANTIC", "CAPE MAY", "HUDSON", "OCEAN", "SALEM", "UNION", "BRONX", "KINGS", "NEW YORK",
"QUEENS", "RICHMOND", "SUFFOLK", "WESTCHESTER", "BEAUFORT", "BERTIE", "BRUNSWICK", "CARTERET",
"CHOWAN", "CRAVEN", "CURRITUCK", "DARE", "HYDE", "NEW HANOVER", "ONSLOW", "PAMLICO",
"PASQUOTANK", "PENDER", "PERQUIMANS", "TYRRELL", "CLATSOP", "COOS", "CURRY",
"DOUGLAS", "LANE", "TILLAMOOK", "NEWPORT", "PRODIVENCE", "BEAUFORT", "CHARLESTON", "COLLETON",
"GEORGETOWN", "HORRY", "JASPER", "ARANSAS", "BRAZORIA", "CALHOUN", "CHAMBERS",
"GALESTON", "HARRIS", "KENEDY", "KLEBERG", "MATAGORDA", "NEUCES", "REFUGIO", "SAN PATRICIO",
"VICTORIA", "WILLACY", "ACCOMACK", "FAIRFAX", "ISLE OF WIGHT", "JAMES CITY", "KING GEORGE",
"LANCASTER", "MATHEWS", "NORTHAMPTON", "NORTHUMBERLAND", "PRINCE WILLIAM", "STAFFORD", "SURRY",
"WESTMORELAND", "HAMPTON CITY", "NEWPORT NEWS", "NORFOLK CITY", "POQUOSON CITY",
"PORTSMOUTH CITY", "VIRGINIA BEACH", "CLALLAM", "GRAYS HARBOR", "ISALND", "KITSAP",
"MASON", "PACIFIC", "PIERCE", "SAN JUAN", "SKAGIT", "SNOHOMISH", "WAHKIAKUM", "WHATCOM"]
```

In [20]:
```
en = ["Astronomical Low Tide", "Coastal Flood", "Debris Flow", "Flash Flood", "Flood", "Lakeshore Flood", "Marine H
      "Marine High Wind", "Marine Strong Wind", "Marine Thunderstorm Wind", "Rip Current", "Seiche", "Storm Surge/T
     "Tropical Storm", "Waterspout", "Hurricane (Typhoon)", "Hurricane", "Typhoon", "Tropical Cyclone", "Tropical S
```

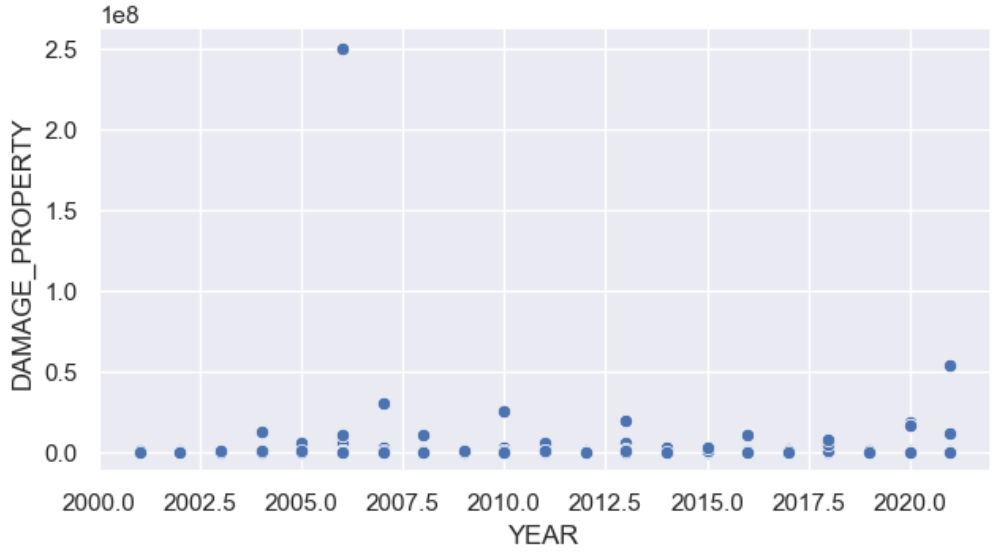
In [21]:
```
dis_red = dis_red[dis_red["STATE"].isin(cst)]
dis_red = dis_red[dis_red["CZ_NAME"].isin(cst)]
dis_red = dis_red[dis_red["EVENT_TYPE"].isin(en)]
dis_red.drop(dis_red.index[dis_red["DAMAGE_PROPERTY"] == 0], inplace=True)
dis_red
```

Out[21]:

	YEAR	MONTH_NAME	BEGIN_DAY	STATE	CZ_NAME	BEGIN_LOCATION	EVENT_TYPE	CZ_TYPE	DAMAGE_PROPERTY
11074	2001	March	3	ALABAMA	WASHINGTON	COUNTYWIDE	Flash Flood	C	30000.0
12528	2001	April	3	ALABAMA	WASHINGTON	MILLRY	Flash Flood	C	3000.0
42466	2001	October	24	HAWAII	HAWAII	HILO	Flash Flood	C	30000.0
42664	2001	November	28	MISSISSIPPI	WASHINGTON	COUNTYWIDE	Flash Flood	C	100000.0
44385	2001	December	12	MISSISSIPPI	WASHINGTON	COUNTYWIDE	Flash Flood	C	1000.0
...	...	...	...	...	...	...	...	...	...
