Contents

Chapter 1

Introduction to Classification and Regression

Classification and regression are two types of supervised learning methods in machine learning and statistics. In supervised learning, the goal is to learn a mapping function from inputs $\mathbf{x}$ to an output y, given a labeled set of input-output pairs.

1.1 Classification Systems

In classification, the output y is a categorical or discrete value. For example, if we are developing a system to predict whether an email is spam or not, y can take two values: "spam" or "not spam". This is an example of a binary classification problem.

Classification problems that have more than two categories are known as multi-class classification problems. For example, predicting the species of an iris flower from a set of measurements of its petals and sepals is a multi-class classification problem, as there are three species of iris flowers.

1.2 Regression Systems

In regression, the output y is a continuous value. For example, if we are developing a system to predict the price of a house given features like its size, location, number of rooms, etc., the output is a continuous number which represents the price.

1.3 Algorithms

There are many algorithms used to solve classification and regression problems, ranging from simple ones like linear regression for regression problems and logistic regression for binary classification problems, to more complex ones like neural networks, which can be used for both classification and regression problems.

1.4 Performance Metrics

Performance of classification and regression models is evaluated using different metrics. For classification, these include accuracy, precision, recall, and F1 score. For regression, common metrics include mean absolute error, mean squared error, and R-squared.

Chapter 2

Simple Linear Regression

Simple linear regression is a statistical method that allows us to summarize and study relationships between two continuous (quantitative) variables:

- One variable, denoted x, is regarded as the predictor, explanatory, or independent variable.
- The other variable, denoted y, is regarded as the response, outcome, or dependent variable.

Because the other terms are used less frequently today, we'll use the "predictor" and "response" terms to refer to the variables encountered in this course. The other terms are mentioned only to make you aware of them should you encounter them in other contexts.

Simple linear regression gets its adjective "simple," because it concerns the study of only one predictor variable. In contrast, multiple linear regression, a topic that will be covered later, gets its adjective "multiple," because it concerns the study of two or more predictor variables.

2.0.1 The model behind simple linear regression

Given a scatterplot of the response variable y versus the predictor variable x, we fit the line

$$y = \beta_0 + \beta_1 x + \epsilon \tag{2.1}$$

that minimizes the distances from the observed points to the line!

- y = dependent variable (output/outcome/prediction/estimation)
- β_0 = y-intercept (constant term)
- β_1 = slope of the regression line (the effect that X has on Y)
- x = independent variable (input variable used in the prediction of Y)
- ϵ = error (the difference between the actual and predicted/estimated value)

This line can be used to predict future values of y given new data values of x.

Chapter 3

Simple Logistic Regression

While linear regression is used for predicting a continuous response variable, logistic regression is used for predicting a categorical response variable. It's particularly useful when the response variable is binary (i.e., it takes on only two possible outcomes, usually coded as 0 and 1).

The primary idea behind logistic regression is to find the probability of the response variable being true (1) given the values of the predictor variables.

In simple logistic regression, we have only one predictor variable. The form of the logistic regression model is:

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x \tag{3.1}$$

where:

- p is the probability of the positive class (i.e., the outcome $y = 1$).
- β_0 and β_1 are the parameters of the model.
- x is the predictor variable.

On the left-hand side, we have the natural log of the odds ratio (also called the logit), rather than just p itself. This is done to ensure that the predicted probabilities lie between 0 and 1. The function $\frac{p}{1-p}$ is called the odds, and can take any value between 0 and ∞.

In contrast to linear regression, where the parameters are estimated using least squares, the parameters in logistic regression are usually estimated using maximum likelihood estimation.

Maximum likelihood estimation finds the parameter values that make the observed data most likely under the model.

In a simple logistic regression model, the probability that $Y = 1$ given x is:

$$p(x) = \frac{e^{\beta_0+\beta_1 x}}{1 + e^{\beta_0+\beta_1 x}} \tag{3.2}$$

And the probability that $Y = 0$ given x is:

$$1 - p(x) = \frac{1}{1 + e^{\beta_0 + \beta_1 x}} \tag{3.3}$$

Chapter 4

Standardizing Data

Data standardization is a preprocessing step in many machine learning algorithms. Standardization transforms the variables in the dataset to have a mean of zero and a standard deviation of one.

The standardization of a variable X is calculated as follows:

$$Z = \frac{X - \mu}{\sigma} \tag{4.1}$$

where:

- Z is the standardized variable.
- X is the original variable.
- μ is the mean of X.
- σ is the standard deviation of X.

4.1 Why Do We Standardize Data?

There are several reasons why standardization is essential:

4.1.1 Homogeneity of Variances

Some statistical techniques assume that all variables have the same variance. Standardizing the data ensures this assumption.

4.1.2 Interpreting Coefficients

In regression analysis, standardizing allows us to interpret the coefficients of the predictors as the change in the response variable associated with a one-standard-deviation increase in the predictor.

4.1.3 Algorithm Convergence

For many machine learning algorithms (like gradient descent), standardization can help the algorithm converge more quickly to the optimum.

4.1.4 Comparing Variables

Standardization puts different variables on the same scale, allowing for meaningful comparisons. For example, it would be challenging to compare a variable measured in kilograms with another measured in kilometers without standardization.

4.1.5 Preventing Numerical Instabilities

Standardizing can help prevent numerical instabilities in computations, particularly when dealing with high-dimensional data.

Remember, though standardization is useful and necessary in many situations, it's not always required. For instance, tree-based models are scale-invariant and don't require standardization.

Chapter 5

One-Hot Encoding

In machine learning and data analysis, it is common to encounter categorical variables. Categorical data refers to variables that contain label values rather than numeric values. Examples include color (”red”, ”blue”, ”green”), size (”small”, ”medium”, ”large”), or geographic designations (city names, country names, etc.). Most machine learning algorithms require numerical input and output variables. One-hot encoding is a process of converting categorical data into a format that could be provided to machine learning algorithms to improve prediction.

5.1 Why One-Hot Encoding?

While some machine learning algorithms can work with categorical variables directly, many machine learning algorithms cannot operate on label data. They require all input variables and output variables to be numeric. Hence, categorical data needs to be converted to a numerical form. One-hot encoding is a popular method to transform categorical variables into a format that works better with classification and regression algorithms.

5.2 How does One-Hot Encoding Work?

In one-hot encoding, for each unique value in the categorical variable, we create a new binary feature that takes a value of 1 if the original feature value matches the unique value and 0 otherwise. If a categorical variable has n unique values, we would create n new features.

For example, consider the categorical variable ”color” with three categories: ”red”, ”blue”, and ”green”. The one-hot encoding process will result in three new features, namely ”is_red”, ”is_blue”, and ”is_green”.

Color	is_red	is_blue	is_green
red	1	0	0
blue	0	1	0
green	0	0	1
red	1	0	0

This encoding helps to convey the information in the categorical variable to the learning algorithm effectively.

However, it's worth noting that one-hot encoding can significantly increase the dimensionality of the data, which can be problematic for some models. Therefore, it is not always the best choice, and other encoding methods might be more suitable depending on the situation.

Chapter 6

Multiple Logistic Regression

The simple logistic regression model, discussed in the last chapter, uses only one predictor variable, while multiple logistic regression, as the name implies, allows for more than one predictor variable.

6.1 Multiple Logistic Regression

In multiple logistic regression, we want to model the relationship between a binary response variable and multiple predictor variables. Let y be the binary response variable and $x_1, x_2, ..., x_p$ be p predictor variables. The multiple logistic regression model has the form:

$$\ln\left(\frac{P(Y=1|X)}{1-P(Y=1|X)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p$$

where $P(Y=1|X)$ is the probability of the event $Y=1$ given the predictor variables, and $\beta_0, \beta_1, ..., \beta_p$ are the parameters of the model. This equation can also be rewritten in terms of the probability $P(Y=1|X)$:

$$P(Y=1|X) = \frac{e^{\beta_0+\beta_1 X_1+\beta_2 X_2+...+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\beta_2 X_2+...+\beta_p X_p}}$$

In this model, each one-unit increase in X_i multiplies the odds of $Y=1$ by e^{β_i}, holding all other predictors constant.

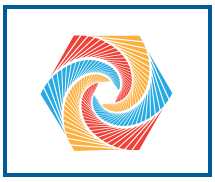
6.2 Divide by 4 Rule

The "Divide by 4" rule is a rule of thumb for interpreting the coefficients in logistic regression. It says that for small values of β_i, a one-unit increase in X_i will change the probability $P(Y=1|X)$ by approximately $\beta_i/4$ at the average value of X_i.

The rule arises from the derivative of the logistic function at its midpoint, and provides a useful and simple way to get an approximate sense of the effect size when interpreting the coefficients.

Appendix A

Answers to Exercises



Index