Regression

Ruwen Qin
Stony Brook University

September 16, 2025

Data used in this module:

- Concrete_Data.csv
- Diamond_Data.csv
- MINIST dataset from sklearn.datasets

Python notebook used in this module

- Regression.ipynb

Considering an entity characterized by an N dimensional feature representation: $\boldsymbol{X} = [X_1, \ldots, X_N]$. We would like to infer measurements of interest, $\boldsymbol{Y} = [Y_1, \ldots, Y_P]$, from the features. Regression is one of the most widely used of all statistical methods to establish the mapping between $\boldsymbol{X}$ and $\boldsymbol{Y}$. In a regression model, the feature representation is the input, named predictors or independent variables. The measurements we would like to infer are the outputs and named responses or dependent variables. The responses and predictors can take both numerical and categorical values. Extra coding effort is needed if a variable in the regression model is a categorical one. This learning module is developed mainly based on references [2, 3].

1 Multiple Linear Regression

Multiple Linear Regression (MLR) handles the special case where there is one response and multiple predictors.

1.1 Maximum Likelihood Estimation (MLE)

Consider a training dataset with M data points: $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_M, y_M)$, where $\boldsymbol{x}_i = [x_{i,1}, \ldots, x_{i,N}]$ are the predictors of a data point in the training dataset and y_i is the response. We assume the data points are independent and identically distributed. The response Y and the predictors $\boldsymbol{X}$ has a linear relationship that we would like to identify from the training data. The assumed target function with noise can be expressed in the following form:

$$p(y_i|\boldsymbol{z}_i, \boldsymbol{w}, \sigma_\epsilon) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\left(-\frac{(y_i - \boldsymbol{z}_i\boldsymbol{w})^2}{2\sigma_\epsilon^2}\right), \tag{1}$$

where $\boldsymbol{z}_i = [1, \boldsymbol{x}_i]$, σ_ϵ^2 is the variance of regression residuals, and $\boldsymbol{w} = [w_0, \ldots, w_N]^\mathrm{T}$ is the vector of regression coefficients. $\boldsymbol{w}$ and σ_ϵ are learnable parameters of this regression model.

Equation (1) assumes that y_i follows a normal distribution,

$$y_i \sim \mathcal{N}(\boldsymbol{z}_i\boldsymbol{w}, \sigma_\epsilon^2). \tag{2}$$

The joint distribution of the independent responses, $\boldsymbol{y} = [y_1, \ldots, y_M]^\mathrm{T}$, conditional on the feature representation and the model parameters is,

$$p(\boldsymbol{y}|\boldsymbol{z}, \boldsymbol{w}, \sigma_\epsilon) = \prod_{i=1}^{M} p(y_i|\boldsymbol{z}_i; \boldsymbol{w}, \sigma_\epsilon^2), \tag{3}$$

which measures the likelihood of the model with respect to the training data.

We usually work with the log likelihood,

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}, \sigma_\epsilon) &= \log\left(\prod_{i=1}^{M} p(y_i|\boldsymbol{z}_i; \boldsymbol{w}, \sigma_\epsilon)\right) \\ &= -\frac{M}{2}\log(2\pi\sigma_\epsilon^2) - \frac{1}{2}\sum_{i=1}^{M}\frac{(y_i - \boldsymbol{z}_i\boldsymbol{w})^2}{\sigma_\epsilon^2}.\end{aligned} \tag{4}$$

The maximum likelihood estimation (MLE) is about seeking the model parameters that can maximize the log likelihood:

$$\widehat{\boldsymbol{w}}, \widehat{\sigma}_\epsilon = \underset{\boldsymbol{w}, \sigma_\epsilon}{\arg\max}\, \mathcal{L}(\boldsymbol{w}, \sigma_\epsilon). \tag{5}$$

When some assumptions hold, to maximize the log likelihood function is about to minimize the total of squared regression residuals measured on the training dataset, leading to the least squares estimation (LSE).

1.2 Ordinary Least Squares Estimator

1.2.1 The Data and Hypothesis Set

Consider a training dataset with M data points. Each data point consists of N predictors and one response. The dataset in the matrix form includes

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_M \end{bmatrix} = \begin{bmatrix} x_{1,1} & \dots & x_{1,N} \\ \vdots & \ddots & \vdots \\ x_{M,1} & \dots & x_{M,N} \end{bmatrix}, \tag{6}$$

and

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix}, \tag{7}$$

The multiple linear regression model relating the response y_i to predictors $x_1, \dots, x_N$ is

$$y_i = w_0 + w_1 x_{i,1} + \cdots + w_N x_{i,N} + \epsilon_i, \tag{8}$$

for $i = 1, \dots, M$. ϵ_i in (8) is called the residual, noise, disturbance, or error.

Let

$$\mathbf{Z} = [\mathbf{1}\ \mathbf{X}] = \begin{bmatrix} \boldsymbol{z}_1 \\ \vdots \\ \boldsymbol{z}_M \end{bmatrix} = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,N} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{M,1} & \dots & x_{M,N} \end{bmatrix}, \tag{9}$$

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_M \end{bmatrix}, \tag{10}$$

$$\boldsymbol{w} = \begin{pmatrix} w_0 \\ \vdots \\ w_N \end{pmatrix}. \tag{11}$$

Then, the multiple linear regression model can be expressed as

$$\boldsymbol{y} = \mathbf{Z}\boldsymbol{w} + \boldsymbol{\epsilon}. \tag{12}$$

Assumptions of the linear regression model are:

1. linearity of the conditional expectation: $\mathrm{E}(y_i|\boldsymbol{z}_i; \boldsymbol{w}) = \boldsymbol{z}_i\boldsymbol{w}$;
2. independent noise: $\epsilon_1, \dots, \epsilon_M$ are independent;
3. constant variance: $\mathrm{Var}(\epsilon_i) = \sigma_\epsilon^2$ for all i;
4. Gaussian noise: ϵ_i is normally distributed for all i.

1.2.2 Least Squares Estimator (LSE)

The least-squares estimator (LSE) for $\boldsymbol{w}$ minimizes the square of L_2 distance between the true responses and their predictions:

$$\|\boldsymbol{y} - \mathbf{Z}\boldsymbol{w}\|_2^2 = (\boldsymbol{y} - \mathbf{Z}\boldsymbol{w})^{\mathrm{T}}(\boldsymbol{y} - \mathbf{Z}\boldsymbol{w}) = \boldsymbol{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\boldsymbol{w} - 2\boldsymbol{w}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\boldsymbol{y} + \boldsymbol{y}^{\mathrm{T}}\boldsymbol{y}, \tag{13}$$

which is a quadratic function of $\boldsymbol{w}$. By setting the derivatives of the objective function in (13) with respect to $\boldsymbol{w}$ equal to 0 and simplifying the resulting equation, one finds that the LSE of $\boldsymbol{w}$ is [1]

$$\widehat{\boldsymbol{w}} = (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\boldsymbol{y}. \tag{16}$$

[1] It should be mention that $\mathbf{Z}^{\mathrm{T}}\mathbf{Z}$ may not be invertible. If that is the case, the sudo-inverse of $\mathbf{Z}^{\mathrm{T}}\mathbf{Z}$ is used to obtain the solution $(\mathbf{X}^{\mathrm{T}}\mathbf{Z})^{\dagger}\mathbf{Z}^{\mathrm{T}}\boldsymbol{y}$. Specifically, let $\mathbf{A} \in \mathbb{R}^{M \times N}$ be a matrix that one would like to find the sudo-inverse. One can perform

The vector of fitted response values is

$$\widehat{\boldsymbol{y}} = \mathbf{Z}\widehat{\boldsymbol{w}} = \{\mathbf{Z}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\}\boldsymbol{y} = \mathbf{H}\boldsymbol{y}, \tag{17}$$

where $\mathbf{H} = \{\mathbf{Z}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\}$ is the *hat matrix*. $\widehat{\boldsymbol{y}} = [\widehat{y}_1, \ldots, \widehat{y}_M]^{\mathrm{T}}$ estimates $\mathrm{E}(\boldsymbol{y}|\mathbf{Z})$, the expectation of response conditional on the predictors and the LSE of the model parameters.

