Contents

1 Linear Regression ................................................................. 2
  1.1 Univariate regression ..................................................... 3
  1.2 Multiple regression ...................................................... 9
  1.3 Ridge Regression: Regularization .................................... 15
  1.4 Kernel Regression ...................................................... 17
  1.5 Further Reading .......................................................... 19
  1.6 Exercises ................................................................. 20
Given a set of attributes or variables \( X_1, X_2, \ldots, X_d \), called the predictor, explanatory, or independent variables, and given an attribute of interest \( Y \), called the response or dependent variable, the aim of regression is to model the effect of the predictor variables on the response variable. That is, the goal is to learn a regression function \( f \), such that

\[
Y = f(X_1, X_2, \ldots, X_d) + \epsilon = f(X) + \epsilon
\]

where \( X = (X_1, X_2, \ldots, X_d)^T \) is the multi-variate random variable, and \( \epsilon = Y - f(X) \) is a random error term (also called residual error), that is assumed to be independent of \( X \). In other words, \( Y \) is comprised of two components, one dependent on the observed predictor attributes, and the other, coming from the error term, independent of the predictor attributes. The error term encapsulates inherent uncertainty in \( Y \), as well as, possibly the effect of unobserved, hidden or latent variables.

In linear regression the function \( f \) is assumed to be linear in \( X \), that is

\[
f(X) = b + w_1X_1 + w_2X_2 + \cdots + w_dX_d = b + \sum_{i=1}^{d} w_iX_i = b + w^T X \tag{1.1}
\]

Here, \( b \) is the unknown bias term, \( w_i \) is the unknown regression coefficient or weight for attribute \( X_i \), \( w = (w_1, w_2, \cdots, w_d)^T \) is the \( d \)-dimensional weight vector. Observe that \( f \) specifies a hyperplane in \( \mathbb{R}^d \), with \( w \) being the weight vector that is normal or orthogonal to the hyperplane, and \( b \) represents the intercept or offset along \( Y \). We can see that \( f \) is completely specified by the \( d + 1 \) parameters comprising \( b \) and \( w_i \), for \( i = 1, \cdots, d \).

Given a training dataset \( D = \{x_i, y_i\}_{i=1, \ldots, n} \) comprising \( n \) points \( x_i \) in a \( d \)-dimensional space, and given the corresponding response values \( y_i \), the goal of linear regression is to estimate the \( d + 1 \) unknown parameters \( b \) and \( w_i \) from the observed data. Given these estimates, we can predict the response for any given input or test point.
1.1 Univariate regression

\( \mathbf{x} = (x_1, x_2, \cdots, x_d)^T \), as follows

\[
\hat{y} = b + w_1 x_1 + \cdots + w_d x_d = b + \mathbf{w}^T \mathbf{x}
\]  

(1.2)

Due to the error term, the predicted value \( \hat{y} \) will not in general match the true (unknown) response \( y \) for the given input \( \mathbf{x} \). This is even true for the training data. The error in the estimate is given as

\[
\epsilon = y - \hat{y} = y - b - \mathbf{w}^T \mathbf{x}
\]  

(1.3)

In fact, a common approach to predicting the bias and regression coefficients is to use the method of least squares. That is, given the training data \( \mathbf{D} = \{ (\mathbf{x}_i, y_i) \}_{i=1}^n \), we seek values \( b \) and \( \mathbf{w} \), so as to minimize the sum of squared residual errors

\[
\text{SSE} = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - b - \mathbf{w}^T \mathbf{x}_i)^2
\]  

(1.4)

We will see below how to estimate the unknown parameters, by first considering the case of a single predictor variable, and then considering the general case of multiple predictors.

1.1 UNIVARIATE REGRESSION

Let us first consider the case where the input data \( \mathbf{D} \) comprises a single predictor attribute \( X \), along with the response variable \( Y \), i.e., \( \mathbf{D} = \{ (x_i, y_i) \}_{i=1}^n \). Since \( f \) is linear, we have

\[
\hat{y}_i = f(x_i) = b + w \cdot x_i
\]  

(1.5)

We can observe that we seek the straight line \( f \) with slope \( w \) and intercept \( b \) that best fits the data. Define the error in the predicted value (also called fitted value) and true value of the response variable as

\[
\epsilon_i = y_i - \hat{y}_i
\]  

(1.6)

Note that \( |\epsilon_i| \) denotes the vertical distance between the fitted and true response. The best fitting line minimizes the sum of squared errors

\[
\arg\min_{b, w} \text{SSE} = \sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - b - w \cdot x_i)^2
\]
To solve the objective, we differentiate the above expression with respect to \( b \) and set the result to 0, to obtain

\[
\frac{\partial}{\partial b} \text{SSE} = 2 \sum_{i=1}^{n} (y_i - b - w \cdot x_i) = 0
\]

\[\Rightarrow \sum_{i=1}^{n} b = \sum_{i=1}^{n} y_i - w \sum_{i=1}^{n} x_i \]

\[\Rightarrow b = \frac{1}{n} \sum_{i=1}^{n} y_i - \frac{1}{n} \sum_{i=1}^{n} x_i \]

\[\Rightarrow b = \mu_Y - w \cdot \mu_X \quad (1.7)\]

where \( \mu_Y \) is the sample mean for the response and \( \mu_X \) is the sample mean for the predictor attribute. Similarly, differentiating w.r.t. \( w \), we obtain

\[
\frac{\partial}{\partial w} \text{SSE} = -2 \sum_{i=1}^{n} x_i (y_i - b - w \cdot x_i) = 0
\]

