standardizing the data

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

In [2]:
```python
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```python
from sklearn import preprocessing
from sklearn import model_selection
from sklearn import metrics
from sklearn import linear_model
from sklearn import ensemble
from sklearn import tree
from sklearn import svm
import xgboost
```

In [4]:
```python
from sklearn.preprocessing import StandardScaler
```

In [5]:
```python
data = pd.read_csv("E:\IBM_Project\weatherAUS.csv")
```

In [6]:
```python
#splitting x and y values
y=data['RainTomorrow']
x=data.drop('RainTomorrow',axis=1)
```

In [7]:
```python
names=x.columns #Loading the names of the x_features
```

In [8]:
```python
names
```

Out[8]:
```
Index(['Date', 'Location', 'MinTemp', 'MaxTemp', 'Rainfall', 'Evaporation',
       'Sunshine', 'WindGustDir', 'WindGustSpeed', 'WindDir9am', 'WindDir3pm',
       'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am', 'Humidity3pm',
       'Pressure9am', 'Pressure3pm', 'Cloud9am', 'Cloud3pm', 'Temp9am',
       'Temp3pm', 'RainToday', 'RISK_MM'],
      dtype='object')
```

In [14]:
```python
sc =StandardScaler()#initializing the standardscaler
```

In [55]:
```python
x=pd.DataFrame(x,columns=names)#scaled data turns into array format,converting them into dataframe
```

splitting the data into train and test

In [56]:
```python
from sklearn import model_selection
```

In [57]:
```python
x_train,x_test,y_train,y_test=model_selection.train_test_split(x,y,test_size=0.2,random_state=0)
```

In [ ]: