

Elements of statistical learning: Chapter 2

July 13, 2020

- 1 Introduction
- 2 Linear models and least squares
 - Linear regression
 - Linear classification
- 3 Nearest-neighbour algorithm
- 4 Statistical decision theory
 - L2 loss function
 - L1 loss function
- 5 MSE

Introduction

Supervised learning

Goal is to use *inputs* to predict *outputs*.

- inputs are also referred to as *predictors*, *features* or *independent variable*
- outputs are also referred to as *response* or *dependent variables*

The outputs may be

- quantitative (takes values in \mathbb{R})
- qualitative (also known as categorical or discrete)
 - Ordered (e.g. small, medium or large)
 - Unordered (e.g. pass or fail, on or off)

Regression vs classification

We use *regression* to predict quantitative outputs and *classification* to predict qualitative outputs.

Notations

For different predictors $\{X_k\}_{k=1}^p$ across different observations $i = 1, \dots, n$ denote

$X_{i,k}$ for random variable and $x_{i,k}$ for an observation

X_i for a $p \times 1$ vector of variables – i.e. $X_i = [X_{i,1}, \dots, X_{i,p}]'$

\mathbf{X} for a $N \times p$ matrix of variables across different obsvs

In other words

$$\mathbf{X} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ X_{21} & \cdots & X_{2p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix} = \begin{bmatrix} X'_1 \\ X'_2 \\ \vdots \\ X'_n \end{bmatrix}$$

Therefore, denotes (X_i, Y_i) as the random variables and (x_i, y_i) as the observed values at i .

Linear models and least squares

To predict Y (which would be denoted by \hat{Y}), we use the linear regression model, which may be expressed as

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \cdots + \beta_p X_{i,p} + \epsilon_i, \quad i = 1, \dots, n \quad (1)$$

or as

$$Y_i = \beta_0 + \sum_{k=1}^p \beta_k X_{i,k} + \epsilon_i, \quad i = 1, \dots, n \quad (2)$$

or as

$$Y_i = X_i' \beta + \epsilon_i, \quad i = 1, \dots, n \quad (3)$$

where $X_i = [1, X_{i,1}, \dots, X_{i,p}]'$ and $\beta = [\beta_0, \beta_1, \dots, \beta_p]'$ are $(p+1) \times 1$ vectors.

In matrix notation, the above can be expressed as

$$\mathbf{Y} = \mathbf{X}\beta + \epsilon \quad (4)$$

which if expanded is expressed as follows

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{1,1} & X_{1,2} & \cdots & X_{1,p} \\ 1 & X_{2,1} & X_{2,2} & \cdots & X_{2,p} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 1 & X_{n,1} & X_{n,2} & \cdots & X_{n,p} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} \quad (5)$$

Least squares estimation

The least squares estimation is one approach to fit the model. In essence, we find the coefficients β that minimize the sum of squared residuals.

Thus, we wish to minimize $RSS(\beta)$

$$\arg \min_{\beta} RSS(\beta) \text{ or } (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$$

The above can be rearranged and expanded to (which is not necessary, as chain rule can be used)

$$\begin{aligned} (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) &= (\mathbf{y}' - \beta'\mathbf{X}')(\mathbf{y} - \mathbf{X}\beta) \\ &= \mathbf{y}'\mathbf{y} - \mathbf{y}'\mathbf{X}\beta - \beta'\mathbf{X}'\mathbf{y} + \beta'\mathbf{X}'\mathbf{X}\beta \end{aligned}$$