41736	2021	July	12	NEW YORK	DELAWARE	HANCOCK	Flash Flood	C	20000.0
46569	2021	August	16	FLORIDA	WASHINGTON	BAHOMA	Flood	C	11500000.0
50578	2021	September	1	NEW YORK	NEW YORK	CENTRAL PARK	Flash Flood	C	54000000.0
50638	2021	October	26	NEW YORK	DELAWARE	HANCOCK	Flash Flood	C	10000.0
50639	2021	October	26	NEW YORK	DELAWARE	MARGARETVILLE	Flash Flood	C	10000.0

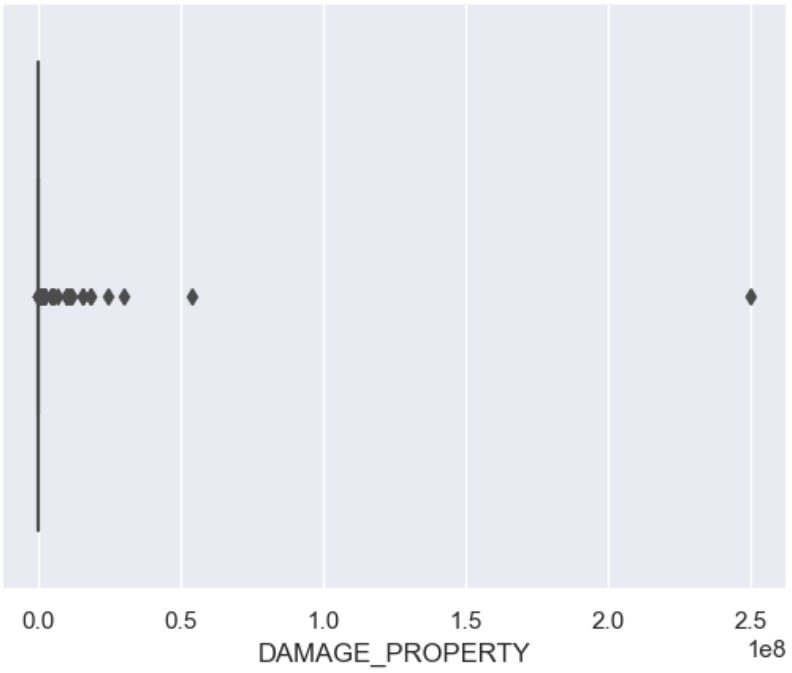
213 rows × 9 columns

In [22]:
```
sns.scatterplot(x = "YEAR", y = "DAMAGE_PROPERTY", data = dis_red)
plt.show()
```



In [23]:
```
sns.boxplot(dis_red["DAMAGE_PROPERTY"])
plt.show()
```

```
C:\Users\datre\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as
a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments
without an explicit keyword will result in an error or misinterpretation.
