Logistic Regression

(L1,L2 applied)

Dear Professor : Mr.Manthouri

Produced By : Ghasemi,morteza

https://github.com/Morteza-Ghasemi

January 2021

Logistic Regression

- Classification is a large domain in the field of statistics and machine learning. Generally, classification can be broken down into two areas
  - **Binary classification**, where we wish to group an outcome into one of two groups.
  - **Multi-class classification**, where we wish to group an outcome into one of multiple (more than two) groups.

Logistic Regression

- Logistic Regression is a type of Generalized Linear Model (GLM).
- uses a logistic function to model a binary variable based on any kind of independent variables.
- LR = LogisticRegression()
- LR. fit(X, y)

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- The task is to predict whether a bank currency note is authentic or not based upon four attributes of the note i.e. skewness of the wavelet transformed image, variance of the image, entropy of the image, and curtosis of the image.
- This is a binary classification problem and then will use **Logistic Regression** algorithm to solve this problem.

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- Importing libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Importing the Dataset**
  - ✓ The data is available for download at the following link: https://drive.google.com/file/d/13nw-uRXPY8XIZQxKRNZ3yYlho-CYm_Qt/view
  - ✓ The detailed information about the data is available at the following link: https://archive.ics.uci.edu/ml/datasets/banknote+authentication

Logistic Regression

Hints :

- the dataset is banknote authentication

Abstract: Data were extracted from images that were taken for the evaluation of an authentication procedure for banknotes.

Data Set Characteristics:	Multivariate	**Number of Instances:**	1372	**Area:**	Computer
Attribute Characteristics:	Real	**Number of Attributes:**	5	**Date Donated**	2013-04-16
Associated Tasks:	Classification	**Missing Values?**	N/A	**Number of Web Hits:**	290722

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Exploratory Data Analysis**
  - ✓ The following code reads bank currency note data into pandas dataframe:

    bankdata = pd.read_csv(r"D:\...\bill_authentication.csv")
  - ✓ To see the rows and columns and of the data, execute the following command:

    bankdata.shape

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Exploratory Data Analysis**
  - ✓ To get a feel of how our dataset actually looks, execute the following command:
    bankdata.head()
  - ✓ You can see that all of the attributes in the dataset are numeric. The label is also numeric i.e. 0 and 1.

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Data Preprocessing**
  - ✓ Dividing the data into attributes and labels :
    - ✓ X = bankdata.drop('Class', axis=1)
    - ✓ y = bankdata['Class']
  - ✓ dividing the data into training and testing sets:
    - ✓ from sklearn.model_selection import train_test_split
    - ✓ X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20)

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Training the Algorithm**
  - ✓ The fit method of **Logistic Regression** class is called to train the algorithm on the training data, which is passed as a parameter to the fit method.
    - ✓ from sklearn.linear_model import LogisticRegression
    - ✓ logreg = LogisticRegression()
    - ✓ logreg.fit(X_train, y_train)

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Making Predictions**
  - ✓ To make predictions, the predict method of the **Logistic Regression** class is used.
    - ✓ y_pred = logreg.predict(X_test)

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Evaluating the Algorithm**
  - ✓ Confusion matrix, precision, recall, and F1 measures are the most commonly used metrics for classification tasks.
    - ✓ from sklearn.metrics import classification_report, confusion_matrix
    - ✓ print(confusion_matrix(y_test,y_pred))
    - ✓ print(classification_report(y_test,y_pred))

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Evaluating the Algorithm(Penalty = None)**
  - ✓ The output of the **Logistic Regression(penalty='none', solver='saga')** looks like this:

```
In [18]: print
[[146   2]
 [  2 125]]

In [19]: print
              precision    recall  f1-score   support

           0       0.99      0.99      0.99       148
           1       0.98      0.98      0.98       127

avg / total       0.99      0.99      0.99       275
```

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Evaluating the Algorithm (Penalty = L1)**
  - ✓ The output of the **Logistic Regression(penalty='l1', solver='liblinear')** looks like this:

```
[[136   4]
 [  0 135]]

In [17]: print
             precision    recall  f1-score   support

          0       1.00      0.97      0.99       140
          1       0.97      1.00      0.99       135

avg / total       0.99      0.99      0.99       275
```

Logistic Regression

Implementing Logistic Regression with Scikit-Learn

- **Evaluating the Algorithm (Penalty = L2)**
  - ✓ The output of the **Logistic Regression(penalty='l2', solver='liblinear')** looks like this:

```
In [18]: print
[[143   1]
 [  0 131]]

In [19]: print
             precision    recall  f1-score   support

          0       1.00      0.99      1.00       144
          1       0.99      1.00      1.00       131

avg / total       1.00      1.00      1.00       275
```

The End