\[\Rightarrow \sum_{i=1}^{n} x_i \cdot y_i - b \sum_{i=1}^{n} x_i - w \sum_{i=1}^{n} x_i^2 = 0 \]

substituting \( b \) from above, we have

\[\Rightarrow \sum_{i=1}^{n} x_i \cdot y_i - \mu_Y \sum_{i=1}^{n} x_i + w \cdot \mu_X \sum_{i=1}^{n} x_i - w \sum_{i=1}^{n} x_i^2 = 0 \]

\[\Rightarrow w \left( \sum_{i=1}^{n} x_i^2 - n \mu_X^2 \right) = \sum_{i=1}^{n} x_i \cdot y_i - n \cdot \mu_X \cdot \mu_Y \]

\[\Rightarrow w = \frac{\sum_{i=1}^{n} x_i \cdot y_i - n \cdot \mu_X \cdot \mu_Y}{\sum_{i=1}^{n} x_i^2 - n \cdot \mu_X^2} \quad (1.8)\]

The regression coefficient \( w \) can also be written as

\[w = \frac{\sum_{i=1}^{n} (x_i - \mu_X)(y_i - \mu_Y)}{\sum_{i=1}^{n} (x_i - \mu_X)^2} = \frac{\sigma_{XY}}{\sigma_X^2} \quad (1.9)\]

where \( \sigma_X^2 \) is the variance of \( X \) and \( \sigma_{XY} \) is the covariance between \( X \) and \( Y \). Noting the fact that the correlation between \( X \) and \( Y \) is given as \( \rho_{XY} = \frac{\sigma_{XY}}{\sigma_X \sigma_Y} \), we can also write \( w \) as

\[w = \rho_{XY} \frac{\sigma_Y}{\sigma_X} \]

One of the properties of the estimated values for \( b \) and \( w \) is that the sum of errors is zero, since

\[\sum_{i=1}^{n} \epsilon_i = \sum_{i=1}^{n} (y_i - b - w \cdot x_i) \]

\[= \sum_{i=1}^{n} (y_i - \mu_Y + w \cdot \mu_X - w \cdot x_i) \]
Figure 1.1: Scatterplot: petal length ($X$) versus petal width ($Y$). The solid circle (black) shows the mean point. The residual error is shown for two sample points $x_9$ and $x_{35}$.

\[
\begin{align*}
\sum_{i=1}^{n} y_i - n \cdot \mu_Y + w \cdot \left( n \mu_X - \sum_{i=1}^{n} x_i \right) \\
= n \cdot \mu_Y - n \cdot \mu_Y + w( n \cdot \mu_X - n \cdot \mu_X) = 0
\end{align*}
\]

In other words, the sum of the errors above and below the line cancel each other out.

Also, we can observe that the fitted line must pass through the mean value of $Y$ and $X$; plugging in the optimal values of $b$ and $w$ into the regression equation Eq. (1.5), we have

\[
\hat{y}_i = b + w \cdot x_i \\
= \mu_Y - w \cdot \mu_X + w \cdot x_i \\
= \mu_Y + w(x_i - \mu_X)
\]

That is, when $x_i = \mu_X$, we have $\hat{y}_i = \mu_Y$. Thus, the point $(\mu_X, \mu_Y)$ lies on the regression line.

**Example 1.1 (Univariate Regression).** Figure 1.1 shows the scatterplot between the two attributes petal length ($X$; the predictor variable) and petal width ($Y$; the response variable) in the Iris dataset. There are a total of $n = 150$ data points. The
mean values for these two variables are
\[
\mu_X = \frac{1}{150} \sum_{i=1}^{150} x_i = \frac{563.8}{150} = 3.759
\]
\[
\mu_Y = \frac{1}{150} \sum_{i=1}^{150} y_i = \frac{179.8}{150} = 1.199
\]

The variance and covariance is given as
\[
\sigma_X^2 = \frac{1}{150} \sum_{i=1}^{150} (x_i - \mu_X)^2 = 3.092
\]
\[
\sigma_Y^2 = \frac{1}{150} \sum_{i=1}^{150} (y_i - \mu_Y)^2 = 0.579
\]
\[
\sigma_{XY} = \frac{1}{150} \sum_{i=1}^{150} (x_i - \mu_X) \cdot (y_i - \mu_Y) = 1.288
\]

Assuming a linear relationship between the response and predictor variables, we use Eq. (1.7) and Eq. (1.8) to obtain the slope and intercept terms as follows
\[
w = \frac{\sigma_{XY}}{\sigma_X^2} = \frac{1.288}{3.092} = 0.416
\]
\[
b = \mu_Y - w \cdot \mu_X = 1.199 - 0.416 \cdot 3.759 = -0.365
\]

Thus, we have
\[
\hat{Y} = -0.365 + 0.416 \cdot X
\]

Fig. 1.1 plots the best-fitting or regression line; we can observe that the mean point \((\mu_X, \mu_Y) = (3.759, 1.199)\) lies on the line. The figure also shows the residual error \(\epsilon_9\) for the point \(x_9\), and the error \(\epsilon_{35}\) for \(x_{35}\).

