Universidad de Guadalajara

Centro Universitario de Ciencias Exactas e Ingenierías

División de ingenierías para la integración ciber-humana

Profesora: Gerardo Garcia Gil
Juan Antonio Pérez Juárez
215660996

Introduction

In the rapidly evolving landscape of data science and artificial intelligence, the ability to make accurate predictions is paramount. While linear regression has long been a foundational tool for modeling relationships between variables, many real-world problems require us to predict categorical outcomes rather than continuous values. This is where logistic regression emerges as a powerful and versatile technique.

Logistic regression is widely used for classification tasks, enabling researchers and practitioners to estimate the probability that a given input belongs to a particular category. From medical diagnoses and financial risk assessments to spam detection and image recognition, logistic regression provides a mathematically rigorous yet intuitively accessible framework for decision-making.

This article explores the fundamental principles of logistic regression, its mathematical underpinnings, and practical applications. By understanding how logistic regression transforms data into actionable insights, we can better appreciate its significance in modern analytical workflows and its role in shaping intelligent systems.

Logistic Regression

In statistics, a logistic model (or logit model) is a statistical model that models the log-odds of an event as a linear combination of one or more independent variables. In regression analysis, logistic regression (or logit regression) estimates the parameters of a logistic model (the coefficients in the linear or non linear combinations). In binary logistic regression there is a single binary dependent variable, coded by an indicator variable, where the two values are labeled "0" and "1", while the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value). The corresponding probability of the value labeled "1" can vary between 0 (certainly the value "0") and 1 (certainly the value "1"), hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name. The unit of measurement for the log-odds scale is called a logit, from the logistic unit, hence the alternative names.

Binary variables are widely used in statistics to model the probability of a certain class or event taking place, such as the probability of a team winning, of a patient being healthy, etc. and the logistic model has been the most commonly used model for binary regression since about 1970. Binary variables can be generalized to categorical variables when there are more than two possible values (e.g. whether an image is of a cat, dog, lion, etc.), and the binary logistic regression generalized to multinomial logistic regression. If the multiple categories are ordered, one can use the ordinal logistic regression (for example the proportional odds ordinal logistic mode). The logistic regression model itself simply models probability of output in terms of input and does not perform statistical classification (it is not a classifier), though it can be used to make a classifier, for instance by choosing a cutoff value and classifying inputs with probability greater than the cutoff as one class, below the cutoff as the other; this is a common way to make a binary classifier.

Analogous linear models for binary variables with a different sigmoid function instead of the logistic function (to convert the linear combination to a probability) can also be used, most notably the probit model; The defining characteristic of the logistic model is that increasing one of the independent variables multiplicatively scales the odds of the given outcome at a constant rate, with each independent variable having its own parameter; for a binary dependent variable this generalizes the odds ratio. More

abstractly, the logistic function is the natural parameter for the Bernoulli distribution, and in this sense is the "simplest" way to convert a real number to a probability.

The parameters of a logistic regression are most commonly estimated by maximum-likelihood estimation (MLE). This does not have a closed-form expression, unlike linear least squares; Logistic regression by MLE plays a similarly basic role for binary or categorical responses as linear regression by ordinary least squares (OLS) plays for scalar responses: it is a simple, well-analyzed baseline model; The logistic regression as a general statistical model was originally developed and popularized primarily by Joseph Berkson, beginning in Berkson (1944), where he coined "logit".

Logistic regression is used in various fields, including machine learning, most medical fields, and social sciences. For example, the Trauma and Injury Severity Score (TRISS), which is widely used to predict mortality in injured patients, was originally developed by Boyd et al. using logistic regression.[6] Many other medical scales used to assess severity of a patient have been developed using logistic regression. Logistic regression may be used to predict the risk of developing a given disease (e.g. diabetes; coronary heart disease), based on observed characteristics of the patient (age, sex, body mass index, results of various blood tests, etc.). Another example might be to predict whether a Nepalese voter will vote Nepali Congress or Communist Party of Nepal or for any other party, based on age, income, sex, race, state of residence, votes in previous elections, etc. The technique can also be used in engineering, especially for predicting the probability of failure of a given process, system or product. It is also used in marketing applications such as prediction of a customer's propensity to purchase a product or halt a subscription, etc. In economics, it can be used to predict the likelihood of a person ending up in the labor force, and a business application would be to predict the likelihood of a homeowner defaulting on a mortgage. Conditional random fields, an extension of logistic regression to sequential data, are used in natural language processing. Disaster planners and engineers rely on these models to predict decisions taken by householders or building occupants in small-scale and large-scales evacuations, such as building fires, wildfires, hurricanes among others. These models help in the development of reliable disaster managing plans and safer design for the built environment.

Logistic regression is a supervised machine learning algorithm widely used for binary classification tasks, such as identifying whether an email is spam or not and diagnosing diseases by assessing the presence or absence of specific conditions based on patient test results. This approach utilizes the logistic (or sigmoid) function to transform a linear combination of input features into a probability value ranging between 0 and 1. This probability indicates the likelihood that a given input corresponds to one of two predefined categories. The essential mechanism of logistic regression is grounded in the logistic function's ability to model the probability of binary outcomes accurately. With its distinctive S-shaped curve, the logistic function effectively maps any real-valued number to a value within the 0 to 1 interval. This feature renders it particularly suitable for binary classification tasks, such as sorting emails into "spam" or "not spam". By calculating the probability that the dependent variable will be categorized into a specific group, logistic regression provides a probabilistic framework that supports informed decision-making.

Definition

A dataset contains N points. Each point i consists of a set of m input variables x1,i ... xm,i (also called independent variables, explanatory variables, predictor variables, features, or attributes), and a binary outcome variable Yi (also known as a dependent variable, response variable, output variable, or class), i.e. it can assume only the two possible values 0 (often meaning "no" or "failure") or 1 (often meaning "yes" or "success"). The goal of logistic regression is to use the dataset to create a predictive model of the outcome variable.

As in linear regression, the outcome variables *Yi* are assumed to depend on the explanatory variables $x_{1,i} \ldots x_{m,i}$.

Explanatory variables

The explanatory variables may be of any type: real-valued, binary, categorical, etc. The main distinction is between continuous variables and discrete variables.

(Discrete variables referring to more than two possible choices are typically coded using dummy variables (or indicator variables), that is, separate explanatory variables taking the value 0 or 1 are created for each possible value of the discrete variable, with a 1 meaning "variable does have the given value" and a 0 meaning "variable does not have that value".)

Outcome variables

Formally, the outcomes Yi are described as being Bernoulli-distributed data, where each outcome is determined by an unobserved probability pi that is specific to the outcome at hand, but related to the explanatory variables. This can be expressed in any of the following equivalent forms:

$$Y_i \mid x_{1,i}, \ldots, x_{m,i} \sim \text{Bernoulli}(p_i)$$

$$\mathbb{E}[Y_i \mid x_{1,i}, \ldots, x_{m,i}] = p_i$$

$$\Pr(Y_i = y \mid x_{1,i}, \ldots, x_{m,i}) = \begin{cases} p_i & \text{if } y = 1 \\ 1 - p_i & \text{if } y = 0 \end{cases}$$

$$\Pr(Y_i = y \mid x_{1,i}, \ldots, x_{m,i}) = p_i^y (1 - p_i)^{(1-y)}$$

The meanings of these four lines are:

The first line expresses the probability distribution of each Yi : conditioned on the explanatory variables, it follows a Bernoulli distribution with parameters pi, the probability of the outcome of 1 for trial i. As noted above, each separate trial has its own probability of success, just as each trial has its own explanatory variables. The probability of success pi is not observed, only the outcome of an individual Bernoulli trial using that probability.

The second line expresses the fact that the expected value of each Yi is equal to the probability of success pi, which is a general property of the Bernoulli distribution. In other words, if we run a large number of Bernoulli trials using the same probability of success pi, then take the average of all the 1 and 0 outcomes, then the result would be close to pi. This is because doing an average this way simply computes the proportion of successes seen, which we expect to converge to the underlying probability of success.

The third line writes out the probability mass function of the Bernoulli distribution, specifying the probability of seeing each of the two possible outcomes.

The fourth line is another way of writing the probability mass function, which avoids having to write separate cases and is more convenient for certain types of calculations. This relies on the fact that Yi can take only the value 0 or 1. In each case, one of the exponents will be 1, "choosing" the value under it, while the other is 0, "canceling out" the value under it. Hence, the outcome is either pi or 1 – pi, as in the previous line.

Linear predictor function

The basic idea of logistic regression is to use the mechanism already developed for linear regression by modeling the probability pi using a linear predictor function, i.e. a linear combination of the explanatory

variables and a set of regression coefficients that are specific to the model at hand but the same for all trials. The linear predictor function *f(i)* for a particular data point i is written as:

$f(i) = \beta_0 + \beta_1 x_{1,i} + \cdots + \beta_m x_{m,i}$, where B0, … Bm are regression coefficients indicating the relative effect of a particular explanatory variable on the outcome.

The model is usually put into a more compact form as follows:

The regression coefficients β0, β1, ..., βm are grouped into a single vector β of size m + 1.
For each data point i, an additional explanatory pseudo-variable x0,i is added, with a fixed value of 1, corresponding to the intercept coefficient β0.
The resulting explanatory variables x0,i, x1,i, ..., xm,i are then grouped into a single vector Xi of size m + 1.
This makes it possible to write the linear predictor function as follows:

$f(i) = \boldsymbol{\beta} \cdot \mathbf{X}_i,$

using the notation for a dot product between two vectors.

Example
Suppose we want to predict whether a student is admitted (1) or not admitted (0) to a university based on their scores in two exams.

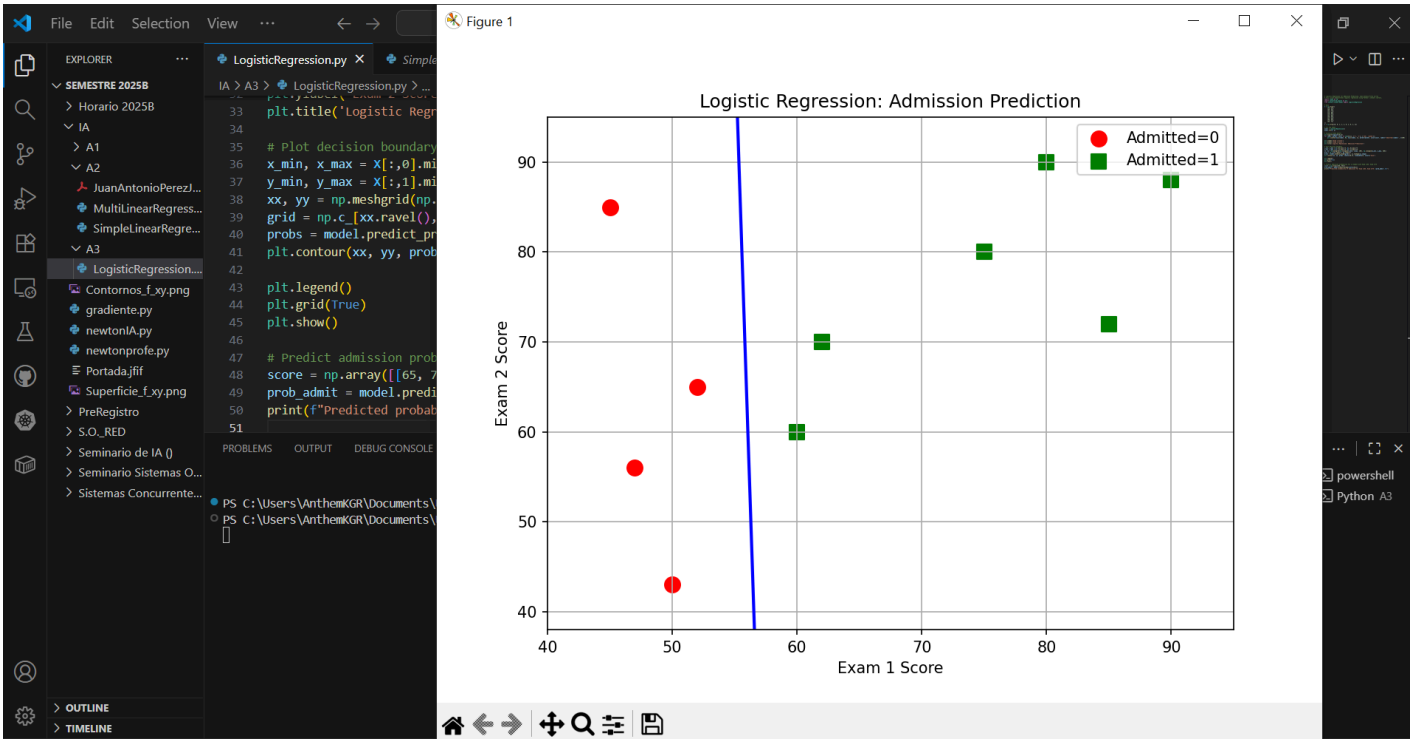
Exam 1 Score	Exam 2 Score	Admitted (1) / Not Admitted (0)
45	85	0
50	43	0
62	70	1
75	80	1
80	90	1
52	65	0
60	60	1
47	56	0
90	88	1
85	72	1

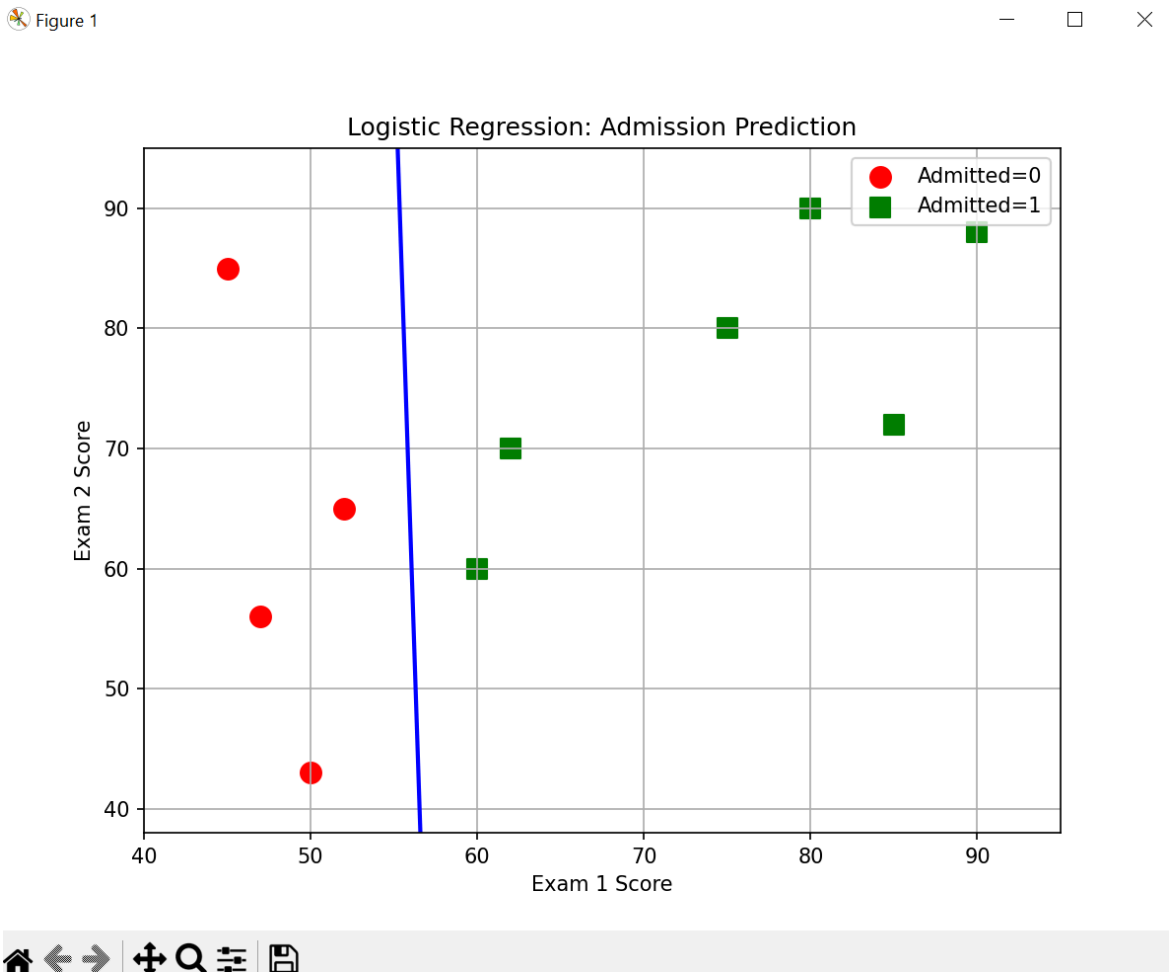
Let’s code it

```python
# Logistic Regression for Admission Prediction, Juan Antonio Pérez Juárez
# This script demonstrates logistic regression using Python's sklearn library.
import numpy as np
import matplotlib.pyplot as plt
```

```
from sklearn.linear_model import LogisticRegression

# Data
X = np.array([
    [45, 85],
    [50, 43],
    [62, 70],
    [75, 80],
    [80, 90],
    [52, 65],
    [60, 60],
    [47, 56],
    [90, 88],
    [85, 72]
])
y = np.array([0, 0, 1, 1, 1, 0, 1, 0, 1, 1])

# Fit the model
model = LogisticRegression()
model.fit(X, y)

# Visualize data points
plt.figure(figsize=(8,6))
for label, marker, color in zip([0,1], ['o', 's'], ['red', 'green']):
    plt.scatter(X[y==label, 0], X[y==label, 1], marker=marker, color=color,
label=f'Admitted={label}', s=100)

plt.xlabel('Exam 1 Score')
plt.ylabel('Exam 2 Score')
plt.title('Logistic Regression: Admission Prediction')

# Plot decision boundary
x_min, x_max = X[:,0].min()-5, X[:,0].max()+5
y_min, y_max = X[:,1].min()-5, X[:,1].max()+5
xx, yy = np.meshgrid(np.linspace(x_min, x_max, 100), np.linspace(y_min, y_max, 100))
grid = np.c_[xx.ravel(), yy.ravel()]
probs = model.predict_proba(grid)[:, 1].reshape(xx.shape)
plt.contour(xx, yy, probs, levels=[0.5], linewidths=2, colors='blue')

plt.legend()
plt.grid(True)
plt.show()

# Predict admission probability for a student with Exam 1=65, Exam 2=75
score = np.array([[65, 75]])
prob_admit = model.predict_proba(score)[0][1]
print(f"Predicted probability of admission for Exam 1=65, Exam 2=75: {prob_admit:.2f}")
```

Screenshots





The plot visually shows how logistic regression separates admitted and not admitted students based on their exam scores.
The blue line is the “tipping point” where the model changes its prediction.

It helps you see, for any new student, whether they're likely to be admitted just by looking at where their scores fall on the plot.

Conclusion

Logistic regression may seem simple at first glance, but its power and versatility make it a cornerstone of modern classification tasks. Whether predicting admissions, diagnosing medical conditions, or filtering spam emails, logistic regression helps us turn data into actionable decisions. By understanding how it draws boundaries between outcomes, we gain valuable insight into the patterns hidden in our data.

Referencias

Wikipedia contributors, "Logistic regression," Wikipedia, Jul. 24, 2025.
https://en.wikipedia.org/wiki/Logistic_regression

"LogisticRegression," Scikit-learn.
https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

A. Navlani, "Understanding logistic regression in Python," Aug. 11, 2024.
https://www.datacamp.com/tutorial/understanding-logistic-regression-python