Logistics Regression Explained

DECODING DATA SCIENCE

DECODING
DATA SCIENCE

JOIN OUR AI COMMUNITY: HTTPS://NAS.IO/ARTIFICIALINTELLIGENCE

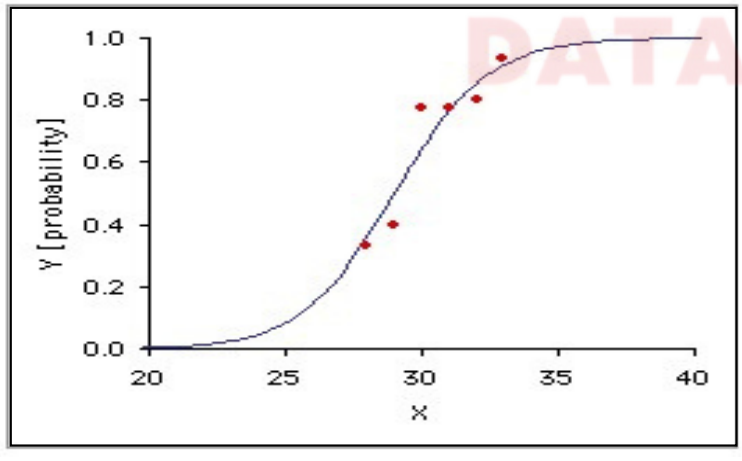
Logistic Regression – Introduction

In Linear regression, the outcome variable is continuous and the predictor variables can be a mix of numeric and categorical. But often there are situations where we wish to evaluate the effects of multiple explanatory variables on a binary outcome variable

For example, the effects of a number of factors on the development or otherwise of a disease. A patient may be cured or not; a prospect may respond or not, should we grant a loan to particular person or not, etc.

When the outcome or dependent variable is binary, and we wish to measure the effects of several independent variables on it, we uses Logistic Regression

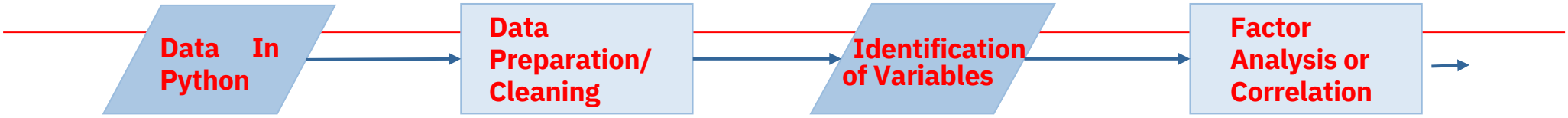
- **The binary outcome variable can be coded as 0 or 1.**
- **The logistic curve is shown in the figure below:**



We estimate the probability of success by the equation:

$$P = \frac{e^{a+bX}}{1+e^{a+bX}}$$

Process Flow

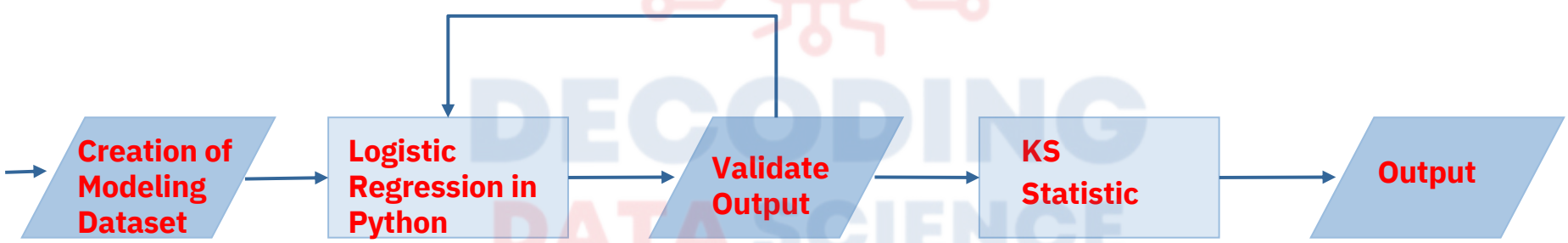


Data is obtained in pandas dataframe

- Missing Value Imputation
- Trash value
- Outlier Treatment

Independent and dependent variables should be identified

- FA is done in order to get the variables into groups
- Good to choose factor solution near the Eigen value of 1
- As a further Check Correlation Analysis is done



Divide data into Development and Validation Sample in ratio 70:30 or 80:20

- Assume all assumptions hold
- Check for the significance of the variables
- Run Regression on Development sample

- Validate the output on the Validation sample , by running the same model on the Validation sample

Results will be presented in PowerPoint

Python code

Step 1: Importing the dataset

```
dataset                                          =
pd.read_csv('car_purchase_Ads.csv')  X  =
dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
```

Step 2: Splitting the dataset into the Training set and Test set

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.25,
random_state = 0)
```

Step 2: Feature Scaling

```
from        sklearn.preprocessing        import
StandardScaler sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

Step 4 : Training the Logistic Regression model on the Training set

```
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(X_train, y_train)
```

Python code

Step 5:Predicting a new result

```
print(classifier.predict(sc.transform([[30,87000]])))
```

Step 6: Predicting the Test set results

```
y_pred =classifier.predict(X_test)
print(np.concatenate((y_pred.reshape(len(y_pred),1),
y_test.reshape(len(y_test),1)),1))
```

Step 7: Making the Confusion Matrix

```
fromsklearn.metricsimport
confusion_matrix,accuracy_score
cm
=confusion_matrix(y_test,y_pred)
print(cm)
accuracy_score(y_test,y_pred)
```

Practice

For location of code and dataset

https://github.com/arshad831/Modelling-Exercise/blob/main/logistic_regression.ipynb

Join Our AI Community

Being part of an AI community like ours offers multiple benefits. You can network with like-minded individuals, learn from experienced professionals, and stay up-to-date with the latest AI trends and developments.

Whether you're a beginner looking to start your journey in AI, or an experienced professional looking to enhance your skills, our community has something to offer.

Join us and be a part of this exciting journey. Learn more, Grow more!

Click the link in the footer to join us today!

JOIN OUR AI COMMUNITY: HTTPS://NAS.IO/ARTIFICIALINTELLIGENCE