Finally, we can compute the SSE value (see Eq. (1.4)) as follows:
\[
\text{SSE} = \sum_{i=1}^{150} \epsilon_i^2 = \sum_{i=1}^{150} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{150} (y_i + 0.365 - 0.416 \cdot x_i)^2 = 1797.96
\]

1.1.1 Geometry of Univariate Regression

We now turn to the attribute-centric view, which provides important geometric insight for univariate regression. First recall that we are given \(n\) equations in two unknowns, namely \(\hat{y}_i = b + w \cdot x_i\), for \(i = 1, \cdots, n\). Let \(X = (x_1, x_2, \cdots, x_n)^T\) be the \(n\)-dimensional vector denoting the training data sample, \(Y = (y_1, y_2, \cdots, y_n)^T\) the \(n\)-dimensional vector for the dependent variable, and \(\hat{Y} = (\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_n)^T\) the vector of predicted values, then we
1.1 Univariate regression

Figure 1.2.: Geometry of univariate regression: non-orthogonal basis. All the vectors conceptually lie in the \(n\)-dimensional space spanned by the \(n\) data points.

can express the \(n\) equations in a single equation as follows:

\[
\hat{Y} = b \cdot 1 + w \cdot X
\]  

(1.11)

where \(1 \in \mathbb{R}^n\) is the \(n\)-dimensional vector of all ones. This equation indicates that fitted or predicted vector \(\hat{Y}\) is a linear combination of \(1\) and \(X\), i.e., it must lie in the column space spanned by \(1\) and \(X\), given as \(\text{span}(\{1, X\})\). On the other hand, the response vector \(Y\) will not usually lie in the same column space. The geometry of the problem, shown in Fig. 1.2, makes it clear that \(\hat{Y}\) is the orthogonal projection of \(Y\) onto the vector subspace spanned by \(1\) and \(X\). In fact the error vector \(\epsilon = (\epsilon_1, \epsilon_2, \cdots, \epsilon_n)^T\) captures the deviation between the response and predicted values

\[
\epsilon = Y - \hat{Y}
\]

The error vector is orthogonal to the subspace spanned by \(1\) and \(X\), and the square of the length of the error vector equals the sum of squared residual errors in Eq. (1.4), since

\[
\|\epsilon\|^2 = \|Y - \hat{Y}\|^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2
\]

At this point it is worth noting that even though \(1\) and \(X\) are linearly independent and form a basis for the column space, they need not be orthogonal (see Fig. 1.2). We can create an orthogonal basis by decomposing \(X\) into a component along \(1\) and a component orthogonal to \(1\) as shown in Fig. 1.3. Recall that the projection of a vector \(b\) onto the vector \(a\) is given as

\[
\text{proj}_a(b) = \left( \frac{b^T a}{a^T a} \right) \cdot a
\]

Now the projection of \(X\) onto \(1\) is given as

\[
\text{proj}_1(X) = \left( \frac{X^T 1}{1^T 1} \right) \cdot 1 = \left( \frac{\sum_{i=1}^{n} x_i}{n} \right) \cdot 1 = \mu_X \cdot 1
\]
Thus, we can rewrite $X$ as

$$X = \mu_X \cdot \mathbf{1} + (X - \mu_X \cdot \mathbf{1}) = \mu_X \cdot \mathbf{1} + Z$$

where $Z = X - \mu_X \cdot \mathbf{1}$ is simply the centered attribute vector, obtained by subtracting the mean $\mu_X$ from all points.

The two vectors $\mathbf{1}$ and $Z$ form an orthogonal basis for the subspace. We can thus obtain the predicted vector $\hat{Y}$ by projecting $Y$ onto $\mathbf{1}$ and $Z$, and summing up these two components, as shown in Fig. 1.4. That is

$$\hat{Y} = \text{proj}_{\mathbf{1}}(Y) \cdot \mathbf{1} + \text{proj}_Z(Y) \cdot Z$$

$$= \left( \frac{Y^T \mathbf{1}}{\mathbf{1}^T \mathbf{1}} \right) \mathbf{1} + \left( \frac{Y^T Z}{Z^T Z} \right) Z,$$

substituting $w = \frac{Y^T Z}{Z^T Z}$ we have

$$= \mu_Y \cdot \mathbf{1} + w \cdot Z,$$

this decomposition is shown in Figure 1.4

$$= \mu_Y \cdot \mathbf{1} + w(X - \mu_X \cdot \mathbf{1})$$

$$= (\mu_Y - w \cdot \mu_X) \cdot \mathbf{1} + w \cdot X$$
1.2 Multiple regression

\[ y = b \cdot 1 + w \cdot X \]

where the bias term \( b = \mu_Y - w \cdot \mu_X \) matches Eq. (1.7), and the weight \( w \) also matches Eq. (1.8), since

\[
w = \frac{Y^T Z}{Z^T Z} = \frac{Y^T (X - \mu_X \cdot 1)}{\|X - \mu_X \cdot 1\|^2} = \frac{\sum_{i=1}^n y_i x_i - n \cdot \mu_X \cdot \mu_Y}{\sum_{i=1}^n x_i^2 - n \cdot \mu_X^2}.
\]

In other words, the slope \( w \) of the best fitting line is the length of the projection of \( Y \) in terms of \( X \). It is also proportional to the cosine of the angle between \( Y \) and the centered vector \( Z = X - \mu_X \cdot 1 \), since

\[
w = \frac{Y^T Z}{Z^T Z} = \frac{\|Y\| \cdot \|Z\| \cos \theta}{\|Z\|^2} = \frac{\|Y\|}{\|Z\|} \cos \theta
\]

**Example 1.2 (Geometry of Regression).** Let us consider the regression of petal length \((X)\) on the petal width \(Y\) for the Iris dataset, with \( n = 150 \). First, we center \( X \) by subtracting the mean \( \mu_X = 3.759 \). Next, we compute the projections of \( Y \) onto \( 1 \) and \( Z \), to obtain the weight and bias terms, as follows:

\[
\mu_Y = \text{proj}_1(Y) = \frac{Y^T 1}{1^T 1} = 179.8 \quad \text{and} \quad w = 1.199
\]

\[
\quad w = \text{proj}_Z(Y) = \frac{Y^T Z}{Z^T Z} = \frac{193.2}{463.9} = 0.416
\]

Thus, the bias term \( b \) is given as

\[ b = \mu_Y - w \cdot \mu_X = 1.199 - 0.416 \cdot 3.759 = 1.199 - 1.564 = -0.365 \]

We can see that these values match those in Example 1.1. Finally, we can compute the SSE value (see Eq. (1.4)) as the squared length of the error vector

\[ \text{SSE} = \|\epsilon\|^2 = \|Y - \hat{Y}\|^2 = (Y - \hat{Y})^T (Y - \hat{Y}) = 1797.96 \]

1.2 MULTIPLE REGRESSION

We now consider the more general case called multiple regression \(^2\) when there are multiple independent attributes \( X_1, X_2, \ldots, X_d \) and a single dependent attribute \( Y \). The training data sample \( D = \{x_i, y_i\}_{i=1}^n \) comprises \( n \) points \( x_i = (x_{i1}, x_{i2}, \ldots, x_{id})^T \) in a \( d \)-dimensional space, along with the corresponding true response value \( y_i \). The predicted response value for input \( x_i \) is given as

\[ \hat{y}_i = b + w_1 x_{i1} + w_2 x_{i2} + \cdots + w_d x_{id} = b + w^T x_i \quad (1.12) \]

\(^2\) We follow the usual terminology and reserve the term multivariate regression when there are multiple dependent attributes \( Y_1, Y_2, \ldots, Y_q \) and multiple independent attributes \( X_1, X_2, \ldots, X_d \).
where $w = (w_1, w_2, \ldots, w_d)$ is the weight vector comprising the regression coefficients or weights $w_j$ along each attribute $X_j$. The equation above defines a hyperplane with offset $b$ and normal vector $w$.

Instead of dealing with the bias $b$ separately from the weights $w_i$, we can introduce a new “constant” valued attribute $X_0$ whose value is always fixed at 1. Thus, each input point $(x_{i1}, x_{i2}, \ldots, x_{id})^T \in \mathbb{R}^d$ is mapped to the point $(x_{i0}, x_{i1}, x_{i2}, \ldots, x_{id})^T \in \mathbb{R}^{d+1}$, where $x_{i0} = 1$. Setting the weight $w_0 = b$, the predicted response value for the augmented $(d + 1)$ dimensional point $x_i$ can be written as

$$\hat{y}_i = w_0 x_{i0} + w_1 x_{i1} + w_2 x_{i2} + \cdots + w_d x_{id} = w^T x_i$$

(1.13)

where $w = (w_0, w_1, \ldots, w_d)^T \in \mathbb{R}^{d+1}$. Since there are $n$ points, in fact we have $n$ such equations, one per point, and there are $(d + 1)$ unknowns, namely the elements of the weight vector $w$. We can compactly write all these $n$ equations in a single matrix equation, given as

$$\hat{Y} = Dw$$

(1.14)

where $D \in \mathbb{R}^{n \times (d+1)}$ is the $(d + 1)$ dimensional augmented data matrix, which includes the constant attribute $X_0$, and $\hat{Y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)^T$ is the vector of predicted responses.

The multiple regression task can now be stated as finding the best fitting hyperplane defined by the weight vector $w$ that minimizes the sum of squared errors

$$\arg\min_w SSE = \sum_{i=1}^{n} \epsilon_i^2 = \|\epsilon\|^2 = \|Y - \hat{Y}\|^2$$

$$= (Y - \hat{Y})^T (Y - \hat{Y}) = Y^T Y - 2Y^T \hat{Y} + \hat{Y}^T \hat{Y}$$

$$= Y^T Y - 2Y^T (Dw) + (Dw)^T (Dw)$$

$$= Y^T Y - 2w^T (D^T Y) + w^T (D^T D) w$$

where we substituted $\hat{Y} = Dw$ from Eq. (1.14), and we used the fact that $Y^T (Dw) = (Dw)^T Y$.

To solve the objective, we differentiate the above expression with respect to $w$ and set the result to 0, to obtain

$$\frac{\partial}{\partial w} SSE = -2D^T Y + 2(D^T D) w = 0$$

$$\implies (D^T D) w = D^T Y$$

$$\implies w = (D^T D)^{-1} D^T Y$$

(1.15)

Substituting the optimal value of $w$ into Eq. (1.14), we get

$$\hat{Y} = Dw = D(D^T D)^{-1} D^T Y = HY$$

where $H = D(D^T D)^{-1} D^T$ is called the hat matrix, since it puts the “hat” on $Y$. 
1.2 Multiple regression

Figure 1.5: Multiple Regression: sepal length ($X_1$) and petal length ($X_2$) with dependent attribute petal width ($Y$). The vertical bars show the residual error for the points.

Example 1.3 (Multiple Regression). Fig. 1.5 shows the multiple regression of sepal length ($X_1$) and petal length ($X_2$) on the dependent attribute petal width ($Y$) for the Iris dataset with $n = 150$ points. We first add an extra attribute $X_0 = \mathbf{1}_{150}$, which is a vector of all ones in $\mathbb{R}^{150}$. The augmented dataset $D \in \mathbb{R}^{150 \times 3}$ comprises $n = 150$ points along three attributes $X_0, X_1$ and $X_2$.

Next, we compute the uncentered $3 \times 3$ covariance matrix $D^T D$ and its inverse

$$D^T D = \begin{pmatrix} 150.0 & 876.50 & 563.80 \\ 876.5 & 5223.85 & 3484.25 \\ 563.8 & 3484.25 & 2583.00 \end{pmatrix}, \quad (D^T D)^{-1} = \begin{pmatrix} 0.793 & -0.176 & 0.064 \\ -0.176 & 0.041 & -0.017 \\ 0.064 & -0.017 & 0.009 \end{pmatrix}$$

We also compute $D^T Y$, given as

$$D^T Y = \begin{pmatrix} 179.80 \\ 1127.65 \\ 868.97 \end{pmatrix}$$

The augmented weight vector $w$, is then given as

$$w = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \end{pmatrix} = (D^T D)^{-1} \cdot (D^T Y) = \begin{pmatrix} -0.014 \\ -0.082 \\ 0.45 \end{pmatrix}$$

The bias term is therefore $b = w_0 = -0.014$, and the fitted model is

$$\hat{Y} = -0.014 - 0.082 \cdot X_1 + 0.45 \cdot X_2$$

Fig. 1.5 shows the fitted hyperplane. It also shows the residual error for each point. The white colored points have positive residuals (i.e., $\epsilon_i > 0$ or $\hat{y}_i > y_i$), whereas the gray points have negative residual values (i.e., $\epsilon_i < 0$ or $\hat{y}_i < y_i$).
1.2.1 Geometry of Multiple Regression

Let \( \mathbf{D} \) be the augmented data matrix comprising the \( d \) independent attributes \( X_i \), along with the new constant attribute \( X_0 = 1 \) \( \in \mathbb{R}^e \), and let \( \mathbf{w} = (w_0, w_1, \ldots, w_d) \mathbb{R}^{(d+1)} \) be the weight vector which incorporates the bias term \( b = w_0 \). Recall that the predicted vector of responses is given as

\[
\hat{Y} = b \cdot 1 + w_1 \cdot X_1 + w_2 \cdot X_2 + \cdots + w_d \cdot X_d = \sum_{i=0}^{d} w_i \cdot X_i = \mathbf{Dw}
\]

The equation makes it clear that the predicted vector must lie in the column space of the augmented data matrix \( \mathbf{D} \), denoted \( \text{col}(\mathbf{D}) \), i.e., is must be a linear combination of the attribute vectors \( X_i, i = 0, \ldots, d \). Further, to minimize the error in prediction, \( \hat{Y} \) must be the orthogonal projection of \( Y \) onto the vector subspace spanned by the independent attributes

\[
\hat{Y} = \text{proj}_{\text{col}(\mathbf{D})}(Y)
\]

The error vector is \( \mathbf{e} = Y - \hat{Y} \) is thus orthogonal to the subspace \( \text{col}(\mathbf{D}) \), which means that it is orthogonal to each attribute vector \( X_i \). That is,

\[
X_i^T \mathbf{e} = 0
\]

\[
\implies X_i^T (Y - \hat{Y}) = 0
\]

\[
\implies X_i^T \hat{Y} = X_i^T Y
\]

\[
\implies X_i^T (\mathbf{Dw}) = X_i^T Y^T
\]

\[
\implies w_0 \cdot X_i^T X_0 + w_1 \cdot X_i^T X_1 + \cdots + w_d \cdot X_i^T X_d = X_i^T Y
\]

We thus have \( (d+1) \) equations (called the normal equations) in \( (d+1) \) unknowns, namely the regression coefficients or weights \( w_i \). The solution to these simultaneous equations yields the weight vector \( \mathbf{w} = (w_0, w_1, \ldots, w_d)^T \). The \( (d+1) \) equations are

\[
\begin{align*}
& w_0 \cdot X_0^T X_0 + w_1 \cdot X_0^T X_1 + \cdots + w_d \cdot X_0^T X_d = X_0^T Y \\
& w_0 \cdot X_1^T X_0 + w_1 \cdot X_1^T X_1 + \cdots + w_d \cdot X_1^T X_d = X_1^T Y \\
& \quad \vdots \\
& w_0 \cdot X_d^T X_0 + w_1 \cdot X_d^T X_1 + \cdots + w_d \cdot X_d^T X_d = X_d^T Y
\end{align*}
\]

which can be written compactly in matrix notation to solve for \( \mathbf{w} \) as follows

\[
\begin{pmatrix}
X_0^T X_0 & X_0^T X_1 & \cdots & X_0^T X_d \\
X_1^T X_0 & X_1^T X_1 & \cdots & X_1^T X_d \\
\vdots & \vdots & \cdots & \vdots \\
X_d^T X_0 & X_d^T X_1 & \cdots & X_d^T X_d
\end{pmatrix}
\begin{pmatrix}
w_0 \\
w_1 \\
\vdots \\
w_d
\end{pmatrix}
= \mathbf{D}^T Y
\]

\[
(\mathbf{D}^T \mathbf{D}) \mathbf{w} = \mathbf{D}^T Y
\]

\[
\mathbf{w} = (\mathbf{D}^T \mathbf{D})^{-1} (\mathbf{D}^T Y)
\]
1.2 Multiple regression

This matches the expression given in Eq. (1.15).

