Project - Phishing Detector using LR

Description :

The dataset is a text file which provides the following resources that can be used as inputs for model building :

1. A collection of website URLs for 11000+ websites. Each sample has 30 website parameters and a class label identifying it as a phishing website or not (1 or -1).

2. The code template containing these code blocks:
    a. Import modules (Part 1)
    b. Load data function + input/output field descriptions

The dataset also serves as an input for project scoping and tries to specify the functional and non-functional requirements for it.

Background of the Problem Statement :

You are expected to write the code for a binary classification model (phishing website or not) using Python Scikit-Learn that trains on the data and calculates the accuracy score on the test data. You have to use one or more of the classification algorithms to train a model on the phishing website dataset.

Domain : Cyber Security and Web Mining

Dataset Description :

Data Dictionary – Variable and Description

- **UsingIP** (categorical - signed numeric) **:** { -1,1 }
- **LongURL** (categorical - signed numeric) **:** { 1,0,-1 }
- **ShortURL** (categorical - signed numeric) **:** { 1,-1 }
- **Symbol@** (categorical - signed numeric) **:** { 1,-1 }
- **Redirecting//** (categorical - signed numeric) **:** { -1,1 }
- **PrefixSuffix-** (categorical - signed numeric) **:** { -1,1 }
- **SubDomain**s (categorical - signed numeric) **:** { -1,0,1 }
- **HTTPS** (categorical - signed numeric) **:** { -1,1,0 }
- **DomainRegLen** (categorical - signed numeric) **:** { -1,1 }
- **Favicon** (categorical - signed numeric) **:** { 1,-1 }
- **NonStdPort** (categorical - signed numeric) **:** { 1,-1 }
- **HTTPSDomainURL** (categorical - signed numeric) **:** { -1,1 }
- **RequestURL** (categorical - signed numeric) **:** { 1,-1 }

- **AnchorURL** (categorical - signed numeric) **:** { -1,0,1 }
- **LinksInScriptTags** (categorical - signed numeric) **:** { 1,-1,0 }
- **ServerFormHandler** (categorical - signed numeric) **:** { -1,1,0 }
- **InfoEmail** (categorical - signed numeric) **:** { -1,1 }
- **AbnormalURL** (categorical - signed numeric) **:** { -1,1 }
- **WebsiteForwarding** (categorical - signed numeric) **:** { 0,1 }
- **StatusBarCust** (categorical - signed numeric) **:** { 1,-1 }
- **DisableRightClick** (categorical - signed numeric) **:** { 1,-1 }
- **UsingPopupWindow** (categorical - signed numeric) **:** { 1,-1 }
- **IframeRedirection** (categorical - signed numeric) **:** { 1,-1 }
- **AgeOfDomain** (categorical - signed numeric) **:** { -1,1 }
- **DNSRecording** (categorical - signed numeric) **:** { -1,1 }
- **WebsiteTraffic** (categorical - signed numeric) **:** { -1,0,1 }
- **PageRank** (categorical - signed numeric) **:** { -1,1 }
- **GoogleIndex** (categorical - signed numeric) **:** { 1,-1 }
- **LinksPointingToPage** (categorical - signed numeric) **:** { 1,0,-1 }
- **StatsReport** (categorical - signed numeric) **:** { -1,1 }
- **class** (categorical - signed numeric) **:** { -1,1 }

Dataset Size : 11055 rows x 31 columns

Hint :

- The dataset is a ".txt" file with no headers and has only the column values.

- The actual column-wise header is described above and, if needed, you can add the header manually.

- The header list is as follows :
  ['UsingIP', 'LongURL', 'ShortURL', 'Symbol@', 'Redirecting//',
  'PrefixSuffix-', 'SubDomains', 'HTTPS', 'DomainRegLen', 'Favicon',
  'NonStdPort', 'HTTPSDomainURL', 'RequestURL', 'AnchorURL',
  'LinksInScriptTags', 'ServerFormHandler', 'InfoEmail', 'AbnormalURL',
  'WebsiteForwarding', 'StatusBarCust', 'DisableRightClick',
  'UsingPopupWindow', 'IframeRedirection', 'AgeofDomain',
  'DNSRecording', 'WebsiteTraffic', 'PageRank', 'GoogleIndex',
  'LinksPointingToPage', 'StatsReport', 'class']

Questions to be answered with analysis :

1. Write the code for a binary classification model (phishing website or not) using Python Scikit-Learn that trains on the data and calculates the accuracy score on the test data.

2. Use one or more of the classification algorithms to train a model on the phishing website dataset.

Project Guidelines :

1. **Initiation :**
   - Begin by creating a new ipynb file and load the dataset in it.

2. **Exercise 1 :**
   - Build a phishing website classifier using Logistic Regression with “C” parameter = 100.
   - Use 70% of data as training data and the remaining 30% as test data.
     [Hint: Use Scikit-Learn library LogisticRegression]
     [Hint: Refer to the logistic regression tutorial taught earlier in the course]
   - Print count of misclassified samples in the test data prediction as well as the accuracy score of the model.

3. **Exercise 2 :**
   - Train with only two input parameters - parameter Prefix_Suffix and 13 URL_of_Anchor.
   - Check accuracy using the test data and compare the accuracy with the previous value.
   - Plot the test samples along with the decision boundary when trained with index 5 and index 13 parameters.