|  |  |
| --- | --- |
| **EX NO :1(b)** | **Logistic Regression** |
| **DATE:** 19/07/2022 | **URK20AI1061** |

**AIM:**

The purpose of this expriment is to predict solar radiation through Logistic regression.

**DESCRIPTION:**

### Logistic regression is basically a supervised classification algorithm. Logistic regression is a supervised classification algorithm that uses the sigmoid function to predict the probability that a given data entry belongs to the category numbered as "1"And “0”.

**ALGORITHM:**

**STEP 1:** Start

**STEP 2**: Numpy, pandas, and matplotlib must be imported

**STEP 3:** The dataset should be analyzed

**STEP 4:**  From linear\_model, import the LogisticRegession model

**STEP 5:** X\_train and Y\_train data must be passed

**STEP 6:** Predict the values of y\_pred

**STEP 7:** Stop.

**CODE:**

import numpy as np

import pandas as pd

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import classification\_report,confusion\_matrix,accuracy\_score

X=pd.read\_csv('https://github.com/joshuacecil/Datasets/raw/main/Train%20-%20Train.csv')

X.columns

X\_train=X[['Latitude', 'Longitude', 'Altitude', 'min Temo', 'Max Temp','Sunshine Hour']]

X\_train.shape

y\_train=X['Solar Radiation']

y\_train.shape

lr = LogisticRegression()

lr.fit(X\_train,y\_train)

T = pd.read\_csv('https://github.com/joshuacecil/Datasets/raw/main/Solar%20Test%20-%20Solar%20Test.csv')

X\_test=T[['Latitude', 'Longitude', 'Altitude', 'min Temo', 'Max Temp','Sunshine Hour']]

y\_test=T['Solar Radiation']

y\_pred=lr.predict(X\_test)

y\_pred.shape

print(confusion\_matrix(y\_test,y\_pred))

print(classification\_report(y\_test,y\_pred))

a=(accuracy\_score(y\_test,y\_pred))

print(a)

error = 1 - a

print(error)

x1=X.sample()

x1\_test=x1[['Latitude', 'Longitude', 'Altitude', 'min Temo', 'Max Temp','Sunshine Hour']]

**OUTPUT:**

****