More insight can be obtained by noting that the attribute vectors comprising the column space of \( D \) are not necessarily orthogonal, even if we assume they are linearly independent. To obtain the projected vector \( \hat{Y} = \text{proj}_{\text{col}(D)}(Y) \), we first need to obtain an orthogonal basis for \( \text{col}(D) \). We can do this via Gram-Schmidt orthogonalization, given as

\[
\begin{align*}
U_0 &= X_0 \\
U_1 &= X_1 - p_{10} \cdot U_0 \\
U_2 &= X_2 - p_{20} \cdot U_0 - p_{21} \cdot U_1 \\
\vdots &= \vdots \\
U_d &= X_d - p_{d0} \cdot U_0 - p_{d1} \cdot U_1 - \cdots - p_{d,d-1} \cdot U_{d-1}
\end{align*}
\]

where \( p_{ji} = \text{proj}_{U_i}(X_j) = \frac{X_j^T U_i}{\|U_i\|^2} \) denotes the projection of attribute \( X_j \) onto the basis vector \( U_i \). Essentially, to obtain \( U_j \), we subtract from each vector \( X_j \) its projections along all previous basis vectors \( U_0, U_1, \ldots, U_{j-1} \).

The Gram-Schmidt method results in the so-called QR-factorization of the data matrix, namely \( D = QR \), where by construction \( Q \) is an \( n \times (d+1) \) matrix with orthogonal columns

\[
Q = \begin{pmatrix}
U_0 \\
U_1 \\
\vdots \\
U_d
\end{pmatrix}
\]

and \( R \) is the \((d+1) \times (d+1)\) upper-triangular matrix

\[
R = \begin{pmatrix}
1 & p_{10} & p_{20} & p_{30} & \cdots & p_{d0} \\
0 & 1 & p_{21} & p_{31} & \cdots & p_{d1} \\
0 & 0 & 1 & p_{32} & \cdots & p_{d2} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & 1 & p_{d-1,d} \\
0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
\]

So that, in the column view the QR-factorization of the augmented data matrix is given as:

\[
\begin{pmatrix}
X_0 \\
X_1 \\
\vdots \\
X_d
\end{pmatrix}_{D} = \begin{pmatrix}
U_0 \\
U_1 \\
\vdots \\
U_d
\end{pmatrix}_{Q} \cdot \begin{pmatrix}
1 & p_{10} & p_{20} & p_{30} & \cdots & p_{d0} \\
0 & 1 & p_{21} & p_{31} & \cdots & p_{d1} \\
0 & 0 & 1 & p_{32} & \cdots & p_{d2} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & 0 & 1 & p_{d-1,d} \\
0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}_{R}
\]

Since the new basis vectors \( U_0, U_1, \ldots, U_d \) form an orthogonal basis for the column space of \( D \), we can obtain the predicted response vector as a sum of the projections of
Y along each new basis vector, given as
\[
\hat{Y} = \text{proj}_{U_0}(Y) \cdot U_0 + \text{proj}_{U_1}(Y) \cdot U_1 + \cdots + \text{proj}_{U_d}(Y) \cdot U_d
\]

It can be shown that the bias term \( b = w_0 \) is given as follows:
\[
w_0 = \text{proj}_{U_0}(Y) - \sum_{j=1}^{d} w_j \cdot \text{proj}_{U_0}(X_j) = \mu_Y - \sum_{j=1}^{d} w_j \cdot \mu_{X_j}
\]
\[
= \mu_Y - w_1 \cdot \mu_{X_1} - w_2 \cdot \mu_{X_2} - \cdots - w_d \cdot \mu_{X_d}
\]
(1.16)

Above, we used the fact that the projection of any attribute vector onto \( U_0 = X_0 = 1 \) yields the mean value of that attribute, since
\[
\text{proj}_{U_0}(Y) = \frac{Y^T 1}{1^T 1} = \frac{1}{n} \sum_{i=1}^{n} y_i = \mu_Y
\]
\[
\text{proj}_{U_0}(X_j) = \frac{X_j^T 1}{1^T 1} = \frac{1}{n} \sum_{i=1}^{n} x_{ji} = \mu_{X_j}
\]

Example 1.4 (Multiple Regression). Consider the multiple regression of \( \text{sepal length} (X_1) \) and \( \text{petal length} (X_2) \) on the dependent attribute \( \text{petal width} (Y) \) for the Iris dataset with \( n = 150 \) points, as shown in Fig. 1.5. The augmented dataset \( D \in \mathbb{R}^{150 \times 3} \) comprises \( n = 150 \) points along three attributes \( X_0, X_1 \) and \( X_2 \), where \( X_0 = 1 \). Running the Gram-Schmidt orthogonalization results in the following QR-factorization:
\[
\begin{pmatrix}
X_0 & X_1 & X_2 \\
1 & 1 & 1
\end{pmatrix}
= \begin{pmatrix}
U_0 & U_1 & U_2 \\
1 & 1 & 1
\end{pmatrix}
\cdot
\begin{pmatrix}
1 & 5.843 & 3.759 \\
0 & 1 & 1.858 \\
0 & 0 & 1
\end{pmatrix}
\]

The inverse of \( R \) is also upper-triangular, and is given as
\[
R^{-1} = \begin{pmatrix}
1 & -5.843 & 7.095 \\
0 & 1 & -1.858 \\
0 & 0 & 1
\end{pmatrix}
\]

We can now write \( Q \) in terms of the original attributes via \( Q = D \cdot R^{-1} \) to obtain
\[
U_0 = X_0
\]
\[
U_1 = -5.843 \cdot X_0 + X_1
\]
\[
U_2 = 7.095 \cdot X_0 - 1.858 \cdot X_1 + X_2
\]

The projection of the response vector onto each one the new basis vectors is as follows:
\[
\text{proj}_{U_0}(Y) = 1.199 \quad \text{proj}_{U_1}(Y) = 0.754 \quad \text{proj}_{U_2}(Y) = 0.45
\]
Finally, the fitted response vector is given as:

\[
\hat{Y} = \text{proj}_{U_0}(Y) \cdot U_0 + \text{proj}_{U_1}(Y) \cdot U_1 + \text{proj}_{U_2}(Y) \cdot U_2
\]

\[
= 1.199 \cdot X_0 + 0.754(-5.843 \cdot X_0 + X_1) + 0.45(7.095 \cdot X_0 - 1.858 \cdot X_1 + X_2)
\]

\[
= (1.199 - 4.406 + 3.193) \cdot X_0 + (0.754 - 0.836) \cdot X_1 + 0.45 \cdot X_2
\]

\[
= -0.014 \cdot X_0 - 0.082 \cdot X_1 + 0.45 \cdot X_2
\]

We can see that the results match those in Example 1.3.

1.3 RIDGE REGRESSION: REGULARIZATION

We have seen that for linear regression, the predicted response vector \( \hat{Y} \) lies in the span of the column vectors comprising the augmented data matrix \( D \). However, often the data can have noise and uncertainty, and it may not be desirable to fit to that data exactly. Instead, we can use regularization to constrain the solution vector \( w \) to have a small norm. In other words, instead of trying to simply minimize the squared error \( \| Y - \hat{Y} \|^2 \), we add a regularization term involving the squared norm of the weight vector \( \| w \|^2 \) as follows:

\[
\min_w J(w) = \| Y - \hat{Y} \|^2 + \alpha \cdot \| w \|^2
\]

(1.17)

Here \( \alpha \geq 0 \) is a regularization constant that trade-offs the norm of the weight vector versus the squared error. When \( \alpha = 0 \), there is no regularization, and as \( \alpha \) increases there is more emphasis on minimizing the regression coefficients.

The solve the new regularized objective we differentiate Eq. (1.17) with respect to \( w \) and set the results to 0 to obtain

\[
\frac{\partial}{\partial w} J(w) = \frac{\partial}{\partial w} \left[ Y^T Y - 2w^T (D^T Y) + w^T (D^T D) w + \alpha \cdot w^T w \right] = 0
\]

\[
\implies -2D^T Y + 2(D^T D) w + 2\alpha \cdot w = 0
\]

(1.18)

\[
\implies (D^T D + \alpha \cdot I) w = D^T Y
\]

(1.19)

where \( I \in \mathbb{R}^{d \times d} \) is the identity matrix. The matrix \( (D^T D + \alpha \cdot I) \) is always invertible for \( \alpha > 0 \) even if \( D^T D \) is not invertible. The reason is because if \( \lambda_i \) is an eigenvalue of \( D^T D \), then \( \lambda_i + \alpha \) is an eigenvalue of \( (D^T D + \alpha \cdot I) \). Since \( D^T D \) is positive semi-definite it has non-negative eigenvalues. Thus, even if an eigenvalue of \( D^T D \) is zero, \( \lambda_i = 0 \), the corresponding eigenvalue of \( (D^T D + \alpha \cdot I) \) is \( \alpha > 0 \).

Regularized regression is also called ridge regression since we add a “ridge” along the main diagonal of the \( D^T D \) matrix, i.e., the optimal solution depends on the regularized matrix \( (D^T D + \alpha \cdot I) \). Another advantage of regularization is that if we choose a small positive \( \alpha \) we are always guaranteed a solution.
Example 1.5 (Ridge Regression). Figure 1.1 shows the scatterplot between the two attributes petal length ($X$; the predictor variable) and petal width ($Y$; the response variable) in the Iris dataset. There are a total of $n = 150$ data points. The uncentered covariance matrix is given as

$$D^TD = \begin{pmatrix} 150.0 & 563.8 \\ 563.8 & 2583.0 \end{pmatrix}$$

Using Eq. (1.19) we obtain different lines of best fit for different value of the regularization constant $\alpha$:

- $\alpha = 0 : \hat{Y} = -0.365 + 0.416 \cdot X$ \quad $\|w\|^2 = \|{-0.365, 0.416}^T\|^2 = 0.308$
- $\alpha = 10 : \hat{Y} = -0.244 + 0.388 \cdot X$ \quad $\|w\|^2 = \|{-0.244, 0.388}^T\|^2 = 0.210$
- $\alpha = 100 : \hat{Y} = -0.021 + 0.328 \cdot X$ \quad $\|w\|^2 = \|{-0.021, 0.328}^T\|^2 = 0.108$

Figure Fig. 1.6 shows these best-fitting or regression lines. We can see that as $\alpha$ increases there is more emphasis on minimizing the norm of $w$. 
We now consider how to generalize linear regression to the non-linear case, i.e., finding a non-linear fit to the data to minimize the squared error, along with regularization. For this we will adopt the kernel trick, i.e., we will show that all relevant operations can be carried out via the kernel matrix $K$ in input space.

Let $\phi$ correspond to a mapping from input space to the feature space, so that each point $\phi(x_i)$ in feature space is a mapping for the input point $x_i$. Let $D_\phi = \{\phi(x_i)^T\}_{i=1,\ldots,n}$ denote the augmented dataset in feature space, i.e., each transformed point appears as a row of $D_\phi$. Let $Y = \{y_i\}_{i=1,\ldots,n}$ denote the true response vector. We model the predicted response as

$$\hat{Y} = D_\phi w$$

where $w$ is the coefficient vector in feature space.

For regularized regression, we have to solve for the following objective in feature space:

$$\min_w J(w) = \|Y - \hat{Y}\|^2 + \alpha \cdot \|w\|^2$$

(1.21)

where $\alpha \geq 0$ is a regularization constant. Expanding the above objective we obtain

$$Y^T Y - 2w^T (D_\phi^T Y) + w^T (D_\phi^T D_\phi) w + \alpha \cdot w^T w$$

Following a similar derivation to Eq. (1.18), we have

$$\alpha \cdot w = D_\phi^T Y - (D_\phi^T D_\phi) w$$

$$w = D_\phi^T \left( \frac{1}{\alpha} (Y - D_\phi w) \right)$$

$$w = D_\phi^T c = \sum_{i=1}^n c_i \phi(x_i)$$

(1.22)

where $c = (c_1, c_2, \ldots, c_n)^T = \frac{1}{\alpha} (Y - D_\phi w)$ is the coefficient vector. The above equation indicates that the weight vector $w$ is a linear combination of the feature points.

Rearranging the terms in the expression for $c$, we have

$$c = \frac{1}{\alpha} (Y - D_\phi w)$$

$$\alpha \cdot c = Y - D_\phi w$$

Now, plugging in the form of $w$ from Eq. (1.22) we get

$$\alpha \cdot c = Y - D_\phi (D_\phi^T c)$$

$$(D_\phi D_\phi^T + \alpha \cdot I) c = Y$$

$$c = (D_\phi D_\phi^T + \alpha \cdot I)^{-1} Y$$

$$c = (K + \alpha \cdot I)^{-1} Y$$

(1.23)
where \( I \in \mathbb{R}^{n \times n} \) is the \( n \times n \) identity matrix, and \( D_\phi D_\phi^T \) is in fact the kernel matrix \( K \), since

\[
D_\phi D_\phi^T = \{ \phi(x_i)^T \phi(x_j) \}_{i,j=1,2,\ldots,n} = \{ K(x_i, x_j) \}_{i,j=1,2,\ldots,n} = K
\]

Putting it all together, we can substitute Eq. (1.23) and Eq. (1.22) into Eq. (1.20) to obtain the expression for the predicted response

\[
\hat{Y} = D_\phi w = D_\phi D_\phi c = (D_\phi D_\phi^T)(K + \alpha \cdot I)^{-1} Y = K(K + \alpha \cdot I)^{-1} Y \tag{1.24}
\]

where \( K(K + \alpha \cdot I)^{-1} \) is the kernel hat matrix.

In general, for any new point \( x_j \) we can use Eq. (1.22) to predict the response, as follows:

\[
\hat{y}_j = \phi(x_j)^T w = \sum_{i=1}^{n} c_i \cdot \phi(x_j)^T \phi(x_i) = \sum_{i=1}^{n} c_i \cdot K(x_j, x_i)
\]

That is, we compute the vector of kernel values of \( x_j \) with respect to all of the data points in \( D \), and take its dot product with the coefficient vector \( c \) to obtain the predicted response.

Figure 1.7: Kernel regression on Nonlinear Iris dataset
Example 1.6. Consider the nonlinear Iris dataset shown in Fig. 1.7, obtained via a nonlinear transformation applied on the centered Iris data. In particular, the sepal length ($A_1$) and sepal width attributes ($A_2$) were transformed as follows:

\[
X = A_2 \\
Y = 0.2A_1^2 + A_2^2 + 0.1A_1A_2
\]

We treat $Y$ as the response variable and $X$ is the independent attribute. The points show a clear quadratic (nonlinear) relationship between the two variables.

We find the lines of best fit using both a linear and an inhomogeneous quadratic kernel, with regularization constant $\alpha = 0.1$. The linear kernel yields the following fit

\[
Y = 0.167 \cdot X
\]

On the other hand, using the quadratic (inhomogeneous) kernel yields the fit

\[
Y = -0.171 + 0.026 \cdot X + 0.914 \cdot X^2
\]

The linear (in gray) and quadratic (in black) fit are both shown in Fig. 1.7. The SSE error $\|Y - \hat{Y}\|^2$ is 13.82 for the linear kernel and 4.33 for the quadratic kernel. It is clear that the quadratic kernel (as expected) gives a much better fit to the data.

1.5 Further Reading

For a geometrical approach to multivariate statics see Wickens (2014) and Saville and Wood (2012).


1.6 Exercises

Table 1.1. Data for Q1

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

**Q1.** Consider the data in Table 1.1, with \( Y \) as the response variable and \( X \) as the independent variable. Answer the following questions:

(a) Compute the predicted response vector \( \hat{Y} \) for least square regression using the geometric approach.

(b) Based on the geometric approach extract the value of the bias and slope, and give the equation of the best fitting regression line.

**Q2.** Show that Eq. (1.9) holds, that is

\[
\begin{align*}
    w & = \frac{\sum_{i=1}^{n} x_i \cdot y_i - n \cdot \mu_X \cdot \mu_Y}{\sum_{i=1}^{n} x_i^2 - n \cdot \mu_X^2} = \frac{\sum_{i=1}^{n} (x_i - \mu_X) (y_i - \mu_Y)}{\sum_{i=1}^{n} (x_i - \mu_X)^2}
\end{align*}
\]

**Q3.** Derive an expression for the bias term \( b \) and the weights \( w = (w_1, w_2, \ldots, w_d)^T \) in multiple regression using Eq. (1.12), without adding the augmented column.

**Q4.** Show that the bias term in multiple regression, as shown in Eq. (1.16), is given as

\[
\begin{align*}
    w_0 & = \mu_Y - w_1 \cdot \mu_X_1 - w_2 \cdot \mu_X_2 - \cdots - w_d \cdot \mu_X_d
\end{align*}
\]

**Q5.** Show that if \( \lambda_i \) is an eigenvalue of \( D^T D \) corresponding to the eigenvector \( u_i \), then \( \lambda_i + \alpha \) is an eigenvalue of \( (D^T D + \alpha \cdot I) \) for the same eigenvector \( u_i \).