

OUTLIER DETECTION AND REMOVING

1) Z-Score (Apply only for uniform data/gaussian data)

```
upper_limit=df['column'].mean()+3*df['column'].std()
```

```
lower_limit=df['column'].mean()-3*df['column'].std()
```

```
# or
```

```
z_score=df['column']-df['column'].mean()/df['column'].std()
```

or

Trimming for Z-Score z_score between -1 and +1 otherwise all are outliers

```
new_df = df[(df['z_score'] < 3) & (df['z_score'] > -3)]
```

2) IQR (apply non-uniform data)

```
percentile25 = df['column'].quantile(0.25)
```

```
percentile75 = df['column'].quantile(0.75)
```

```
iqr = percentile75 - percentile25
```

```
lower_limit = percentile25 - 1.5 * iqr
```

```
upper_limit = percentile75 + 1.5 * iqr
```

3) Percentile Method

```
lower_limit = df['column'].quantile(0.01)
```

```
upper_limit = df['column'].quantile(0.99)
```

#Clapping / Winsorization (clap or compreeses to range limit i.e on boundry)

```
df['column'] = np.where( df['column']>upper_limit, upper_limit,  
                        np.where( df['column']<lower_limit, lower_limit,  
                        df['column']))
```

#Trimming (remvng rows having outliers from data)

```
new_df = df[(df['column'] < upperlimit) & (df['column'] > lower_limit)]  
new_df
```