```



In [24]:
```
percentile25 = dis_red["DAMAGE_PROPERTY"].quantile(0.25)
percentile75 = dis_red["DAMAGE_PROPERTY"].quantile(0.75)
iqr = percentile75-percentile25
```

In [25]:
```
upper_limit = percentile75 + 1.5 * iqr
lower_limit = percentile25 - 1.5 * iqr
```

In [26]:
```
dis_red[dis_red["DAMAGE_PROPERTY"] > upper_limit]
dis_red[dis_red["DAMAGE_PROPERTY"] < lower_limit]
```

Out[26]:

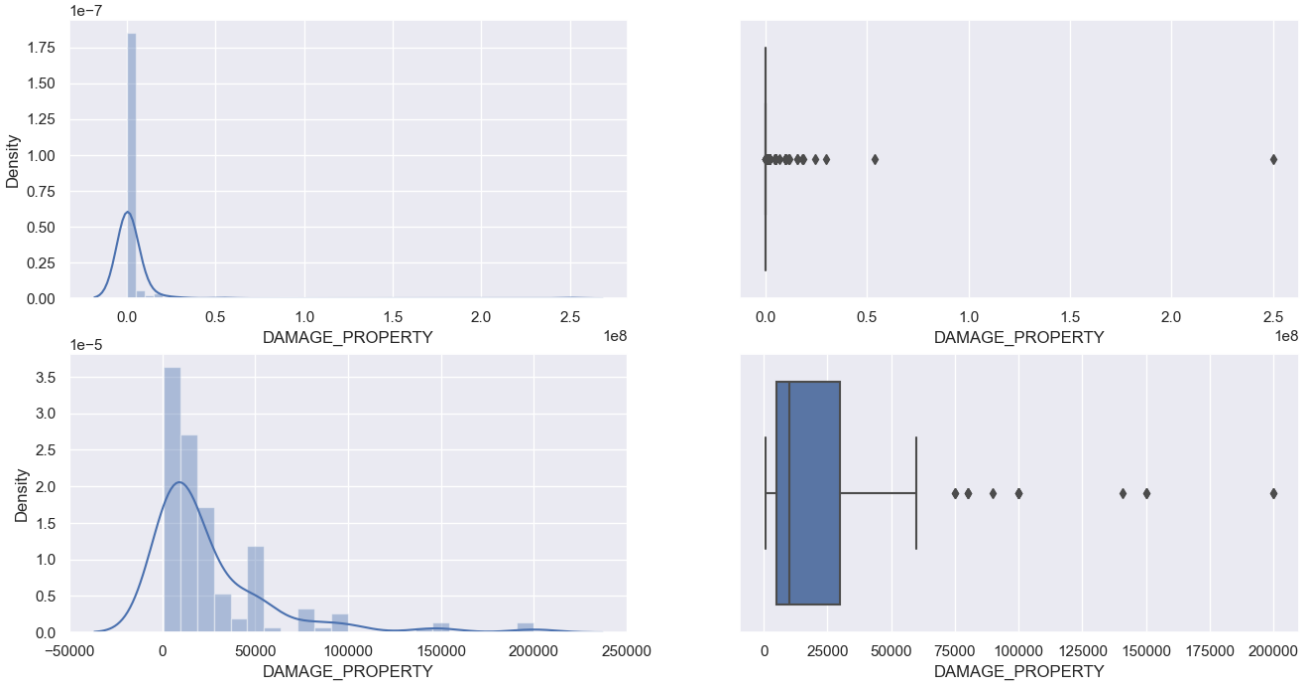
YEAR	MONTH_NAME	BEGIN_DAY	STATE	CZ_NAME	BEGIN_LOCATION	EVENT_TYPE	CZ_TYPE	DAMAGE_PROPERTY

In [27]:
```
dis_new = dis_red[dis_red["DAMAGE_PROPERTY"] < upper_limit]
dis_new.shape
```

Out[27]: (167, 9)

In [28]:
```
plt.figure(figsize=(16,8))
plt.subplot(2,2,1)
sns.distplot(dis_red["DAMAGE_PROPERTY"])
plt.subplot(2,2,2)
sns.boxplot(x = dis_red["DAMAGE_PROPERTY"])
plt.subplot(2,2,3)
sns.distplot(dis_new["DAMAGE_PROPERTY"])
plt.subplot(2,2,4)
sns.boxplot(x = dis_new["DAMAGE_PROPERTY"])
plt.show()
```

```
C:\Users\datre\anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for histograms).
C:\Users\datre\anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for histograms).
```



In [29]:
```
percentile25 = dis_new["DAMAGE_PROPERTY"].quantile(0.25)
percentile75 = dis_new["DAMAGE_PROPERTY"].quantile(0.75)
iqr = percentile75-percentile25
```

In [30]:
```
upper_limit = percentile75 + 1.5 * iqr
lower_limit = percentile25 - 1.5 * iqr
```

In [31]:
```
dis_new[dis_new["DAMAGE_PROPERTY"] > upper_limit]
dis_new[dis_new["DAMAGE_PROPERTY"] < lower_limit]
```

Out[31]:

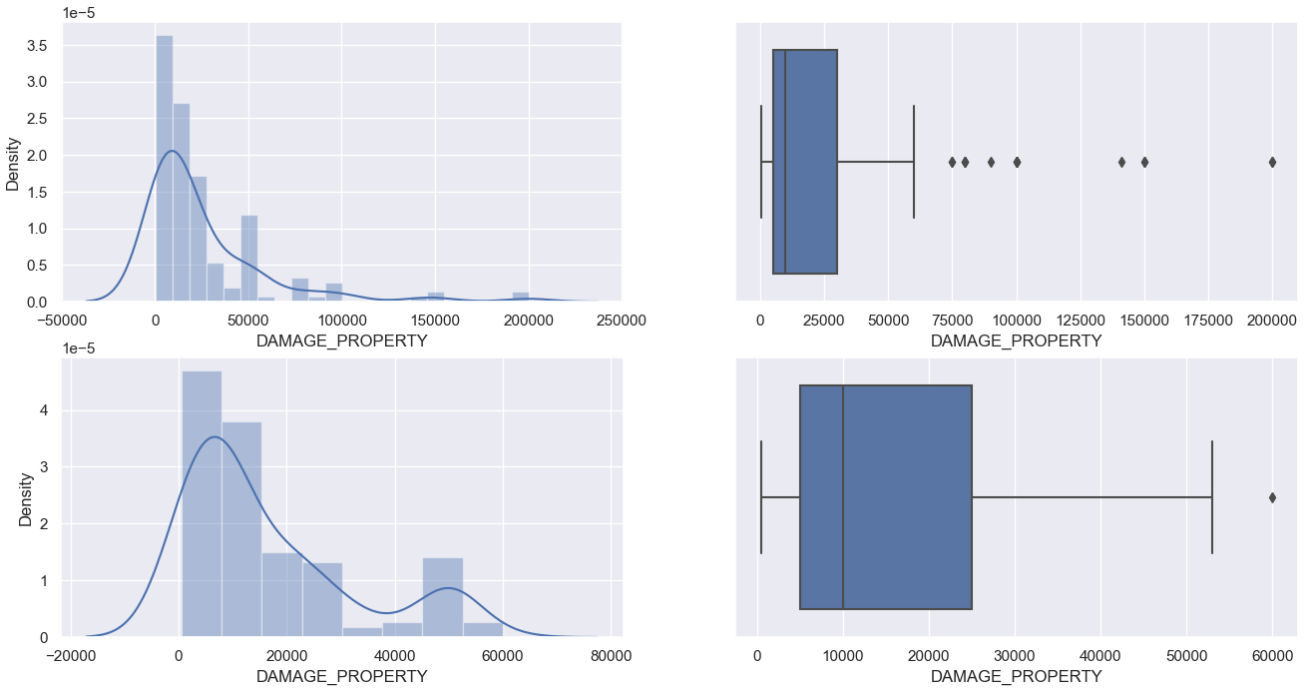
YEAR	MONTH_NAME	BEGIN_DAY	STATE	CZ_NAME	BEGIN_LOCATION	EVENT_TYPE	CZ_TYPE	DAMAGE_PROPERTY

In [32]:
```
dis_new2 = dis_new[dis_new["DAMAGE_PROPERTY"] < upper_limit]
dis_new2.shape
```

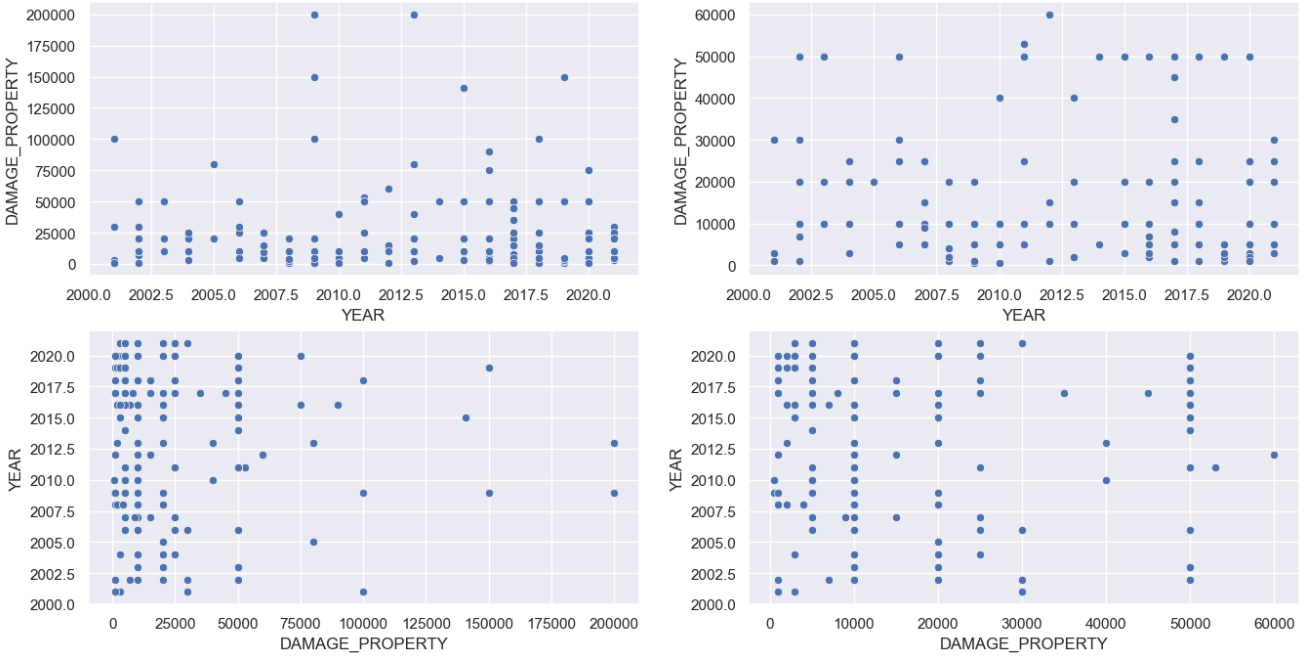
Out[32]: (152, 9)

In [33]:
```
plt.figure(figsize=(16,8))
plt.subplot(2,2,1)
sns.distplot(dis_new["DAMAGE_PROPERTY"])
plt.subplot(2,2,2)
sns.boxplot(x = dis_new["DAMAGE_PROPERTY"])
plt.subplot(2,2,3)
sns.distplot(dis_new2["DAMAGE_PROPERTY"])
plt.subplot(2,2,4)
sns.boxplot(x = dis_new2["DAMAGE_PROPERTY"])
plt.show()
```

```
C:\Users\datre\anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for histograms).
C:\Users\datre\anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for histograms).
```



In [34]:
```
plt.figure(figsize=(16,8))
plt.subplot(2,2,1)
sns.scatterplot(x = dis_new["YEAR"], y = dis_new["DAMAGE_PROPERTY"])