Differentiating with respect to β yields

$$\begin{aligned}\frac{\partial RSS(\beta)}{\partial \beta} &= 0 - \mathbf{X}'\mathbf{y} - \mathbf{X}'\mathbf{y} + 2\beta\mathbf{X}'\mathbf{X} \\ &= -2\mathbf{X}'\mathbf{y} + 2\beta\mathbf{X}'\mathbf{X}\end{aligned}$$

and as this is a minimization problem

$$\begin{aligned}\frac{\partial RSS(\beta)}{\partial \beta} &= 0 \\ -2\mathbf{X}'\mathbf{y} + 2\beta\mathbf{X}'\mathbf{X} &= 0 \\ \beta\mathbf{X}'\mathbf{X} &= \mathbf{X}'\mathbf{y}\end{aligned}$$

and thus so long as $\mathbf{X}'\mathbf{X}$ is non-singular, the LS estimator of $\hat{\beta}$ is

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} \quad (6)$$

Logit models

Now let us consider the case , where the dependent variable Y_i can assume only two categories (say win or lose), and hence two discrete values (i.e. $Y_i = 0$ or $Y_i = 1$), where as the vector of independent variables are continuous, say $X_i \in \mathbb{R}^p$.

In order to restrict Y_i to 0 and 1. In this case it would make sense to make the probability of $Y_i = 1$ and not the value of Y_i itself. This leads to a probability model, which specifies the the probability of the outcome as a function of the predictor:

$$P[Y_i = 1] = P[X_i, \beta] \quad (7)$$

$$P[Y_i = 0] = 1 - P[X_i, \beta] \quad (8)$$

Since P is a probability, it is bounded between 0 and 1. The regression equation may be revived by briefly denoting

$$P(X_i, \beta) = X_i' \beta$$

As we wish the probability to vary monotonically with X , we may use a *sigmoid* function:

$$P(X_i, \beta) = \frac{\exp(\beta' X_i)}{1 + \exp(\beta' X_i)} \quad (9)$$

Let us denote $Z_i = \beta' X_i$, then

$$\lim_{z \rightarrow \infty} \frac{\exp(z)}{1 + \exp(z)} = 1$$

and

$$\lim_{z \rightarrow -\infty} \frac{\exp(z)}{1 + \exp(z)} = 0$$

Therefore,

$$P[Y_i = 1] = \frac{\exp(\beta'X)}{1 + \exp(\beta'X)}$$

and

$$P[Y_i = 0] = \frac{1}{1 + \exp(\beta'X)}$$

Alternatively, one could look at the odd $P[Y_i = 1]/P[Y_i = 0]$, which may be expressed as

$$\begin{aligned} \frac{P[Y_i = 1]}{P[Y_i = 0]} &= \frac{\exp(\beta'X)}{1 + \exp(\beta'X)} [1 + \exp(\beta'X)]. \\ &= \exp(\beta'X). \end{aligned}$$

now taking the logarithm from both sides will yield

$$\log(odds) = \beta'X \tag{10}$$

where now $\log(odds)$ is no longer bounded by 0 and 1.

Nearest-neighbour algorithm

- A non-parametric approach used for both *classification* and *regression*
- Input consists of the k closest training examples in the feature space.
- Output depends on whether $K - NN$ is used for classification or regression.
- For classification, the output is a class membership
- For regression, this value is the average of the k nearest neighbours
- Specifically it can be defined as

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i, \quad (11)$$

where $N_k(x)$ is the neighbourhood of x defined by the k closest points x_i in the training sample.

- In other words, find the k nearest neighbours with x_i closest to x , and average their responses y_i .

L2 loss function

- Let random variables $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}$ with joint distribution function $F(x, y)$.
- A function $g(X)$ is sought after for predicting Y , given values of the input X .
- This theory requires a loss function $L(Y, g(X))$ for penalizing errors in prediction.
- The most common and convenient is squared error loss

$$L(Y, g(X)) = (Y - g(X))^2 \quad (12)$$

which gives us the following criterion

$$EPE = \mathbb{E}[(Y - g(X))^2] \quad (13)$$

EXTRA: Some probability recap

The expectation operator \mathbb{E} is defined for discrete and continuous variables as follows

