# **Logistic regression**

Logistic regression is a statistical method used for binary classification problems, where the outcome is one of two possible values. It's a type of regression analysis used to predict the probability of a certain event occurring.

The goal of logistic regression is to model the probability of a binary outcome (0 or 1, yes or no, true or false) as a function of one or more predictor variables.

**Online Shopper Revenue Prediction**

Given data about online shoppers, let's try to predict whether a given shopper will generate revenue or not.

We will use a logistic regression model to make our predictions.

About The Data

This data set contains the following features:

**Administrative**: Number of pages viewed in the "Administrative" category. This could relate to pages like account settings or customer support.

**Administrative\_Duration**: Total duration (in seconds) spent on "Administrative" pages.

**Informational**: Number of pages viewed in the "Informational" category. This could include pages like FAQs or product information.

**Informational\_Duration**: Total duration (in seconds) spent on "Informational" pages.

**ProductRelated**: Number of pages viewed related to products, such as product listings or details.

**ProductRelated\_Duration**: Total duration (in seconds) spent on pages related to products.

**BounceRates**: Percentage of single-page sessions where the visitor left the site without interacting further.

**ExitRates**: Percentage of sessions where the visitor left the site from a specific page.

**PageValues**: Average value assigned to a page, which can help estimate the revenue generated by visits to that page.

**SpecialDay**: Numeric value representing how close the visit was to a special day (like a holiday). For example, 0.0 might indicate no special day proximity, while higher values might indicate closeness to a special day.

**Month**: The month when the visit occurred, represented as a name (e.g., "Feb" for February).

**OperatingSystems**: Numeric value representing the type of operating system used by the visitor (e.g., 1 for Windows, 2 for MacOS).

**Browser**: Numeric value representing the browser used by the visitor (e.g., 1 for Chrome, 2 for Firefox

**Region**: Numeric value indicating the region from which the visitor is accessing the site (e.g., 1 for North America, 2 for Europe).

**TrafficType**: Numeric value representing the type of traffic source (e.g., 1 for direct, 2 for search engines).

**VisitorType**: Categorical variable indicating whether the visitor is new or returning. Possible values might include "Returning\_Visitor" or "New\_Visitor."

**Weekend**: Boolean value indicating whether the visit occurred on a weekend (e.g., True or False).

**Revenue**: Binary outcome variable indicating whether the visitor made a purchase (e.g., True for a purchase, False for no purchase).

**Encoding-**

Encoding is the process of converting categorical data into numerical form so that machine learning algorithms can process it. Here are some common encoding techniques:

1. **Label Encoding**

 Assigns each category a unique integer value.

* **Example**: For categories ['Red', 'Green', 'Blue'], the encoding might be {'Red': 0, 'Green': 1, 'Blue': 2}.
* Suitable when the categorical feature has an ordinal relationship (i.e., there's a meaningful order).

1. **One-Hot Encoding**

* Creates binary columns for each category, where each column represents one category.
* **Example**: For categories ['Red', 'Green', 'Blue'], the one-hot encoding would create three columns: Red, Green, Blue, where only one column is 1 and the others are 0 for each instance.
* Ideal for categorical features without an ordinal relationship.

You can choose one of the encoding methods based on your data's nature and apply it consistently across all categorical features (e.g., month, visitor type, and boolean variables like weekend and revenue).

**Standardization** is used when the goal is to adjust the data to have a mean of 0 and a standard deviation of 1. It is particularly useful when the data follows a normal distribution or when working with algorithms that assume normally distributed data.

**Normalization** is used when the goal is to scale the data within a fixed range, such as [0, 1]. This is useful for algorithms that require data to be within a specific range, like neural networks

**Confusion Matrix**

**True Positives (TP)**: 145  
(Correctly predicted positive cases)

**True Negatives (TN)**: 2009  
(Correctly predicted negative cases)

**False Positives (FP)**: 46  
(Incorrectly predicted as positive, but actually negative)

**False Negatives (FN)**: 266  
(Incorrectly predicted as negative, but actually positive)

**Classification Report**

1. Precision

High precision means that when the model predicts a positive class, it is often correct.

( true positive predictions out of all positive predictions)

1. Recall

High recall means that the model identifies most of the positive instances.( true positive predictions out of all actual positive instances.)

1. F1-Score

The F1-score is useful when you need a balance between precision and recall, especially when dealing with imbalanced datasets.

1. Support

The number of actual occurrences of the class in the dataset.

**Data Questions:**

* How did you handle missing values and duplicates in the dataset? What steps did you take to ensure the data was clean before applying logistic regression?
* What specific challenges did you face in encoding categorical features such as Month, VisitorType, and Weekend, and how did you overcome them?
* How did you split the data into training and testing sets? What considerations did you have in mind while deciding the split ratio?
* How did you evaluate the performance of your logistic regression model? What practical steps did you follow to compute and interpret metrics such as accuracy, precision, recall, and F1-score
* How did you evaluate the model's performance?