plt.subplot(2,2,2)
sns.scatterplot(x = dis_new2["YEAR"], y = dis_new2["DAMAGE_PROPERTY"])
plt.subplot(2,2,3)
sns.scatterplot(x = dis_new["DAMAGE_PROPERTY"], y = dis_new["YEAR"])
plt.subplot(2,2,4)
sns.scatterplot(x = dis_new2["DAMAGE_PROPERTY"], y = dis_new2["YEAR"])
plt.show()
```



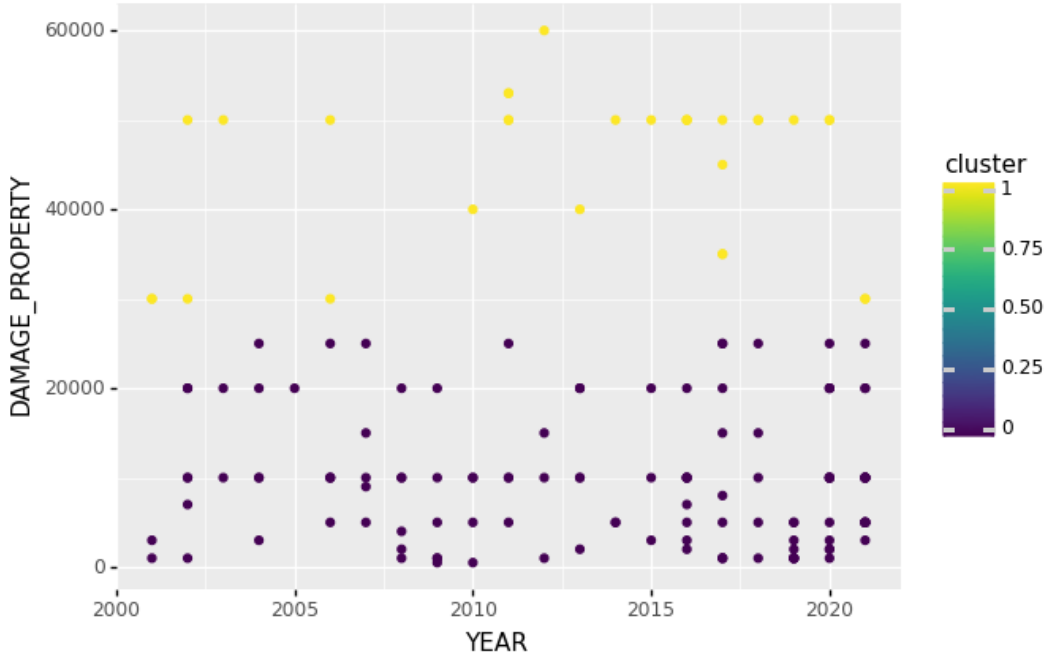
In [36]:
```
features = ["YEAR", "DAMAGE_PROPERTY"]
X2 = dis_new2[features]
```

In [37]:
```
model = GaussianMixture(n_components = 2)
model.fit(X2)
model_y2 = model.predict(X2)
print("Silhouette:", silhouette_score(X2, model_y2))
```

```
Silhouette: 0.7266865754169947
```

In [38]:
```
X2["cluster"] = model_y2
```

In [39]:
```
(ggplot(X2, aes(x = "YEAR", y = "DAMAGE_PROPERTY", color = "cluster"))
     + geom_point())
```



Out[39]: <ggplot: (109851218574)>

In [40]:
```
dis_new2["cluster"] = X2["cluster"]
```

In [ ]:
```

```

In [42]:
```
risk = dis_new2.loc[dis_new2["cluster"] == 1]
risk
```

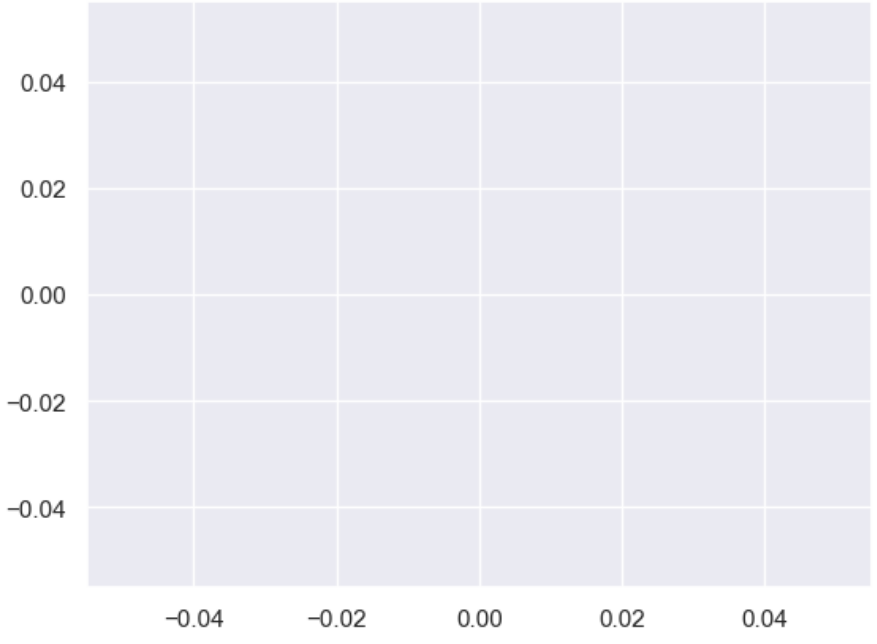
Out[42]:

	YEAR	MONTH_NAME	BEGIN_DAY	STATE	CZ_NAME	BEGIN_LOCATION	EVENT_TYPE	CZ_TYPE	DAMAGE_PROPERTY	c
11074	2001	March	3	ALABAMA	WASHINGTON	COUNTYWIDE	Flash Flood	C	30000.0	
42466	2001	October	24	HAWAII	HAWAII	HILO	Flash Flood	C	30000.0	
30788	2002	May	27	NEW YORK	DELAWARE	HANCOCK	Flash Flood	C	50000.0	