- Discrete variables:

$$\mathbb{E}(X) = \sum_{i \in k} p_i x_i$$

where k are the number of categories. E.g. A coin has two possible states of head (quantified as 1) and tail (quantified as 0) with equal probability. Therefore, the expected value of the outcome of a coin toss is

$$\mathbb{E}(X) = 0.5 \times 0 + 0.5 \times 1 = 0.5$$

- Continuous variables:

$$\mathbb{E}[X] = \int_{\mathbb{R}} xf(x)dx$$

if density $f(x)$ exists. Otherwise,

$$\mathbb{E}[X] = \int_{\mathbb{R}} x dF(x)$$

(noting that $dF(X)/dx = f(x)$)

- Multivariate continuous variable

$$\mathbb{E}[g(X_1, \dots, X_n)] = \int_{x_1} \cdots \int_{x_n} g(x_1, \dots, x_n) dF(x_1, \dots, x_n)$$

and where the density $f(x_1, \dots, x_n)$ exists

$$\mathbb{E}[g(X_1, \dots, X_n)] = \int_{x_1} \cdots \int_{x_n} g(x_1, \dots, x_n) f(x_1, \dots, x_n) dx_1 \cdots dx_n$$

Finally, note that

$$F(x, y) = F(x | y)F(y)$$

$$F(x, y) = F(y | x)F(x)$$

Therefore, returning to (13)

$$\begin{aligned}\mathbb{E}[(Y - g(X))^2] &= \int_x \int_y (y - g(x))^2 dF(y, x) \\ &= \int_x \int_y (y - g(x))^2 f(y, x) dy dx \\ &= \int_x \int_y (y - g(x))^2 f(y | x) f(x) dy dx \\ &= \int_x \mathbb{E}_{y|x}[(Y - g(X))^2 | X] f(x) dx \\ &= \mathbb{E}_X \mathbb{E}_{Y|X}[(Y - g(X))^2 | X]\end{aligned}$$

- As the above expression is conditioned on X , there is no longer any dependency between X and the function g .
- furthermore, $[Y - g]^2$ is a convex function and we may minimize to solve for f

$$g(x) = \arg \min_g \mathbb{E}_{y|x}[(Y - g(X))^2 | X = x]$$

$$\rightarrow \frac{\partial}{\partial g} \int [y - g]^2 f(y | x) dy = 0$$

$$\rightarrow \int \frac{\partial}{\partial g} [y - g]^2 f(y | x) dy = 0$$

$$\rightarrow -2 \int [y - g] f(y | x) dy = 0$$

$$\rightarrow 2g \int f(y | x) dy = 2 \int y f(y | x) dy$$

$$\rightarrow 2g \int f(y | x) dy = 2\mathbb{E}_{Y|X}[Y | X = x]$$

Note that

$$\int_{\mathbb{R}} f(y | x) dy = 1$$

- Thus

$$\mathbb{E}[Y | X = x] = g(x)$$

- This conditional function is also known as the regression function. Thus, the best predictor of Y at any point $X = x$ is the conditional mean.

L1 loss function

- The L2 loss function is analytically more desirable, but an L1 criteria of the sort

$$\mathbb{E}[|Y - g(X)|]$$

is more robust to outliers. We may express (13) in the following manner using the L1 criteria:

$$\mathbb{E}[|Y - g(X)|] = \int \int |Y - g(X)| f(x, y) dy dx$$

where as before it may be expressed as

$$\mathbb{E}_X \mathbb{E}_{Y|X}[|Y - g(X)| | X]$$

and minimized by differentiating with respect to g - i.e.

$$\frac{\partial}{\partial g} \int |y - g(x)| f(y | x) dy = 0$$

The latter can be approximated by

$$\begin{aligned}\frac{\partial}{\partial g} \int |y - g| f(y | x) dy &= \frac{\partial}{\partial g} \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n |y_i - g| \\ &\approx \frac{\partial}{\partial g} \frac{1}{n} \sum_{i=1}^n |y_i - g|\end{aligned}\tag{14}$$

Note that the absolute function is piecewise

$$|y_i - g| = \begin{cases} y_i - g, & y_i - g > 0 \\ g - y_i, & y_i - g < 0 \\ 0, & y_i = g \end{cases}$$

So that taking the derivative is not continuous at zero

$$\frac{\partial}{\partial g}|y_i - g| = \begin{cases} -1, & y_i - g > 0 \\ 1, & y_i - g < 0 \\ 0, & y_i = g \end{cases}$$

which is very similar to a sign function - i.e.

$$\text{sgn}(z) = \begin{cases} -1, & z < 0 \\ 1, & z > 0 \\ 0, & z = 0 \end{cases}$$

which implies that we may express (14) as

$$\begin{aligned} \frac{\partial}{\partial g} \int |y - g| f(y | x) dy &= 0 \\ \frac{1}{n} \sum_{i=1}^n -\text{sgn}(y_i - g) &= 0 \\ \sum_{i=1}^n \text{sgn}(y_i - g) &= 0 \end{aligned}$$

The mean squared error at a nominated test point x_0 can be obtained by

$$\begin{aligned}MSE(x_0) &= \mathbb{E}_T[(f(x_0) - \hat{y}_0)^2] \\&= \mathbb{E}_T[(f(x_0) - \underbrace{\mathbb{E}_T(\hat{y}_0) + \mathbb{E}_T(\hat{y}_0) - \hat{y}_0}_{\text{add and deduct}})^2]\end{aligned}$$

Noting that by noting $\underbrace{f(x_0) - \mathbb{E}_T(\hat{y}_0)}_a$ and $\underbrace{\mathbb{E}_T(\hat{y}_0) - \hat{y}_0}_a$, the above is equivalent to $\mathbb{E}_T[(a + b)^2]$, and such can be expanded as follows

$$\begin{aligned}MSE(x_0) &= \mathbb{E}_T[(f(x_0) - \mathbb{E}_T(\hat{y}_0))^2 + (\mathbb{E}_T(\hat{y}_0) - \hat{y}_0)^2 \\&\quad + 2[(f(x_0) - \mathbb{E}_T(\hat{y}_0))(\mathbb{E}_T(\hat{y}_0) - \hat{y}_0)]\end{aligned}$$

which may further get expanded as

$$\begin{aligned}MSE(x_0) &= \mathbb{E}_T[(f(x_0) - \mathbb{E}_T(\hat{y}_0))^2 + (\mathbb{E}_T(\hat{y}_0) - \hat{y}_0)^2] \\&+ 2f(x_0)\mathbb{E}_T(\hat{y}_0) - 2f(x_0)\hat{y}_0 - 2\mathbb{E}_T(\hat{y}_0)\mathbb{E}_T(\hat{y}_0) + 2\mathbb{E}_T(\hat{y}_0)\hat{y}_0] \\&= \mathbb{E}_T[(f(x_0) - \mathbb{E}_T(\hat{y}_0))^2] + \mathbb{E}_T[(\mathbb{E}_T(\hat{y}_0) - \hat{y}_0)^2] \\&+ 2f(x_0)^2 - 2f(x_0)\hat{y}_0 - 2f(x_0)^2 + 2f(x_0)\hat{y}_0 \\&= \mathbb{E}_T[(f(x_0) - \mathbb{E}_T(\hat{y}_0))^2] + \mathbb{E}_T[(\mathbb{E}_T(\hat{y}_0) - \hat{y}_0)^2]\end{aligned}$$

Note that by definition

- Variance: $\mathbb{E}_T[(f(x_0) - \mathbb{E}_T(\hat{y}_0))^2]$; and
- Bias: $\mathbb{E}_T[(\mathbb{E}_T(\hat{y}_0) - \hat{y}_0)]$

Thus,

$$MSE(x_0) = \sigma_T^2(\hat{y}_0) + Bias^2(\hat{y}_0) \quad (15)$$