1.2.3 Regression Residuals

Let

$$\widehat{\boldsymbol{\epsilon}} = \boldsymbol{y} - \widehat{\boldsymbol{y}}, \tag{18}$$

and $\widehat{\epsilon}_i$ denote the ith element of $\widehat{\boldsymbol{\epsilon}}$. Then, an unbiased estimator of σ_ϵ^2 is

$$\widehat{\sigma}_\epsilon^2 = \frac{\widehat{\boldsymbol{\epsilon}}^{\mathrm{T}}\widehat{\boldsymbol{\epsilon}}}{M-N-1} = \frac{\sum_{i=1}^{M}\widehat{\epsilon}_i^2}{M-N-1} = \frac{\sum_{i=1}^{M}(y_i - \widehat{y}_i)^2}{M-N-1}. \tag{19}$$

1.2.4 Confidence Interval Estimation of Regression Coefficients

The point estimator of $\boldsymbol{w}$ in (16) is random. [2] The expectation of $\widehat{\boldsymbol{w}}$ is

$$\mathrm{E}[\widehat{\boldsymbol{w}}|\mathbf{Z}, \boldsymbol{y}] = (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\mathrm{E}[\boldsymbol{y}] = (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\widehat{\boldsymbol{w}} = \mathbf{I}\widehat{\boldsymbol{w}} = \widehat{\boldsymbol{w}}, \tag{20}$$

and the covariance matrix of $\widehat{\boldsymbol{w}}$ is

$$\begin{aligned}\mathrm{Cov}(\widehat{\boldsymbol{w}}|\mathbf{Z}, \boldsymbol{y}) &= (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\mathrm{Cov}(\boldsymbol{y}|\mathbf{Z})\mathbf{Z}((\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1})^{\mathrm{T}} \\ &= (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}(\sigma_\epsilon^2\mathbf{I})\mathbf{Z}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1} \\ &= \sigma_\epsilon^2(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}.\end{aligned} \tag{21}$$

Therefore, the standard error (se) of $\widehat{w}_j$ is the square root of the jth element on the diagonal of $\sigma_\epsilon^2(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}$:

$$\mathrm{se}(\widehat{w}_j) = \widehat{\sigma}_\epsilon \mathrm{diag}\left(\sqrt{(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}}\right)_j. \tag{22}$$

An interval estimation of the regression coefficients at the $1-\alpha$ confidence level can be obtained:

$$\widehat{\boldsymbol{w}} \pm t_{1-\alpha/2, M-N-1}\mathrm{se}(\widehat{\boldsymbol{w}}) \tag{23}$$

where $M-N-1$ is the degrees of freedom for the regression residuals, $t_{1-\alpha/2,M-N-1}$ is the t value providing an area of $\alpha/2$ in the right tail of the t-distribution with $M-N-1$ degrees of freedom.

1.3 Regularized Linear Regression

We may add one or more regulation of the regression coefficients $\boldsymbol{w}$ for various purposes, for example preventing model over-fitting. We add a regularizer to the objective function of the least squared estimation:

$$\min_{\boldsymbol{w}} \|\boldsymbol{y} - \mathbf{Z}\boldsymbol{w}\|_2^2 + \lambda\|\boldsymbol{w}\|_p \tag{24}$$

where λ determines the intensity of regularization of the regression coefficients.

the singular decomposition of A:

$$A = \mathbf{U}\begin{pmatrix} \mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}\mathbf{V}^{\mathrm{T}}, \tag{14}$$

where $\mathbf{S}$ is the section containing all positive eighenvalues of $\mathbf{A}$. The sudo-inverse of $\mathbf{A}$ is

$$\mathbf{A}^{\dagger} = \mathbf{V}\begin{pmatrix} \mathbf{S}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}\mathbf{U}^{\mathrm{T}}, \tag{15}$$

[2] Suppose $\mathbf{U} \sim N(\mu, \Sigma)$ is a multivariate normal vector. $\mathbf{V} = \mathbf{c} + \mathbf{D}\mathbf{U}$ is a linear transfer of $\mathbf{U}$, where $\mathbf{c}$ is a vector and $\mathbf{D}$ is a matrix. $\mathbf{V} \sim N(\mathbf{c} + \mathbf{D}\mu, \mathbf{D}\Sigma\mathbf{D}^{\mathrm{T}})$.

When $p = 1$, it is *Lasso regression* that encourages the use of less predictors for the regression:

$$\min_{\boldsymbol{w}} \|\boldsymbol{y} - \mathbf{Z}\boldsymbol{w}\|_2^2 + \lambda \|\boldsymbol{w}\|_1. \tag{25}$$

When $p = 2$, it is the *Ridge regression* that penalizes large regression coefficients to avoid over-fitting.

$$\min_{\boldsymbol{w}} \|\boldsymbol{y} - \mathbf{Z}\boldsymbol{w}\|_2^2 + \lambda \|\boldsymbol{w}\|_2. \tag{26}$$

If both norm-1 and norm-2 regularizers are added to the objective function, it becomes *Elastic Net regression.*

$$\min_{\boldsymbol{w}} \|\boldsymbol{y} - \mathbf{Z}\boldsymbol{w}\|_2^2 + \lambda(\gamma \|\boldsymbol{w}\|_1 + (1-\gamma)\|\boldsymbol{w}\|_2), \tag{27}$$

where $\gamma \in \mathbb{R} \cap [0, 1]$ is the norm-1 ratio that indicates the weight on the norm-1 regularizer. It becomes the Lasso regression if γ=1, and Ridge regression if $\gamma = 0$,

Please refer to [2] for the detailed discussion of regularized linear regression.

1.4 Analysis of Variance

1.4.1 Degree of Freedom

There are degrees of freedom (dof) associated with each of these sources of variation. The dof for regression is N, the number of predictors. The total degrees of freedom is $M - 1$. The degrees of freedom for regression residuals is $M - N - 1$.

1.4.2 Variance Partitioning

The total variation in $\boldsymbol{y}$ (SST) can be partitioned into two parts: the variation that can be predicted by predictors (SSR) and the variation that cannot be predicted (SSE). Mathematically,

$$\begin{aligned} \text{SST} &= \sum_{i=1}^{M} (y_i - \overline{y})^2 \\ &= \sum_{i=1}^{M} (\widehat{y}_i - \overline{y})^2 + \sum_{i=1}^{M} (y_i - \widehat{y}_i)^2 \\ &= \text{SSR} + \text{SSE} \end{aligned} \tag{28}$$

where $\overline{y}$ is the sample mean computed based on $\boldsymbol{y}$.

1.4.3 Mean Squares of Error

The mean sum of squares is defined to be the ratio of sum of squares to its degrees of freedom.

Therefore, the mean squares total is

$$\text{MST} = \frac{\text{SST}}{M-1}, \tag{29}$$

the mean squares of regression is

$$\text{MSR} = \frac{\text{SSR}}{N}, \tag{30}$$

and the mean squares of error is

$$\text{MSE} = \frac{\text{SSE}}{M-N-1}. \tag{31}$$

which is the unbiased estimator of σ_ϵ^2.

1.4.4 R-Squared, R^2

R-squared, denoted by R^2, is

$$R^2 = \frac{\text{SSR}}{\text{SST}} = 1 - \frac{\text{SSE}}{\text{SST}}, \tag{32}$$

and it measures the proportion of the total variation in $\boldsymbol{y}$ that can be linearly predicted by predictors $\mathbf{X}$.

1.4.5 Adjusted R^2

R^2 is biased in favor of large models, because R^2 is always increased by adding more predictors to the model, even if they are independent of the response. The bias in R^2 can be reduced by using adjusted R^2,

$$R^2_{\text{adj}} = 1 - \frac{\text{MSE}}{\text{MST}}, \tag{33}$$

which penalizes large N, the number of predictors. R^2_{adj} can either increase or decrease when more predictors are added to the model. R^2_{adj} increases if the added variables decrease the SSE enough to compensate for the increase in N.

1.4.6 F Test

F test of goodness-of-fit uses the test statistic F:

$$F = \frac{\text{MSR}}{\text{MSE}} \tag{34}$$

to examine the hypotheses:
H_0: $w_0 = \cdots = w_n = 0\,; \forall j$
H_1: $\exists\, w_j \neq 0\ j \in \{0, \ldots, N\}$

Rejecting the null hypothesis means a regression relationship exists between the predictors and the response variable. The p value corresponding to this test statistic is the probability that a random value drawn from the F distribution with N and $M - N - 1$ degrees of freedom is greater than the test statistic. The larger the test stastistic, the smaller the p value and so the chance of making a mistake in rejecting the null hypothesis.

1.5 Statistical Inference of Regression Coefficients

Let $\widehat{w}_j$, for $j = 0, 1, \ldots, N$, be the point estimate of the model coefficient w_j, and $\text{se}(\widehat{w})_j$ be the standard error of $\widehat{w}_j$.

$$\text{se}(\widehat{w}_j) = \widehat{\sigma}_\epsilon \sqrt{\text{diag}[(\mathbf{Z}^{\text{T}}\mathbf{Z})^{-1}]_j} \tag{35}$$

according to (21). The interval estimator of w_j at a $(1 - \alpha)$ confidence level is:

$$\widehat{w}_j \pm t_{1-\alpha/2, M-N-1} \text{se}(\widehat{w}_j) \tag{36}$$

The test statistic to determine

$$t = \frac{\widehat{w}_j}{\text{se}(\widehat{w}_j)} \tag{37}$$

is calculated to examine the hypotheses:
H_0: $w_j = 0$
H_1: $w_j \neq 0$

The p value corresponding to the test statistic is the probability that a random value drawn from the t distribution with $M - N - 1$ degrees of freedom is greater than the test statistic. The smaller the p value, the smaller the chance of making a mistake in rejecting H_0.

1.6 Model Selection

When there are many potential predictors, often we wish to find a subset of them that provide a parsimonious regression model. F-test is not very suitable for model selection. One problem is that there are many possible F-tests and the joint statistical behavior of all of them is not known.

For model selection, it is more appropriate to use a model selection criterion such as Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC). For linear regression models,

$$\text{AIC} = -2\widehat{\mathcal{L}} + 2(1+N), \tag{38}$$

$$\text{BIC} = -2\widehat{\mathcal{L}} + \log(M)(1+N), \tag{39}$$

where $\widehat{\mathcal{L}}$ is the maximium log-likelihood of the regression:

$$\begin{aligned}\widehat{\mathcal{L}} =&\mathcal{L}(\widehat{\boldsymbol{w}}, \widehat{\sigma}_\epsilon)\\ =&-\frac{M}{2}\log(2\pi\widehat{\sigma}_\epsilon^2) - \frac{1}{2}\sum_{i=1}^{M}\frac{(y_i - \boldsymbol{z}_i\widehat{\boldsymbol{w}})^2}{\sigma_\epsilon^2}\\ =&-\frac{M}{2}\log 2\pi - M\log\widehat{\sigma}_\epsilon - \frac{M-N-1}{2}.\end{aligned} \tag{40}$$

The smaller AIC and BIC are, the better the model. BIC penalizes more on larger N (i.e., a more complex model) than AIC does.

AIC, BIC, R^2, and R^2_{adj} can be used together for model selection.

1.7 Prediction Accuracy

The key concept associated with measuring forecast accuracy is forecast error. Various metrics are used to determine how well a particular forecasting method performs. Given $\boldsymbol{x}_i$ from the test set, a regression model predicts the response $\widehat{y}_i$. Multiple metrices are defined for assessing the prediction error for a numerical response:

Mean Absolute Error (MAE)

$$\text{MAE} = \frac{\sum_{i=1}^{L}|y_i - \widehat{y}_i|}{L}, \tag{41}$$

Mean Square Error (MSE)

$$\text{MSE} = \frac{\sum_{i=1}^{L}(y_i - \widehat{y}_i)^2}{L}, \tag{42}$$

Root Mean Square Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{L}(y_i - \widehat{y}_i)^2}{L}}, \tag{43}$$

where L is the size of test dataset.

2 An Example of Multiple Linear Regression

2.1 The Problem

Concrete has 8 attributes:

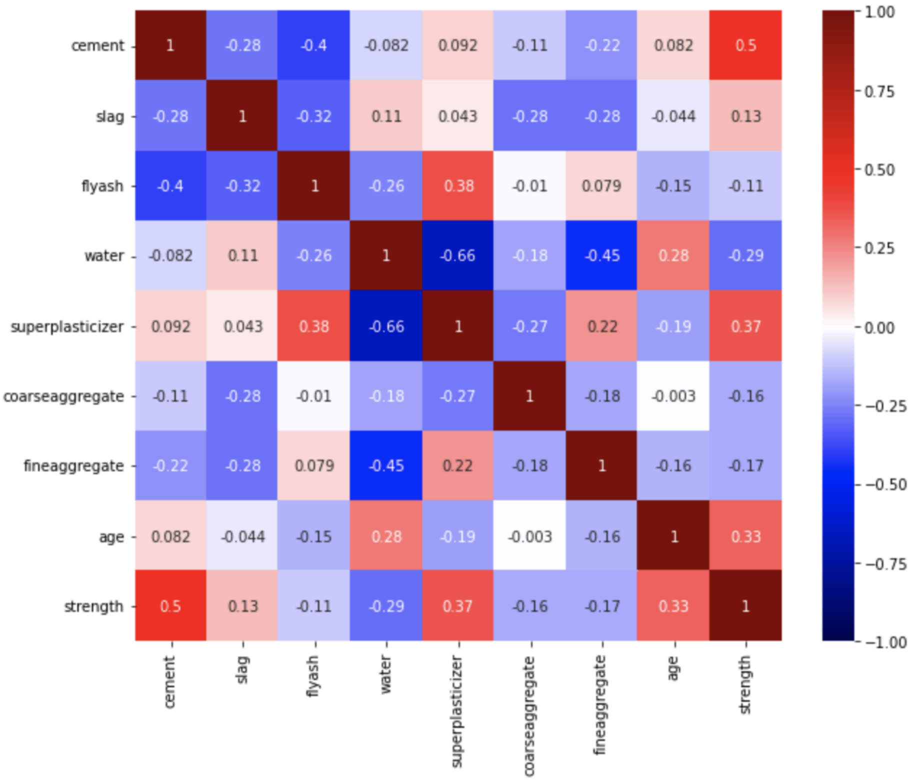
1. Cement – quantitative – kg in a m^3 mixture
2. Blast Furnace Slag – quantitative – kg in a m^3 mixture
3. Fly Ash – quantitative – kg in a m^3 mixture

4. Water – quantitative – kg in a m^3 mixture
5. Superplasticizer – quantitative – kg in a m^3 mixture
6. Coarse Aggregate – quantitative – kg in a m^3 mixture
7. Fine Aggregate – quantitative – kg in a m^3 mixture
8. Age – quantitative – Day (1∼365)

We would like to build a regression model that either uses these hand-crafted features or extracted ones to predict concrete's compressive strength in megapascal (MPa).

2.2 Feature Selection

The following heatmap visualizes the correlation coefficient matrix of the dataset. Do you observe correlations between features?



	cement	slag	flyash	water	superplasticizer	coarseaggregate	fineaggregate	age	strength
cement	1	-0.28	-0.4	-0.082	0.092	-0.11	-0.22	0.082	0.5
slag	-0.28	1	-0.32	0.11	0.043	-0.28	-0.28	-0.044	0.13
flyash	-0.4	-0.32	1	-0.26	0.38	-0.01	0.079	-0.15	-0.11
water	-0.082	0.11	-0.26	1	-0.66	-0.18	-0.45	0.28	-0.29
superplasticizer	0.092	0.043	0.38	-0.66	1	-0.27	0.22	-0.19	0.37
coarseaggregate	-0.11	-0.28	-0.01	-0.18	-0.27	1	-0.18	-0.003	-0.16
fineaggregate	-0.22	-0.28	0.079	-0.45	0.22	-0.18	1	-0.16	-0.17
age	0.082	-0.044	-0.15	0.28	-0.19	-0.003	-0.16	1	0.33
strength	0.5	0.13	-0.11	-0.29	0.37	-0.16	-0.17	0.33	1

Figure 1: Correlation matrix

2.2.1 Extract Features using PCA

To avoid the multicollinearity issue, we could use PCA to transform data into their principal components. We can keep all the 8 principle components for creating the initial regression model.

2.2.2 Distribution of Extracted Features

The distribution plots of the extracted features indicate not all of them are normally distributed. but the skewness is not severe. Therefore, we temporarily ignore this issue and may revisit it when needed.

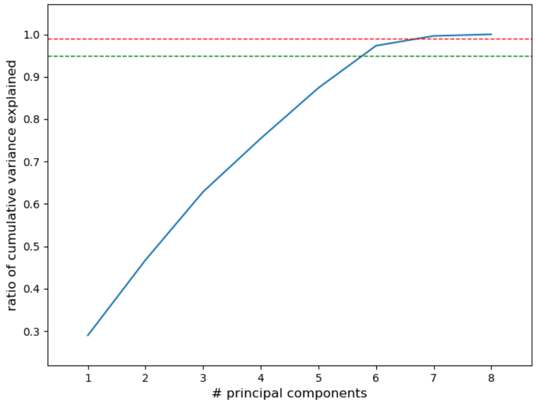


Figure 2: Proportion of cumulative variance explained by top principal components

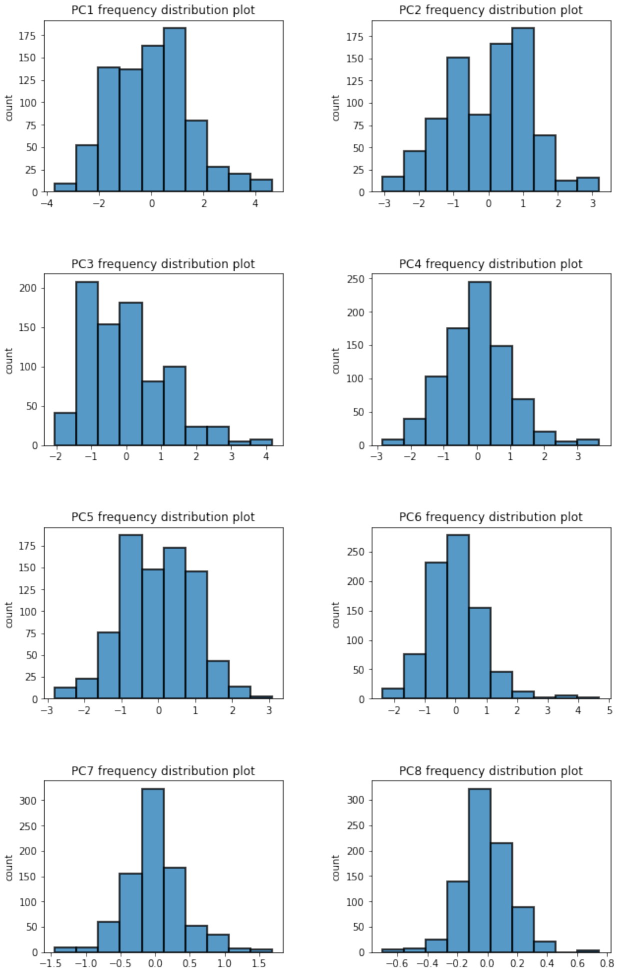


Figure 3: Distribution of principal components

2.3 Ordinary Linear Regression

2.3.1 The Attained Model

We fit a linear regression model that predicts the compressive strength of concrete using the 8 principal components extracted from the eight concrete attributes as predictors. The 95% confidence intervals of the coefficients for C1 and C2 contain zero, indicating these two PCs may not be important predictors.

```
     Coefficient  Margin of error
w0        35.666      0.722
w1        -0.156      0.474
w2        -0.468      0.605
w3         7.375      0.636
w4         1.329      0.719
w5        -5.037      0.739
w6         7.749      0.809
w7        -7.370      1.680
w8        11.752      4.150
```

2.3.2 Residuals

First, we visually checked the residuals from the regression to make sure assumptions about residuals for the regression model hold. If needed, quantitative test results for the regression model can be calculated and they are more precise information.

Figure 4 is the plots of residuals attained on the training and testing datasets. The plots indicate residuals are independent.

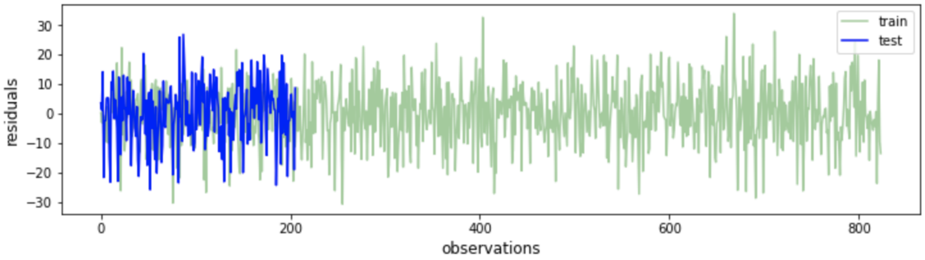


Figure 4: Residuals

Figure 5 shows the distributions of residuals obtained on the training and the testing datasets. The distributions are close to a normal distribution.

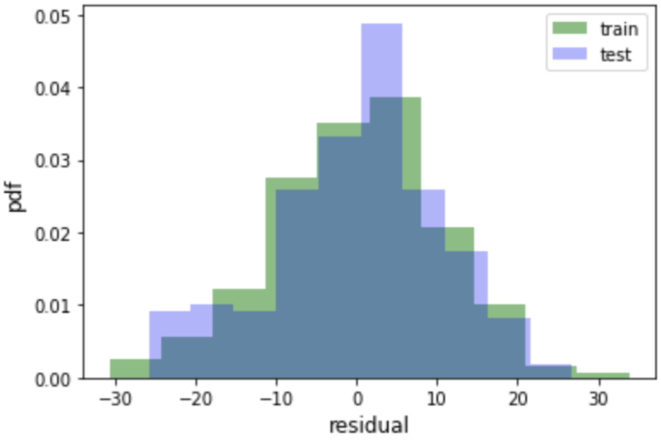


Figure 5: Residual Distribution

Figure 6 shows residuals at predicted response values. The plot shows the distribution of residuals increases when the predicted response value increases. Residuals are not homogeneously distributed. This raises an alarm. Therefore, the current linear model might not be the best choice. We will revisit this issue in later sections. In the following, we continued evaluating this linear model.

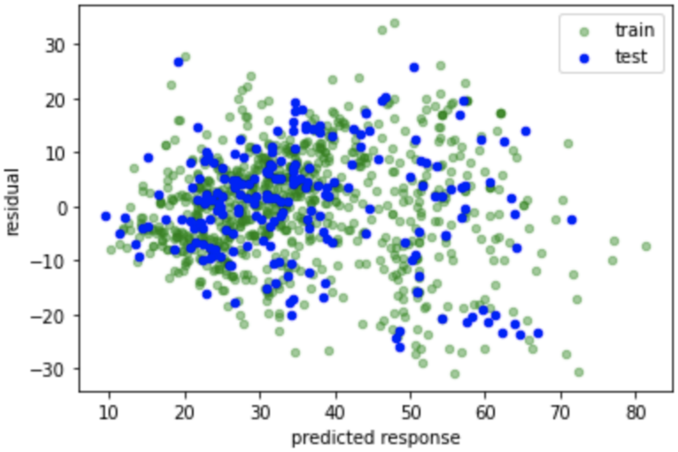


Figure 6: Residuals vs. predicted response

2.3.3 Goodness-of-Fit

In this example, the average R^2 achieved on the training dataset using a five-fold cross validation is 56.41%, meaning the amount of data variation captured by the model.

In testing the model on a test dataset, the accuracy is measured as below:

```
MAE: 7.618
MSE: 99.141
RMSE: 9.957
R Square: 0.616
```

2.3.4 Model Selection

It was identified that PC1 and PC2 may not be important predictors. We tried to drop one feature, or two (PC1 and PC2 are selected). The results are below. No significant improvements were observed. Adding more predictor variables might help improve the prediction ability of the model.

Model	R^2	AIC	BIC
all PCs	0.564	-6050	-6083
drop PC1	0.572	-6050	-6083
drop PC2	0.566	-6052	-6085
drop PC1 and PC2	0.573	-6052	-6085

In testing the three models above on the test dataset, we also didn't see significant change about the test accuracy measures.

2.4 Regularization

We fit a Lasso regression model by setting λ (the intensity of regularization) at 0.1, the point estimates of the regression coefficients for C1 and C8 are zeros (i.e., $\widehat{w}_1 = \widehat{w}_8 = 0$). This suggests dropping the first two principal components from the model. We also fit a Ridge regression model, and a Elastic Net model. Choosing the hypoparameters for those models needs to be addressed. We fine tuned the Elastic Net model and found that the regularization strength $\lambda = 0.0582$ and the norm-1 ratio $\gamma = 1$ are suitable hyperparameters for the Elastic Net. The values indicate that only norm-1 regularizer plays a more important role, but the overall intensity of the regularization is small.

3 Polynomial Linear Regression

Polynomial linear regression fits a polynomial function of predictors $\boldsymbol{X} = [X_1, \ldots, X_N]$ to predict the response Y. A polynomial function in N predictors with degree of K is:

$$\sum_{l=1}^{L} w_l \prod_{j=1}^{N} x_j^{e_{i,j}} \tag{44}$$

where exponents $e_{i,j}$ are nonnegative integers and $\sum_{j=1}^{N} e_{i,j} = i$ for $i = 0, \ldots, K$. Therefore, the polynomial in N predictors with degree of K could have $L = C_K^{N+K}$ unique polynomial terms.

If we treat each unique term $\prod_{j=1}^{n} x_j^{e_{i,j}}$ in the polynomial as a predictor, the polynomial in (44) can be seen as a linear function of polynomial terms:

$$\phi(\boldsymbol{x})\boldsymbol{w} \tag{45}$$

where the regression coefficients $\boldsymbol{w}$ are

$$\boldsymbol{w} = [w_1, w_2, \ldots, w_L]^{\mathrm{T}}, \tag{46}$$

and terms in thepolynomial are

$$\phi(\boldsymbol{x}) = [\phi_1, \ldots, \phi_L]. \tag{47}$$

For example, a polynomial in three predictors $[x_1, x_2, x_3]$ with degree of 2 has 10 $(=C_2^{3+2})$ polynomial terms: 1, x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, x_1^2, x_2^2, x_3^2.

The conditional probability for the response can be written as

$$p(y|\boldsymbol{x}, \boldsymbol{w}, \sigma_\epsilon) = \frac{1}{\sqrt{2\pi\sigma_\epsilon^2}} \exp\left(-\frac{(y - \phi(\boldsymbol{x})\boldsymbol{w})^2}{2\sigma_\epsilon^2}\right). \tag{48}$$

In summary, we can fit a polynomial regression function with degree of K using the linear regression model. Specifically, we code the original N predictors into the C_K^{N+K} unique polynomial terms and treat them as predictors to fit a linear regression model. Choosing the degree K is not easy and a large K could introduce too many predictor variables to the regression model. To avoid over-fitting, we may consider adding the norm-1 regularizer to the objective function of least squared estimation, which may help drop unnecessary polynomial terms.

We fit a 3-degree polynomial function using the 8 principal components for predicting the concrete strength. There are $C_3^{8+3} = 165$ possible predictors in the regression model. We fit a Lasso regression (i.e., linear regression with a norm-1 regularizer) through fine tuning the hyperparameters of Elastic Net model. With the norm-1 regularization of the regression coefficients, 70 polynomial features are dropped. Actually, some polynomial terms have very small coefficients whose interval estimates contain zero. Those could be dropped too. We skip this step of model selection and let students themselves to finish it.

We continued with the model with 95 predictors. The summary is below. By looking at the following three plots, the concern about the residuals' lack of homogeneity became less severe as compared to that attained in the ordinary linear regression model.

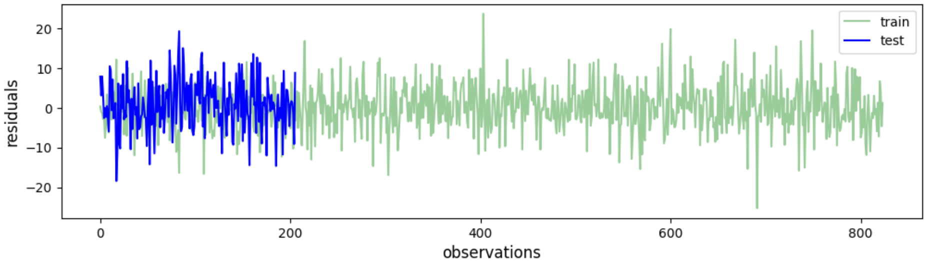


Figure 7: Residuals

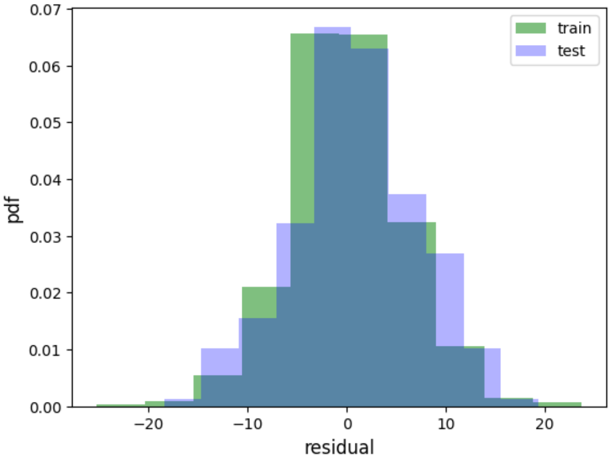


Figure 8: Residual Distribution

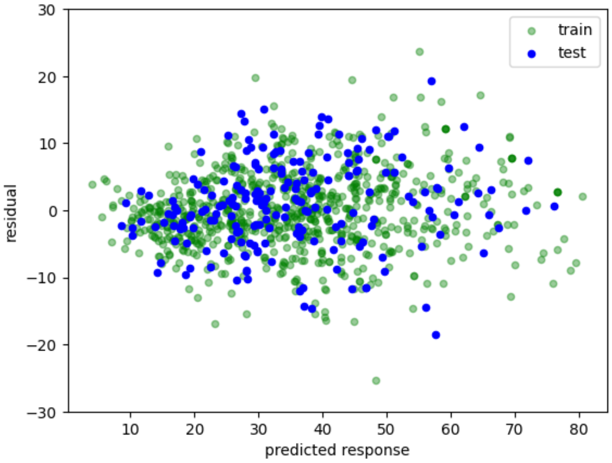


Figure 9: Residuals vs. predicted response

Table 1: Ordinary linear regression vs polynomial regression models on the test dataset

Model	MAE	RMSE	R^2
OLR	7.618	9.957	0.616
PLR	4.940	6.290	0.847

The R^2 on the training dataset using five-fold cross-validation is 0.779, and the R^2 on the test dataset is 0.847. The polynomial regression model fit data better than the ordinary linear regression model.

4 Multivariate Linear Regression

Consider a training dataset with M data points. Each data point is composed of N predictors and P responses. The training data in the form of matrix are:

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_M \end{bmatrix} = \begin{bmatrix} x_{1,1} & \dots & x_{1,N} \\ \vdots & \ddots & \vdots \\ x_{M,1} & \dots & x_{M,N} \end{bmatrix}, \tag{49}$$

and

$$\mathbf{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \dots, \boldsymbol{y}_P] = \begin{bmatrix} y_{1,1} & y_{1,2} & \dots & y_{1,P} \\ y_{2,1} & y_{2,2} & \dots & y_{2,P} \\ \vdots & \vdots & \ddots & \vdots \\ y_{M,1} & y_{M,2} & \dots & y_{M,P} \end{bmatrix}. \tag{50}$$

If one assumes a linear relationship exist between the predictors and the responses, a multivariate linear regression model can be developed:

$$\mathbf{Y} = \mathbf{Z}\boldsymbol{W} + \boldsymbol{\epsilon}, \tag{51}$$

where

$$\mathbf{Z} = [\mathbf{1}\ \mathbf{X}] = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \dots & x_{1,N} \\ 1 & x_{2,1} & x_{2,2} & \dots & x_{2,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{M,1} & x_{M,2} & \dots & x_{M,N} \end{bmatrix}, \tag{52}$$

$\boldsymbol{W}$ is the regression coefficient matrix:

$$\boldsymbol{W} = [\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_P] = \begin{bmatrix} w_{0,1} & w_{0,2} & \dots & w_{0,P} \\ w_{1,1} & w_{1,2} & \dots & w_{1,P} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N,1} & w_{N,2} & \dots & w_{N,P} \end{bmatrix}, \tag{53}$$

and $\boldsymbol{\epsilon}$ is the error matrix:

$$\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \dots, \epsilon_P] = \begin{bmatrix} \epsilon_{1,1} & \epsilon_{1,2} & \dots & \epsilon_{1,P} \\ \epsilon_{2,1} & \epsilon_{2,2} & \dots & \epsilon_{2,P} \\ \vdots & \vdots & \ddots & \vdots \\ \epsilon_{M,1} & \epsilon_{M,2} & \dots & \epsilon_{M,P} \end{bmatrix}. \tag{54}$$

with

$$\mathrm{E}(\epsilon) = \mathbf{0} \quad \text{and} \quad \mathrm{COV}(\epsilon_i, \epsilon_j) = \sigma_{i,j} \quad \text{for } i, j = 1, 2, \dots, p. \tag{55}$$

The least-squares estimation of $\boldsymbol{W}$ minimizes

$$\|\boldsymbol{\epsilon}\|_F^2 = \mathrm{Tr}[\boldsymbol{\epsilon}^{\mathrm{T}}\boldsymbol{\epsilon}] = \mathrm{Tr}[(\mathbf{Y} - \mathbf{Z}\boldsymbol{W})^{\mathrm{T}}(\mathbf{Y} - \mathbf{Z}\boldsymbol{W})] = \sum_{i=1}^{P}(\boldsymbol{w}_i^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\boldsymbol{w}_i - 2\boldsymbol{w}_i^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\boldsymbol{y}_i + \boldsymbol{y}_i^{\mathrm{T}}\boldsymbol{y}_i]. \tag{56}$$

which is the total sum of squared errors. By setting the partial derivative of (56) with respect to $\boldsymbol{w}_i$ equal to zero, the LSE for $\widehat{\boldsymbol{w}_i}$ is found to be $(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\boldsymbol{y}_i$. Thus,

$$\widehat{\boldsymbol{W}} = [\widehat{\boldsymbol{w}}_0, \dots, \widehat{\boldsymbol{w}}_P] = (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\mathbf{Y}. \tag{57}$$

The vector of fitted values is

$$\widehat{\mathbf{Y}} = \mathbf{Z}\widehat{\boldsymbol{W}} = \{\mathbf{Z}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}\}\mathbf{Y} = \mathbf{H}\mathbf{Y}, \tag{58}$$

where $\mathbf{H} = \mathbf{Z}(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\mathbf{Z}^{\mathrm{T}}$ is the hat matrix. $\widehat{\mathbf{Y}}$ estimates $\mathrm{E}(\mathbf{Y}|\mathbf{X})$.

The resulting residual matrix is

$$\widehat{\boldsymbol{\epsilon}} = \mathbf{Y} - \widehat{\mathbf{Y}} = (\mathbf{I} - \mathbf{H})\mathbf{Y}. \tag{59}$$

The estimator of the covariance matrix of responses is

$$\Sigma = \frac{\widehat{\boldsymbol{\epsilon}}^{\mathrm{T}}\widehat{\boldsymbol{\epsilon}}}{M - N - P} = \begin{bmatrix} \widehat{\sigma}_1^2 & \dots & \widehat{\sigma}_{1,P} \\ \vdots & \ddots & \vdots \\ \widehat{\sigma}_{P,1} & \dots & \widehat{\sigma}_P^2 \end{bmatrix}. \tag{60}$$

The point estimator $\widehat{\boldsymbol{W}}$ is random, and

$$\mathrm{Cov}(\widehat{\boldsymbol{w}}_j|\mathbf{Z}) = (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}\sigma_j^2. \tag{61}$$

Therefore, the standard error of $\widehat{\boldsymbol{w}}_j$ is

$$\mathrm{se}(\widehat{\boldsymbol{w}}_j) = \widehat{\sigma}_j \mathrm{Diag}\left(\sqrt{(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{-1}}\right) \tag{62}$$

for $j = 1, \dots, P$, and $\mathrm{Diag}(\cdot)$ is the diagonal of a square matrix saved as a column vector.

5 Binary Logistic Regression

5.1 The Data and Hypothesis Set

A training dataset contains m data points. Each data point consists of the observations of n predictors and a response. The training data in the matrix format are

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_m \end{bmatrix} = \begin{bmatrix} x_{1,1} & \dots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \dots & x_{m,n} \end{bmatrix}, \tag{63}$$

and,

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}. \tag{64}$$

Binary logistic regression models the conditional probability that a binary response is 1 given predictors and their coefficients. The binary logistic regression can take the following form:

$$p_i = p(y_i = 1|\boldsymbol{z}_i, \boldsymbol{w}) = \mathrm{sigmoid}(\boldsymbol{z}_i\boldsymbol{w}) = \frac{1}{1 + \exp(-\boldsymbol{z}_i\boldsymbol{w})}, \tag{65}$$

where $\boldsymbol{w}$ is the vector of regression coefficients

$$\boldsymbol{w}_{((n+1)\times 1)} = \begin{bmatrix} w_0 \\ \vdots \\ w_n \end{bmatrix}, \tag{66}$$

and $\boldsymbol{z}_i = [1\ \boldsymbol{x}_i]$. We further define $\boldsymbol{z}$ by appending a column of ones to the left of $\mathbf{X}$:

$$\boldsymbol{z}_{(m\times(n+1))} = [\mathbf{1}\ \mathbf{X}] = \begin{bmatrix} \boldsymbol{z}_1 \\ \vdots \\ \boldsymbol{z}_m \end{bmatrix} = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{m,1} & \dots & x_{m,n} \end{bmatrix}. \tag{67}$$

By learning from the training dataset, the regression coefficients $\boldsymbol{w}$ can be estimated, for example, using maximum likelihood estimation.

5.2 The Maximum Likelihood Estimation

5.2.1 The Log Likelihood

The likelihood on the training dataset is the joint probability of y_i's in the dataset:

$$p(\boldsymbol{y}|\boldsymbol{z},\boldsymbol{w}) = \prod_{i=1}^{m} p_i^{y_i}(1-p_i)^{(1-y_i)}, \tag{68}$$

where p_i is defined in (65).

The corresponding log likelihood is

$$\mathcal{L}(\boldsymbol{w}) = \sum_{i=1}^{m}[y_i \log p_i + (1-y_i)\log(1-p_i)]. \tag{69}$$

A *cross-entropy* loss function is defined accordingly:

$$L(\boldsymbol{w}) = -\frac{1}{m}\mathcal{L}(\boldsymbol{w}) = -\frac{1}{m}\sum_{i=1}^{m}[y_i \log p_i + (1-y_i)\log(1-p_i)]. \tag{70}$$

We choose the regression coefficients $\boldsymbol{w}$ to minimize the cross-entropy loss,

$$\widehat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} L(\boldsymbol{w}). \tag{71}$$

The cross-entropy loss function is strictly convex. The proof has been presented in detail in [2], and we skip the detail here.

5.2.2 MLE

Since the cross-entropy loss function is strictly convex, the global minimum exists and it is obtained by solving the following problem:

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}) = \mathbf{0}, \tag{72}$$

using a gradient method.

Let's define

$$a_i(\boldsymbol{w}) = \boldsymbol{z}_i\boldsymbol{w}, \tag{73}$$

which is called logit in the logistic regression.Then

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m}\frac{\partial L}{\partial p_i}\frac{\mathrm{d}p_i}{\mathrm{d}a_i}\frac{\mathrm{d}a_i}{\mathrm{d}\boldsymbol{w}} \tag{74}$$

With some simple derivation, we find that

$$\frac{\partial L}{\partial p_i} = -\left[\frac{y_i}{p_i} - \frac{1-y_i}{1-p_i}\right], \tag{75}$$

$$\frac{\mathrm{d}p_i}{\mathrm{d}a_i} = p_i(1-p_i), \tag{76}$$

$$\frac{\mathrm{d}a_i}{\mathrm{d}\boldsymbol{w}} = \boldsymbol{z}_i. \tag{77}$$

Therefore,

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}) = -\frac{1}{m}\sum_{i=1}^{m}(y_i - p_i)\boldsymbol{z}_i. \tag{78}$$

Let's define the regression residual as

$$\epsilon_i = y_i - p_i. \tag{79}$$

Equation (78) indicates the gradient is weighted average of $\boldsymbol{z}_i$'s and their weights are the regression residuals.

5.2.3 Stochastic Gradient Descent

we can use a gradient-based optimizer to search the global optimal solution $\widehat{\boldsymbol{w}}$. For example, stochastic gradient descent (SGD) is a simple gradient-based optimizer that uses single training data points or mini-batches (i.e., a small subset of the training dataset) to sequentially update regression coefficients:

$$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} - \eta\frac{\sum_{i\in B}(y_i - p_i)\boldsymbol{z}_i}{|B|}, \tag{80}$$

for $t = 0, 1, \ldots$. Here B is a mini-batch with one or more data points, η is a learning rate, and $p_i = p(y_i|\boldsymbol{z}_i, \boldsymbol{w}^{(t)})$ for $i \in B$.

5.3 An Example

Logistic regression is commonly used in classification problems. Let's consider a problem similar to the concrete example we studied. The response y is a binary variable indicates the class of a concrete slab after a year of use: "defective" vs. "non-defective". y takes the value 1 meaning defective, and zero meaning non-defective. We split the data by 80-20. That is, 80% of the data are used for training a model and 20% for testing the developed model.

The maximum likelihood estimates of regression coefficients are:

```
 Coefficient
1   -1.9312
x1  -0.9264
x2  -0.2590
x3  -0.9778
x4  -1.9127
x5  -0.9377
x6  -2.7294
x7  -0.8332
x8  -1.3011
```

The predicted class is

$$\widehat{y}_i = \begin{cases} 1 & \text{if } p_i > 0.5 \\ 0 & \text{if } p_i \leq 0.5. \end{cases} \tag{81}$$

Figure 10 is the confusion matrix. Among the 145 true ones, 131 are predicted correctly, and 14 are predicted mistakenly. Among the 61 true zeros, 52 are predicted correctly, and 9 are predictably mistakenly. The accuracy is $0.888 (= (131 + 52)/(131 + 14 + 9 + 52))$.

5.4 Other Link Functions for Binary Response

Besides sigmoid function, there are several other functions that can be used to model the mapping from a linear combination of predictors $\boldsymbol{z}_i\boldsymbol{w}$ to a binary response y_i. We skip the rest and interested students can read references to study further.

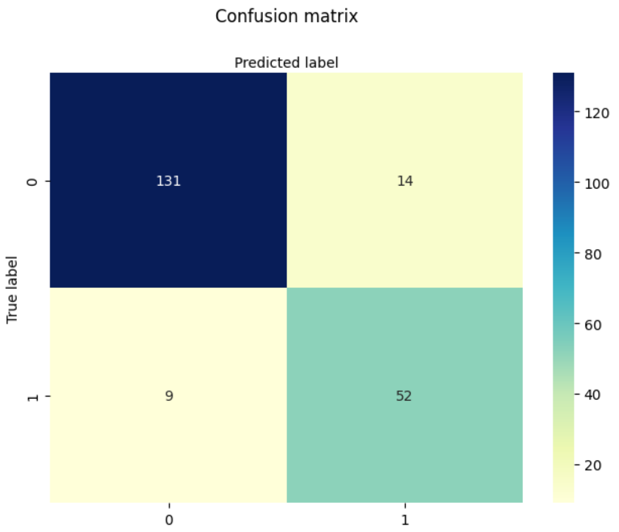


Figure 10: Confusion matrix

- Sigmoid (Logit): $\log p/(1-p)$
- Probit: $\Phi^{-1}(p)$
- Cauchit: $\arctan(\pi(p-0.5))$
- Negative log-log: $-\log(-\log(p))$
- Complementary log-log: $\log(-\log(1-p))$

6 Multinomial Logistic Regression

6.1 The Data and Hypothesis Set

Multinomial logistic regression is an extension of binary logistic regression to support multiclass classification problem.

A training sample contains m data points. Each data point consists of the observations of n predictors and a response with K nominal classes $\{c_1, \dots, c_K\}$. The training data in the matrix form are:

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_m \end{bmatrix} = \begin{bmatrix} x_{1,1} & \dots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \dots & x_{m,n} \end{bmatrix}, \tag{82}$$

and the one-hot encoding of the responses,

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_m \end{bmatrix} = \begin{bmatrix} y_{1,1} & \dots & y_{1,K} \\ \vdots & \ddots & \vdots \\ y_{m,1} & \dots & y_{m,K} \end{bmatrix}. \tag{83}$$

That is, $y_{i,k}$ is binary and $\sum_{k=1}^{K} y_{i,k} = 1$. For example, if a response is c_2, its one-hot vector representation is $[0, 1, 0, \dots]$.

We further define $\boldsymbol{z}$ by appending a column of ones to the left of $\mathbf{X}$,

$$\boldsymbol{z} = [\mathbf{1}\ \mathbf{X}] = \begin{bmatrix} \boldsymbol{z}_1 \\ \vdots \\ \boldsymbol{z}_m \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{x_1} \\ \vdots & \vdots \\ 1 & \boldsymbol{x}_m \end{bmatrix} = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{m,1} & \dots & x_{m,n} \end{bmatrix}, \tag{84}$$

let $\boldsymbol{w}$ be the regression coefficient matrix

$$\boldsymbol{w} = \left[\begin{array}{cc} \boldsymbol{w}_1 & \dots \boldsymbol{w}_K \end{array} \right] = \left[\begin{array}{ccc} w_{0,1} & \cdots & w_{0,K} \\ \vdots & \ddots & \vdots \\ w_{n,1} & \cdots & w_{n,K} \end{array} \right]. \tag{85}$$

and define the logit matrix $\mathbf{A}$

$$\mathbf{A} = \boldsymbol{z}\boldsymbol{w} = \left[\begin{array}{ccc} a_{1,1} & \cdots & a_{1,K} \\ \vdots & \ddots & \vdots \\ a_{m,1} & \cdots & a_{m,K} \end{array} \right]. \tag{86}$$

Define:

$$\mathbf{P} = \left[\begin{array}{c} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_m \end{array} \right] = \left[\begin{array}{ccc} p_{1,1} & \cdots & p_{1,K} \\ \vdots & \ddots & \vdots \\ p_{m,1} & \cdots & p_{m,K} \end{array} \right] \tag{87}$$

where $p_{i,k} = p(y_i = c_k)$ is the probability that y_i is class c_k, and $\sum_{k=1}^{K} p_{i,k} = 1$. That is, $\mathbf{p}_i$ is the predicted probability mess function for y_i.

$p_{i,k}$ can be modeled as a softmax function:

$$p_{i,k} = \frac{\exp(a_{i,k})}{\sum_{j=1}^{K} \exp(a_{i,j})}. \tag{88}$$

Because $\mathbf{P}$ is the prediction of $\mathbf{Y}$, the difference between them is the regression residual ϵ:

$$\epsilon = \left[\begin{array}{c} \epsilon_1 \\ \vdots \\ \epsilon_m \end{array} \right] = \left[\begin{array}{ccc} \epsilon_{1,1} & \cdots & \epsilon_{1,K} \\ \vdots & \ddots & \vdots \\ \epsilon_{m,1} & \cdots & \epsilon_{m,K} \end{array} \right]. \tag{89}$$

6.2 The Maximum Likelihood Estimation

$p_{i,k}$ can be written as:

$$p_{i,k} = \prod_{k=1}^{K} p_{i,k}^{y_{i,k}}. \tag{90}$$

Hence, the likelihood function, expressed as the joint conditional probability distribution of responses in the training sample, is

$$\prod_{i=1}^{m} \prod_{k=1}^{K} p_{i,k}^{y_{i,k}}. \tag{91}$$

Therefore, the log likelihood function is

$$\mathcal{L}(\mathbf{Y}|\boldsymbol{z}, \boldsymbol{w}) = \sum_{i=1}^{m} \sum_{k=1}^{K} y_{i,k} \log p_{i,k}. \tag{92}$$

Accordingly, we define the cross-entropy loss function

$$L(\mathbf{Y}|\boldsymbol{z}, \boldsymbol{w}) = -\frac{1}{m} \mathcal{L}(\mathbf{Y}|\boldsymbol{z}, \boldsymbol{w}) = -\frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} y_{i,k} \log p_{i,k}. \tag{93}$$

The optimal regression coefficient $\widehat{\boldsymbol{w}}$ is obtained by minimizing the cross-entropy loss function

$$\widehat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} L(\mathbf{Y}|\mathbf{X}, \boldsymbol{w}). \tag{94}$$

$\widehat{\boldsymbol{w}}$ is attained at

$$\nabla L_{\boldsymbol{w}}(\boldsymbol{w}) = \mathbf{0}. \tag{95}$$

Because

$$\frac{\partial L}{\mathrm{d}p_{i,k}} = -\frac{y_{i,k}}{p_{i,k}}, \tag{96}$$

$$\frac{\partial p_{i,k}}{a_{i,j}} = p_{i,k}(\delta_{k,j} - p_{i,j}), \tag{97}$$

where

$$\delta_{k,j} = \begin{cases} 1, & \text{if } j = k, \\ 0, & \text{if } j \neq k, \end{cases} \tag{98}$$

and

$$\frac{\mathrm{d}a_{i,j}}{\mathrm{d}\boldsymbol{w}_j} = \boldsymbol{z}_i, \tag{99}$$

one can find that

$$\begin{aligned} \nabla_{\boldsymbol{w}_j} L(\boldsymbol{w}) &= \frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K}\frac{\partial L}{\mathrm{d}p_{i,k}}\frac{\partial p_{i,k}}{a_{i,j}}\frac{\mathrm{d}a_{i,j}}{\mathrm{d}\boldsymbol{w}_j} \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(\sum_{k=1}^{K}\frac{y_{i,k}}{p_{i,k}}p_{i,k}(\delta_{k,j} - p_{i,j})\boldsymbol{z}_i\right) \\ &= -\frac{1}{m}\sum_{i=1}^{m}\left(\sum_{k=1}^{K}\delta_{k,j}y_{i,k}\boldsymbol{z}_i - \left(\sum_{k=1}^{K}y_{i,k}\right)p_{i,j}\boldsymbol{z}_i\right) \\ &= -\frac{1}{m}\sum_{i=1}^{m}((y_{i,j} - p_{i,j})\boldsymbol{z}_i), \end{aligned} \tag{100}$$

for $j = 1, \ldots, K$. The gradient matrix as a function of regression coefficients is:

$$\nabla_{\boldsymbol{w}} L(\boldsymbol{w}) = -\frac{1}{m}\boldsymbol{z}^T(\mathbf{Y} - \mathbf{P}). \tag{101}$$

Just like in binary logistic regression, the gradient matrix for multinomial logistic regression is also a weighted average of predictors and the weights are regression residuals. A gradient-based optimizer can find the global optimal $\widehat{\boldsymbol{w}}$ based on the gradient matrix.

6.3 Example

Let's use the optical recognition of handwritten digits (MNIST) dataset as an example. We split the dataset into training data (80%) and test data (20%). We fit a multinomial logistic regression model with elastic net regularization. The testing accuracy is 0.981. The confusion matrix is presented in Figure 11.

What if we extract a fewer number of features and use them as the predictors to fit a multinomial logistic regression?

7 Ordinal Logistic Regression

In Section 6, classes of the response are nominal. In this section, we consider the case that classes of the response are ordinal. The ordering may be inherent in the categories of choice. For example, a five-point Likert response in a survey could be “Very Dissatisfied", “Dissatisfied", “Neural", “Satisfied", “Very Satisfied". A continuous variable could be split into segments with each being a category. For example, the flood risk in an area is “low risk", “ medium risk", or “high risk" based on a quantify measurement.

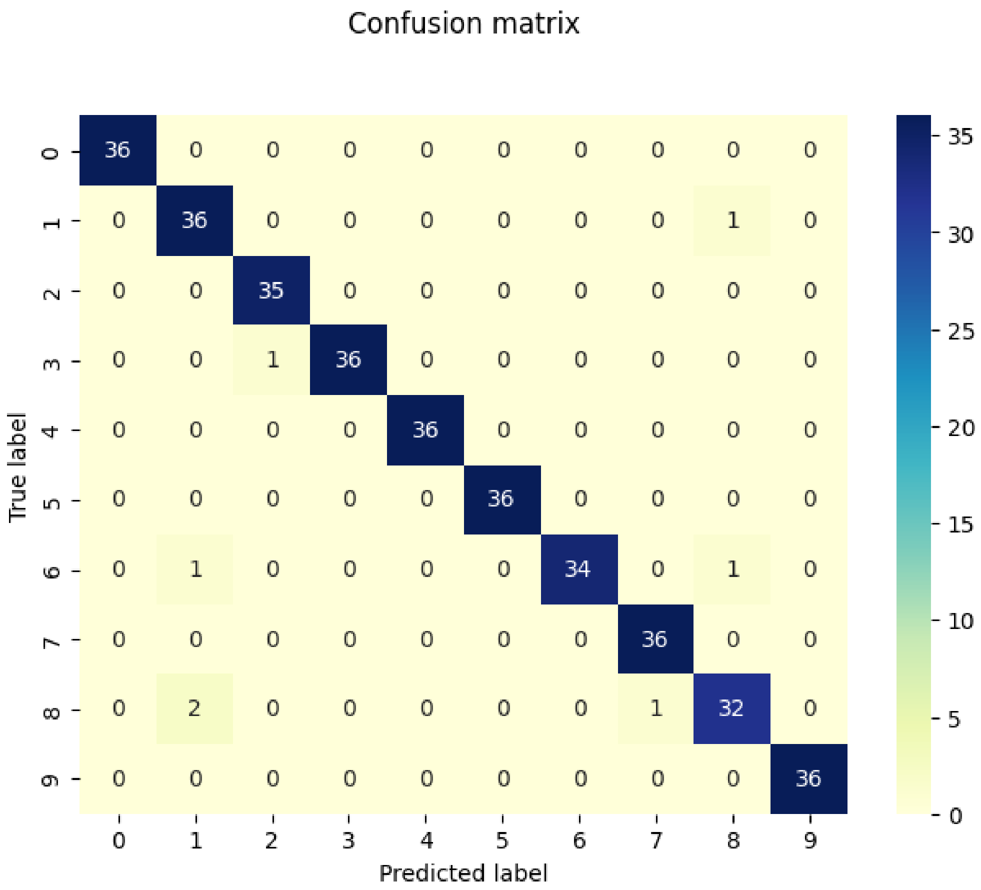


Figure 11: Confusion matrix of the MNIST test result from the multinomial logistic regression

7.1 The Data and Hypothesis Set

A training dataset with m data points that each consists of the observations of n predictors and a response with ordinal classes. We denote $1 < \cdots < K$ to indicate the ordering of classes. The training data in the matrix form are:

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_m \end{bmatrix} = \begin{bmatrix} x_{1,1} & \dots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \dots & x_{m,n} \end{bmatrix}, \tag{102}$$

and

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}. \tag{103}$$

where $y_i \in \{c_1, \dots, c_K\}$.

We can model a series of $K - 1$ cumulative probability distributions (CDF), for example in the form of sigmoid functions

$$F_{i,k} = p(y_i \leq k | \boldsymbol{x}_i, \boldsymbol{w}, \beta_k) = \frac{1}{1 + \exp\left(\beta_k - \boldsymbol{x}_i \boldsymbol{w}\right)} \tag{104}$$

where $\boldsymbol{w}$ is the vector of logistic regression coefficients for predictors:

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} \tag{105}$$

and β_k is the bias for the kth function and

$$\beta_1 > \cdots > \beta_{K-1}. \tag{106}$$

In this approach, the $(K-1)$ CDFs at any fixed value $\boldsymbol{x}\boldsymbol{w}$ split the range $[0,1] \cap \mathbb{R}$ into k mutully exclusive and collective inclusive segments. One chooses $\boldsymbol{w}$ and β to make the resulting probability mass distribution at any value of $\boldsymbol{x}\boldsymbol{w}$,

$$p_{i,k} = p(y_i = k|\boldsymbol{x}_i, \boldsymbol{w}, \beta_k) = F_{i,k} - F_{i,k-1} \tag{107}$$

with $F_{i,0}$=0 and $F_{i,K} = 1$, best fit the response's mass distribution there.

The $K-1$ binary logistic regression models share the same regression coefficients $\boldsymbol{w}$ but they have their specific biases. This reduces the complexity of the models.

We will fit the $K-1$ regression models simultanously, for example in the approach of maximum likelihood estimation. The likelihood is

$$\prod_{i=1}^{m} \prod_{k=1}^{K} p_{i,k}^{y_{i,k}} \tag{108}$$

and thus the log likelihood is

$$\mathcal{L}(\boldsymbol{w}, \beta) = \sum_{i=1}^{m} \sum_{k=1}^{K} y_{i,k} \log p_{i,k} \tag{109}$$

where $y_{i,k}$ is the one-hot encoding of the response y_i. That is, $y_{i,k} = 1$ if $y_i = k$ and $\sum_{k=1}^{K} y_{i,k} = 1$. Here we skip the details. Students whose research requires a thorough understanding of the ordinal logistic regression model can read references such as [1].

8 Closing Remark

In this learning module, we primarily introduced several popularly used regression models falling in the category of *Generalized Linear Models* (GLM). A GLM maps a linear combination of predictors $\boldsymbol{x}_i\boldsymbol{w}$ to the expected value of response $\mathrm{E}[y_i|\boldsymbol{x}_i]$ through a link function $l(\boldsymbol{x}_i\boldsymbol{w}) = \mathrm{E}[y_i|\boldsymbol{x}, \boldsymbol{w}]$.

We considered various distributions of the response in regression: the response in the multiple linear regression follows a univariate Gaussian distribution, a multivariate Gaussian distribution in multivariate linear regression, the Bernoulli distribution in the binary logistic regression, and categorical distribution in the multinomial logistic regression. There are a few more distributions of responses that this learning module did not cover, such as responses following binomial distribution or Poisson distribution. Regression models for those can be developed similarly.

References

[1] Alan Agresti. *Categorical Data Analysis*, volume 792. John Wiley & Sons, 2012.

[2] Kevin P. Murphy. *Probabilistic Machine Learning: An introduction.* MIT Press, 2022.

[3] Kevin P. Murphy. *Probabilistic Machine Learning: Advanced Topics.* MIT Press, 2023.