

```
In [2]: import numpy as np
import pandas as pd
import seaborn as sns
from matplotlib import pyplot as plt
from datetime import date
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.neighbors import LocalOutlierFactor
from sklearn.preprocessing import MinMaxScaler, LabelEncoder, StandardScaler, Robust
```

```
In [5]: def load_application_train():
data = pd.read_csv(r"C:\application_train.csv")
return data

#See the shape of bigger dataset
df = load_application_train()
print(df.shape)
```

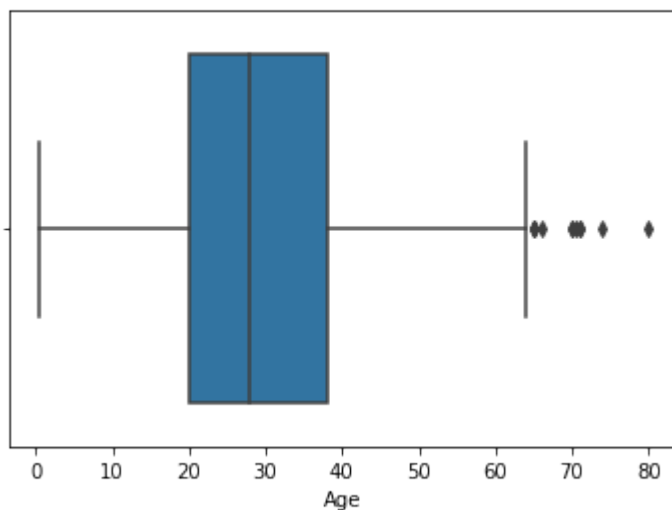
(307511, 122)

```
In [6]: def load_titanic():
data = pd.read_csv(r'Downloads\titanic.csv')
return data

df = load_titanic()
print(df.shape)
```

(891, 12)

```
In [7]: sns.boxplot(x=df["Age"])
plt.show()
```



```
In [12]: q1 = df['Age'].quantile(0.25)
q3 = df['Age'].quantile(0.75)
iqr = q3 - q1
up = q3 + 1.5 * iqr
down = q1 - 1.5 * iqr
print(df[(df['Age'] < down) | (df['Age'] > up)])
```

PassengerId	Survived	Pclass	Name \
33			
34	0	2	Wheadon, Mr. Edward H

54	55	0	1	Ostby, Mr. Engelhart Cornelius
96	97	0	1	Goldschmidt, Mr. George B
116	117	0	3	Connors, Mr. Patrick
280	281	0	3	Duane, Mr. Frank
456	457	0	1	Millet, Mr. Francis Davis
493	494	0	1	Artagaveytia, Mr. Ramon
630	631	1	1	Barkworth, Mr. Algernon Henry Wilson
672	673	0	2	Mitchell, Mr. Henry Michael
745	746	0	1	Crosby, Capt. Edward Gifford
851	852	0	3	Svensson, Mr. Johan