43033	2002	August	15	TEXAS	WASHINGTON	COUNTYWIDE	Flash Flood	C	30000.0	
42576	2003	June	13	NEW YORK	DELAWARE	DEPOSIT	Flash Flood	C	50000.0	
26885	2006	June	26	NEW YORK	DELAWARE	COLCHESTER	Flash Flood	C	50000.0	
51971	2006	November	16	NEW YORK	DELAWARE	HANCOCK	Flash Flood	C	30000.0	
35001	2010	March	14	RHODE ISLAND	WASHINGTON	WESTERLY STATE ARPT	Flood	C	40000.0	
28728	2011	April	28	NEW YORK	WASHINGTON	SOUTH HUDSON FALLS	Flood	C	53000.0	
32029	2011	April	28	NEW YORK	WASHINGTON	WHITEHALL	Flood	C	53000.0	
70230	2011	August	28	NEW YORK	DELAWARE	HARVARD	Flood	C	50000.0	
70231	2011	August	28	NEW YORK	DELAWARE	FISHS EDDY	Flood	C	50000.0	
25314	2012	June	11	MISSISSIPPI	WASHINGTON	HOLLANDALE	Flash Flood	C	60000.0	
39231	2013	May	29	NEW YORK	DELAWARE	EAST DELHI	Flash Flood	C	40000.0	
44580	2014	June	25	NEW YORK	DELAWARE	HALE EDDY	Flash Flood	C	50000.0	
50061	2015	November	9	FLORIDA	WASHINGTON	CROW	Flood	C	50000.0	
12388	2016	March	12	MISSISSIPPI	WASHINGTON	GREENVILLE	Flood	C	50000.0	

	YEAR	MONTH_NAME	BEGIN_DAY	STATE	CZ_NAME	BEGIN_LOCATION	EVENT_TYPE	CZ_TYPE	DAMAGE_PROPERTY	c
24159	2016	June	28	NEW YORK	DELAWARE	TROUT CREEK	Flash Flood	C	50000.0	
24160	2016	June	28	NEW YORK	DELAWARE	BARBOURVIILE	Flash Flood	C	50000.0	
31620	2017	June	5	NEW YORK	DELAWARE	MEREDITH	Flash Flood	C	35000.0	
37451	2017	August	8	MISSISSIPPI	WASHINGTON	HOLLANDALE	Flash Flood	C	50000.0	
41213	2017	February	25	NEW YORK	DELAWARE	HAWLEYS	Flash Flood	C	45000.0	
43685	2017	February	25	NEW YORK	DELAWARE	MERRICKVILLE	Flash Flood	C	35000.0	
41986	2018	May	15	MARYLAND	WASHINGTON	DARGAN	Flash Flood	C	50000.0	
45945	2018	May	15	MARYLAND	WASHINGTON	DARGAN	Flash Flood	C	50000.0	
13045	2019	February	23	MISSISSIPPI	WASHINGTON	GREENVILLE	Flash Flood	C	50000.0	
45860	2020	September	17	GEORGIA	WASHINGTON	HEIDRICH	Flash Flood	C	50000.0	
47220	2020	December	25	NEW YORK	DELAWARE	DAVENPORT	Flash Flood	C	50000.0	
254	2021	September	20	GEORGIA	WASHINGTON	HEBRON	Flash Flood	C	30000.0	
13814	2021	June	8	NEW YORK	DELAWARE	ROXBURY	Flash Flood	C	30000.0	

In [59]:
```
fig, ax = plt.subplots()
color=['teal','wheat']
for k in range(0,2):
    X2 = frame[frame["cluster"]==k]
    plt.scatter(X2["YEAR"],X2["DAMAGE_PROPERTY"],c=color[k])
ax.set_xticklabels(X2["YEAR"].astype(int))
plt.xlabel("Year")
plt.ylabel("Property Damage ($)")
plt.gca().legend(("Low Risk","High Risk"))
plt.show()
```

```
<ipython-input-59-b81271e88369>:6: UserWarning: FixedFormatter should only be used together with FixedLocator
```




Low Risk
High Risk
Property Damage ($)
60000
50000
40000
30000
20000
10000
0
2001
2001
2002
2002
2003
2006
2006
2010
2011
Year

In [ ]:

In [ ]: