

# Elements of statistical learning: Chapter 3

July 30, 2020

## 1 Linear models

- Sampling properties of  $\hat{\beta}$
- Gauss-Markov Theorem

## 2 Multiple regression

- From simple univariate to multiple regressions
- The Gram-Schmidt algorithm

## 3 Shrinkage methods

- Ridge regression
- SVD of ridge regression

# LS estimator

Let  $\mathbf{X}$  be an  $N \times (p + 1)$  matrix of explanatory variables and  $\mathbf{y}$  an  $N \times 1$  vector of outputs. Then we know the LS estimator  $\hat{\beta}$

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

[see lecture slides "ESL1" for recap and proof].

## The "hat" matrix

As such for the fitted linear model

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}\hat{\beta} \\ &= \underbrace{\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'}_H \mathbf{y} \\ &= H\mathbf{y}\end{aligned}$$

where  $H$  is commonly referred to as the hat matrix.

# $H$ the projection matrix

Let us denote the column vectors of  $\mathbf{X}$  by  $\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_p$  with  $\mathbf{x}_0 \equiv 1$ .

- These vectors span a subspace of  $\mathbb{R}^N$ , also referred to as a column vector of  $\mathbf{X}$ .
- We minimize  $RSS(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|^2$  by choosing  $\hat{\beta}$ , so that the residual vector  $\mathbf{y} - \hat{\mathbf{y}}$  is orthogonal to this subspace.
- the hat matrix  $H$  computes the orthogonal projection, and hence it is also known as the projection matrix.

# Variance-covariance matrix

## Assumptions

- 1 Observations  $y_i$  are uncorrelated have constant variance  $\sigma^2$
- 2  $x_i$  are fixed (i.e. non-stochastic)

$$\begin{aligned}\text{var}(\hat{\beta}) &= \text{var} [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}] \\&= \text{var} [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'(\mathbf{X}\beta + \epsilon)] \\&= \text{var} [(\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{X})\beta + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\epsilon] \\&= \text{var} [(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\epsilon] \\&= \mathbb{E} \{ (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\epsilon[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\epsilon]'\} \\&= \mathbb{E} \{ (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\epsilon\epsilon'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \} \\&= \mathbb{E} \{ (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{X})\epsilon\epsilon'(\mathbf{X}'\mathbf{X})^{-1} \}\end{aligned}$$

Note that  $\epsilon$  is the error term and has zero mean and also remember that  $\mathbf{X}$  is fixed, and thus

$$\mathbb{E}[aZ] = a\mathbb{E}[Z]$$

where  $Z$  is a random variable and  $a$  is a constant. Therefore,

$$\begin{aligned} \text{var}(\hat{\beta}) &= \mathbb{E} \{ \epsilon \epsilon' (\mathbf{X}' \mathbf{X})^{-1} \} \\ &= (\mathbf{X}' \mathbf{X})^{-1} E \{ \epsilon \epsilon' \} \\ &= (\mathbf{X}' \mathbf{X})^{-1} \sigma^2 \end{aligned}$$

where  $\sigma^2$  can be calculated by

$$\sigma^2 = \frac{1}{N - p - 1} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

thus, assuming the errors are further Gaussian

$$\hat{\beta} \sim N(\beta, (\mathbf{X}' \mathbf{X})^{-1} \sigma^2)$$

# Gauss-Markov Theorem

Least squares estimator of parameter  $\beta$  has the smallest variance among all linear unbiased estimators. Why is the LS estimator unbiased?

Proof.

$$\begin{aligned}\hat{\beta} &= \mathbb{E}[\hat{\beta}] \\ &= \mathbb{E}[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}] \\ &= \mathbb{E}[(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'(\mathbf{X}\beta + \epsilon)] \\ &= \mathbb{E}[\beta + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\epsilon] \\ &= \beta + (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbb{E}[\epsilon] \\ &= \beta\end{aligned}$$



# From simple univariate to multiple regressions

Suppose first we have a univariate model with no intercept

$$Y_i = X_i\beta + \varepsilon_i$$

The least squares estimates and residuals are

$$\hat{\beta} = \frac{\sum_{i=1}^N x_i y_i}{\sum_{i=1}^N x_i^2}$$

with residuals

$$r_i = y_i - x_i \hat{\beta}$$

which in vector notation can be expressed as

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^N x_i y_i = \mathbf{x}' \mathbf{y}$$

which is the inner product between  $\mathbf{x}$  and  $\mathbf{y}$ .



Thus, the OLS estimator  $\hat{\beta}$  can be expressed as follows

$$\hat{\beta} = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\langle \mathbf{x}, \mathbf{x} \rangle},$$

Suppose now that we have  $p$  inputs  $\mathbf{x}_1, \dots, \mathbf{x}_p$ , which are the columns of the matrix  $\mathbf{X}$  and are orthogonal, such that  $\langle \mathbf{x}_j, \mathbf{x}_k \rangle = 0$  for all  $i \neq j$ . When the inputs are orthogonal, the multiple least squares estimates  $\hat{\beta}_j$  are equal to the univariate estimates - i.e.

$$\hat{\beta}_j = \frac{\langle \mathbf{x}_j, \mathbf{y} \rangle}{\langle \mathbf{x}_j, \mathbf{x}_j \rangle}$$

In other words, the inputs are orthogonal and have no impact on each other's parameters estimates in the model.

Consider the case of an intercept and a single input  $\mathbf{x}$ , then the least squares coefficient of  $\mathbf{x}$  has the form

$$\hat{\beta}_1 = \frac{\langle \mathbf{x} - \bar{\mathbf{x}}1, \mathbf{y} \rangle}{\langle \mathbf{x} - \bar{\mathbf{x}}1, \mathbf{x} - \bar{\mathbf{x}}1 \rangle}$$

The steps of the algorithm can be seen as follows

- 1 Regress  $\mathbf{x}$  on 1 to obtain  $\bar{\mathbf{x}}1$
- 2 Obtain the residuals  $\mathbf{z} = \mathbf{x} - \bar{\mathbf{x}}1$
- 3 Regress  $\mathbf{y}$  on  $\mathbf{z}$  to obtain the coefficient  $\hat{\beta}_1$

$$\hat{\beta}_1 = \frac{\langle \mathbf{z}, \mathbf{y} \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle}$$

Step 1 orthogonalizes  $\mathbf{x}$  with respect to  $\mathbf{x}_0 = 1$ .

# The Gram-Schmidt algorithm

The idea is similar in the presence of more predictors. In the case of two predictors and an intercept, say,  $\mathbf{x}_0 = 1, \mathbf{x}_1, \mathbf{x}_2$ .

- 1 First regress  $\mathbf{x}_1$  on  $\mathbf{x}_0 = 1$  and obtain the residual vector  $\mathbf{z}_1 = \mathbf{x}_1 - \bar{x}_1$
- 2 Then regress  $\mathbf{x}_2$  on  $\mathbf{x}_0 = 1$  and  $\mathbf{z}_1$  to produce the coefficients  $\hat{\gamma}_1$  and obtain the residual vector  $\mathbf{z}_2 = \mathbf{x}_2 - \bar{x}_2 - \hat{\gamma}_1 \mathbf{z}_1$
- 3 Regress  $\mathbf{y}$  on the residual  $\mathbf{z}_p$  to get the estimate  $\hat{\beta}_p$ .

This algorithm can alternatively be expressed in matrix format. In other words, the second step can be written as follows

$$\mathbf{X} = \mathbf{Z}\mathbf{\Gamma}$$

(**Note:** For a review of QR decomposition and its application to Gram-Schmidt algorithm, [click here](#))

with

$$\mathbf{Z} = \begin{bmatrix} 1 & z_{11} & \cdots & z_{1p} \\ 1 & z_{21} & \cdots & z_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & z_{N1} & \cdots & z_{Np} \end{bmatrix} \quad \text{and} \quad \mathbf{\Gamma} = \begin{bmatrix} \bar{x} & \bar{x} & \bar{x} & \cdots & \bar{x} \\ & \hat{\gamma}_1 & \hat{\gamma}_1 & \cdots & \hat{\gamma}_1 \\ & & \hat{\gamma}_2 & \cdots & \hat{\gamma}_2 \\ & & & \ddots & \vdots \\ 0 & & & & \hat{\gamma}_p \end{bmatrix}$$

we then introduce a diagonal matrix  $\mathbf{D}$  with  $j^{\text{th}}$  diagonal entry  $D_{jj} = \|z_j\|$ ,  
- i.e.

$$\mathbf{D} = \begin{bmatrix} \|z_0\| & 0 & \cdots & 0 \\ 0 & \|z_1\| & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \|z_p\| \end{bmatrix}$$

and we express the matrix  $\mathbf{X}$  as follows

$$\mathbf{X} = \mathbf{Z}\mathbf{D}^{-1}\mathbf{D}\mathbf{\Gamma}$$

Noting that  $\mathbf{Q} = \mathbf{ZD}^{-1}$  is  $N \times (p+1)$  with orthonormal columns, - i.e.  $\mathbf{Q}'\mathbf{Q} = \mathbf{I}$  and  $\mathbf{D}\mathbf{\Gamma}$  is a  $(p+1) \times (p+1)$  upper triangular matrix. Thus, the least squares estimator is given by

$$\begin{aligned}\hat{\beta} &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} &= [(\mathbf{QR})'(\mathbf{QR})]^{-1}\mathbf{Q}'\mathbf{R}\mathbf{y} \\ & &= [\mathbf{R}'\mathbf{Q}'\mathbf{Q}\mathbf{R}]^{-1}\mathbf{Q}'\mathbf{R}\mathbf{y} \\ & &= [\mathbf{R}'\mathbf{I}\mathbf{R}]^{-1}\mathbf{Q}'\mathbf{R}\mathbf{y} \\ & &= \mathbf{R}^{-1}\mathbf{Q}'\mathbf{y}\end{aligned}$$

similarly,

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}\hat{\beta} \\ &= (\mathbf{QR})(\mathbf{R}^{-1}\mathbf{Q}'\mathbf{y}) \\ &= \mathbf{Q}\mathbf{Q}'\mathbf{y}\end{aligned}$$

## Shrinkage methods

Shrinkage methods shrink the regression coefficients by imposing a penalty on their size. The most notable shrinkage methods are the *Lasso* and *Ridge regressions*.

The ridge coefficients minimize a penalized residual sum of squares:

$$\hat{\beta}_{Ridge} = \arg \min_{\beta} \left\{ \sum_{i=1}^N (y_i - \beta_0 - \sum_{j=1}^p x_{ij} \beta_j)^2 + \lambda \sum_{j=1}^p \beta_j^2 \right\} \quad (1)$$

Here  $\lambda \geq 0$  is a complexity parameter that controls the amount of shrinkage.

For reasons outlined in age 65 of the book, let us centre the input  $x_{ij}$  by  $x_{ij} - \bar{x}_j$  and we estimate  $\beta_0$  by  $\bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$ . Thus, the remaining coefficients gets estimated by a ridge regression without an intercept, where  $\mathbf{X}$  has  $p$  instead of  $(p + 1)$  columns. Henceforth, relationship (1) can instead be expressed in the matrix format as follows

$$\text{RSS}(\lambda) = (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta'\beta$$

Thus, the solution to ridge regression can easily be seen

$$\begin{aligned} \hat{\beta}_{\text{Ridge}} &= \arg \min_{\beta} \{(\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta'\beta\} \\ &= \arg \min_{\beta} \{(\mathbf{y}' - \beta'\mathbf{X}')(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta'\beta\} \\ &= \arg \min_{\beta} \{(\mathbf{y}'\mathbf{y} - \mathbf{y}'\mathbf{X}\beta - \beta'\mathbf{X}'\mathbf{y} + \beta'\mathbf{X}'\mathbf{X}\beta) + \lambda\beta'\beta\} \\ &= -\mathbf{X}'\mathbf{y} - \mathbf{X}'\mathbf{y} + 2\beta\mathbf{X}'\mathbf{X} + 2\lambda\beta = 0 \\ &= \beta(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I}) = \mathbf{X}'\mathbf{y} \\ &= (\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{y} \end{aligned}$$

# SVD of ridge regression

The SVD of the  $N \times p$  matrix  $\mathbf{X}$  has the form

$$\mathbf{X} = \mathbf{U}_{N \times p} \mathbf{D}_{p \times p} \mathbf{V}'_{p \times p}$$

where  $\mathbf{U}$  and  $\mathbf{V}$  are orthogonal - i.e.

$$\mathbf{U}'\mathbf{U} = \mathbf{I}, \quad \mathbf{V}'\mathbf{V} = \mathbf{I}$$

(For a quick review of SVD, [click here](#).)

Using the singular value decomposition, we can write the least squares fitted vector as

$$\begin{aligned} \hat{\mathbf{y}} = \mathbf{X} \hat{\boldsymbol{\beta}}_{ls} &= \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} \\ &= \mathbf{UDV}[(\mathbf{UDV})'\mathbf{UDV}]^{-1}(\mathbf{UDV})'\mathbf{y} \\ &= \mathbf{UDV}[\mathbf{V}'\mathbf{D}'\mathbf{U}'\mathbf{UDV}]^{-1}\mathbf{V}'\mathbf{D}'\mathbf{U}'\mathbf{y} \\ &= \mathbf{UDV}[\mathbf{D}'\mathbf{D}]^{-1}\mathbf{V}'\mathbf{D}'\mathbf{U}'\mathbf{y} \\ &= \mathbf{UU}'\mathbf{y} \end{aligned}$$



Now for the ridge regression, we would have

$$\begin{aligned}\hat{y} = \mathbf{X}\hat{\beta}_{ridge} &= \mathbf{X}(\mathbf{X}'\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}'\mathbf{y} \\ &= \mathbf{UDV}[(\mathbf{UDV})'\mathbf{UDV} + \lambda\mathbf{I}]^{-1}(\mathbf{UDV})'\mathbf{y} \\ &= \mathbf{UDV}[\mathbf{V}'\mathbf{D}'\mathbf{U}'\mathbf{UDV} + \lambda\mathbf{I}]^{-1}\mathbf{V}'\mathbf{D}'\mathbf{U}'\mathbf{y} \\ &= \mathbf{UD}[\mathbf{D}'\mathbf{D} + \lambda\mathbf{I}]^{-1}\mathbf{D}'\mathbf{U}\mathbf{y}\end{aligned}$$