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
33	male	66.0	0	0	C.A. 24579	10.5000	NaN	S
54	male	65.0	0	1	113509	61.9792	B30	C
96	male	71.0	0	0	PC 17754	34.6542	A5	C
116	male	70.5	0	0	370369	7.7500	NaN	Q
280	male	65.0	0	0	336439	7.7500	NaN	Q
456	male	65.0	0	0	13509	26.5500	E38	S
493	male	71.0	0	0	PC 17609	49.5042	NaN	C
630	male	80.0	0	0	27042	30.0000	A23	S
672	male	70.0	0	0	C.A. 24580	10.5000	NaN	S
745	male	70.0	1	1	WE/P 5735	71.0000	B22	S
851	male	74.0	0	0	347060	7.7750	NaN	S

```
In [13]: print(df[(df['Age'] < down) | (df['Age'] > up)].index)
```

```
Int64Index([33, 54, 96, 116, 280, 456, 493, 630, 672, 745, 851], dtype='int64')
```

```
In [14]: print(df[(df['Age'] < down) | (df['Age'] > up)].any(axis = True))
```

```
33      True
54      True
96      True
116     True
280     True
456     True
493     True
630     True
672     True
745     True
851     True
dtype: bool
```

```
In [17]: print(df[(df['Age'] < down)].any(axis=False))
```

```
PassengerId    False
Survived        False
Pclass          False
Name            False
Sex             False
Age             False
SibSp           False
Parch           False
Ticket          False
Fare            False
Cabin           False
Embarked        False
dtype: bool
```

```
In [20]: def outlier_thresholds(dataframe, col_name, q1=0.25, q3=0.75):
          quartile1 = dataframe[col_name].quantile(q1)
          quartile3 = dataframe[col_name].quantile(q3)
```

```

interquartile_range = quartile3 - quartile1
up_limit = quartile3 + 1.5 * interquartile_range
low_limit = quartile1 - 1.5 * interquartile_range
return low_limit, up_limit
print(outlier_thresholds(df, "Age"))

low, up = outlier_thresholds(df, "Fare")
print(df[(df["Fare"] < low) | (df["Fare"] > up)].head())

def check_outlier(dataframe, col_name):
    low_limit, up_limit = outlier_thresholds(dataframe, col_name)
    if dataframe[col_name] > up_limit | (dataframe[col_name] < low_limit):
        return True
    else:
        return False

print(check_outlier(df, "Age"))
print(check_outlier(df, "Fare"))

def grab_col_names(dataframe, cat_th=10, car_th=20):
    cat_cols = [col for col in dataframe.columns if dataframe[col].dtypes == "O"]
    num_but_cat = [col for col in dataframe.columns if dataframe[col].nunique() < cat_th]
    cat_but_car = [col for col in dataframe.columns if dataframe[col].nunique() > car_th]
    cat_cols = cat_cols + num_but_cat
    cat_cols = [col for col in cat_cols if col not in cat_but_car]
    num_cols = [col for col in dataframe.columns if dataframe[col].dtypes != "O" and col not in cat_cols]

    print(f"Observations: {dataframe.shape[0]}")
    print(f"Variables: {dataframe.shape[1]}")
    print(f"cat_cols: {len(cat_cols)}")
    print(f"num_cols: {len(num_cols)}")
    print(f"cat_but_car: {len(cat_but_car)}")
    print(f"num_but_cat: {len(num_but_cat)}")

    return cat_cols, num_cols, cat_but_car

cat_cols, num_cols, cat_but_car = grab_col_names(df)
num_cols = [col for col in num_cols if col not in "PassengerId"]
print(num_cols)

for col in num_cols:
    print(col, check_outlier(df, col))

```

```
(-6.6875, 64.8125)
```

	PassengerId	Survived	Pclass	\
1	2	1	1	
27	28	0	1	
31	32	1	1	
34	35	0	1	
52	53	1	1	

	Name	Sex	Age	SibSp	\
1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	
27	Fortune, Mr. Charles Alexander	male	19.0	3	
31	Spencer, Mrs. William Augustus (Marie Eugenie)	female	NaN	1	
34	Meyer, Mr. Edgar Joseph	male	28.0	1	
52	Harper, Mrs. Henry Sleeper (Myna Haxtun)	female	49.0	1	

	Parch	Ticket	Fare	Cabin	Embarked
1	0	PC 17599	71.2833	C85	C
27	2	19950	263.0000	C23 C25 C27	S

```

31      0  PC 17569  146.5208      B78      C
34      0  PC 17604   82.1708      NaN      C
52      0  PC 17572   76.7292      D33      C
True
True
Observations: 891
Variables: 12
cat_cols: 6
num_cols: 3
cat_but_car: 3
num_but_cat: 4
['Age', 'Fare']
Age True
Fare True

```

In [21]:

```

dff = load_application_train()

cat_cols, num_cols, cat_but_car = grab_col_names(dff)
num_cols.remove('SK_ID_CURR')

print()
print()

for col in num_cols:
    print(col, check_outlier(dff, col))

def grab_outliers(dataframe, col_name, outlier_index=False, f = 5):
    low, up = outlier_thresholds(dataframe, col_name)

    if dataframe[((dataframe[col_name] < low) | (dataframe[col_name] > up))].shape[0]:
        print(dataframe[((dataframe[col_name] < low) | (dataframe[col_name] > up))].
    else:
        print(dataframe[((dataframe[col_name] < low) | (dataframe[col_name] > up))])

    if outlier_index:
        out_index = dataframe[((dataframe[col_name] < low) | (dataframe[col_name] >
        return out_index

age_index = grab_outliers(df, "Age", True)
print(age_index)

```

```

Observations: 307511
Variables: 122
cat_cols: 54
num_cols: 67
cat_but_car: 1
num_but_cat: 39

```

```

CNT_CHILDREN True
AMT_INCOME_TOTAL True
AMT_CREDIT True
AMT_ANNUITY True
AMT_GOODS_PRICE True
REGION_POPULATION_RELATIVE True
DAYS_BIRTH False
DAYS_EMPLOYED True
DAYS_REGISTRATION True
DAYS_ID_PUBLISH False
OWN_CAR_AGE True
CNT_FAM_MEMBERS True
HOUR_APPR_PROCESS_START True
EXT_SOURCE_1 False

```

EXT_SOURCE_2 False
EXT_SOURCE_3 False
APARTMENTS_AVG True
BASEMENTAREA_AVG True
YEARS_BEGINEXPLUATATION_AVG True
YEARS_BUILD_AVG True
COMMONAREA_AVG True
ELEVATORS_AVG True
ENTRANCES_AVG True
FLOORSMAX_AVG True
FLOORSMIN_AVG True
LANDAREA_AVG True
LIVINGAPARTMENTS_AVG True
LIVINGAREA_AVG True
NONLIVINGAPARTMENTS_AVG True
NONLIVINGAREA_AVG True
APARTMENTS_MODE True
BASEMENTAREA_MODE True
YEARS_BEGINEXPLUATATION_MODE True
YEARS_BUILD_MODE True
COMMONAREA_MODE True
ELEVATORS_MODE True
ENTRANCES_MODE True
FLOORSMAX_MODE True
FLOORSMIN_MODE True
LANDAREA_MODE True
LIVINGAPARTMENTS_MODE True
LIVINGAREA_MODE True
NONLIVINGAPARTMENTS_MODE True
NONLIVINGAREA_MODE True
APARTMENTS_MEDI True
BASEMENTAREA_MEDI True
YEARS_BEGINEXPLUATATION_MEDI True
YEARS_BUILD_MEDI True
COMMONAREA_MEDI True
ELEVATORS_MEDI True
ENTRANCES_MEDI True
FLOORSMAX_MEDI True
FLOORSMIN_MEDI True
LANDAREA_MEDI True
LIVINGAPARTMENTS_MEDI True
LIVINGAREA_MEDI True
NONLIVINGAPARTMENTS_MEDI True
NONLIVINGAREA_MEDI True
TOTALAREA_MODE True
OBS_30_CNT_SOCIAL_CIRCLE True
DEF_30_CNT_SOCIAL_CIRCLE True
OBS_60_CNT_SOCIAL_CIRCLE True
DAYS_LAST_PHONE_CHANGE True
AMT_REQ_CREDIT_BUREAU_MON True
AMT_REQ_CREDIT_BUREAU_QRT True
AMT_REQ_CREDIT_BUREAU_YEAR True

PassengerId	Survived	Pclass	Name	Sex	\
33	34	0	2	Wheadon, Mr. Edward H	male
54	55	0	1	Ostby, Mr. Engelhart Cornelius	male
96	97	0	1	Goldschmidt, Mr. George B	male
116	117	0	3	Connors, Mr. Patrick	male
280	281	0	3	Duane, Mr. Frank	male

	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
33	66.0	0	0	C.A. 24579	10.5000	NaN	S
54	65.0	0	1	113509	61.9792	B30	C
96	71.0	0	0	PC 17754	34.6542	A5	C
116	70.5	0	0	370369	7.7500	NaN	Q

```
280 65.0      0      0      336439  7.7500  NaN      Q
Int64Index([33, 54, 96, 116, 280, 456, 493, 630, 672, 745, 851], dtype='int64')
```

In [27]:

```
df = load_titanic()
down, up=outlier_thresholds(df, 'Fare')
print(df.shape)

print(df[~((df["Fare"] < low) | (df["Fare"] > up))].shape)

def remove_outlier(dataframe, col_name):
    low_limit, up_limit = outlier_thresholds(dataframe, col_name)
    df_without_outliers = dataframe[~((dataframe[col_name] < low_limit) | (dataframe
    return df_without_outliers

cat_cols, num_cols, cat_but_car = grab_col_names(df)
num_cols.remove('PassengerId')

for col in num_cols:
    df = remove_outlier(df,col)
print(df.shape)
```

```
(891, 12)
(775, 12)
Observations: 891
Variables: 12
cat_cols: 6
num_cols: 3
cat_but_car: 3
num_but_cat: 4
(765, 12)
```

In []: