STA496 STUDY NOTE

Haonan(Eric) Gao
Supervisor: Prof. Jun Young Park

Contents

1 Linear Regression Model

Assume the regression function $E(Y|X)$ is linear with respect to input $X = (x_1, x_2, ..., x_p)$:

$$f(x) = \beta_0 + \sum_{j=1}^{p} x_j \beta_j \tag{1}$$

1.1 Ordinary Least square

We want to estimate parameter β, we have training dataset $(X_1, Y_1), ..., (X_N, Y_N)$, where each X_i is the feature vector for condition i, so $X_i = (x_{i1}, ..., x_{ip})^T$.
We want to choose the $\beta = (\beta_0, ..., \beta_p)^T$ that make the residual sum of squares(RSS) minimum:

$$\begin{aligned} RSS(\beta) &= \sum_{i=1}^{N} (y_i - f(x_i))^2 \\ &= \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{P} x_{ij}\beta_j)^2 \end{aligned} \tag{2}$$

Here, if we add a dimension on X, where $X : N \times (P+1)$ and $\beta : (P+1) \times 1$.
Therefore, we have:

$$X = \begin{bmatrix} 0 & x_{11} & x_{12} & ... & x_{1P} \\ 0 & x_{21} & x_{22} & ... & x_{2P} \\ ... & ... & ... & ... & ... \\ 0 & x_{N1} & x_{N2} & ... & x_{NP} \end{bmatrix} \tag{3}$$

, we get:

$$RSS(\beta) = (Y - X\beta)^T (Y - X\beta) \tag{4}$$

, this is a quadratic function with $P + 1$ parameters.

Next, we derive the analytic form of the least squares in linear regression,

$$\begin{aligned} RSS(\beta) &= (Y - X\beta)^T (Y - X\beta) \\ &= Y^T Y - 2Y^T X\beta + \beta^T X^T X\beta \end{aligned} \tag{5}$$

We differentiate with respect to β:

$$\begin{aligned} \frac{\partial RSS(\beta)}{\partial \beta} &= \frac{\partial}{\partial \beta}(Y^T Y) - \frac{\partial}{\partial \beta}(2Y^T X\beta) + \frac{\partial}{\partial \beta}(\beta^T X^T X\beta) \\ &= 0 - 2X^T Y + 2X^T X\beta \end{aligned} \tag{6}$$

(recall that $\frac{\partial}{\partial x}(y^T x) = y$ and $\frac{\partial}{\partial x}(x^T A x) = 2Ax$, where A is symmetric.)
Since we want to get the extreme value(minimum), we let the differentiating be 0.

$$\begin{aligned} -2X^T Y + 2X^T X\hat{\beta} &= 0 \\ X^T X\hat{\beta} &= X^T Y \\ \hat{\beta} &= (X^T X)^{-1} X^T Y \end{aligned} \tag{7}$$

However, in most conditions we faced in the real world, X's columns are not linearly independent, or we can say that X is not of full rank. Thus, one thing we can do is get rid of some columns(features) on purpose. Sometimes $P > N$(more features than the training set), and we need to reduce dimension or do regularization.

1.2 Multi-collinearity

We use the variance inflation factor(VIF) to describe if certain columns in X are too similar, or to say the multi-collinearity between features:

$$VIF_i = \frac{1}{1 - R_i^2} \tag{8}$$

where $i = 1, 2, ..., k$.
In VIF, R_i^2 means the coefficient of determination that we treat the i-th feature as y(dependent variable), and use the rest $k - 1$ features to apply a linear regression with respect to y.

What is R^2(coefficient of determination?

- Suppose we have data set $(x_1, y_1), ..., (x_n, y_n)$
- After applying linear regression, we have the predicted set: $(\hat{y_1}, ..., \hat{y_n})$.
- Then, we can have:
  - Observation date mean: $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$
  - Residual: $e_i = y_i - \hat{y_i}$
  - Total sum of square: $SST = \sum_{i=1}^{n}(y_i - \bar{y})^2$
  - Regression sum of square: $SSR = \sum_{i=1}^{n}(\hat{y_i} - \bar{y})^2$
  - Residual sum of square: $SSE = \sum_{i=1}^{n}(y_i - \hat{y})^2 = \sum_{i=1}^{n} e_i^2$
- We can calculate R^2 by:

$$R^2 = 1 - \frac{SSE}{SST} = \frac{SSR}{SST} \tag{9}$$

- This shows the ratio between the regression sum of squares and the total sum of squares, it means the level of explanation by X(independent variable) to Y(dependent variable). Therefore, if R^2 is closer to 1, it means higher multi-collinearity.

1.3 variance-covariance matrix of OLS

For OLS, we have already got:

$$\hat{\beta} = (X^T X)^{-1} X^T Y \tag{10}$$

Assume observation value y_i is not relevant to each other and have fixed variance σ^2, so $Y \sim N(X\beta, \sigma^2 I)$, and x_i is fixed, we can have the variance-covariance matrix for $\hat{\beta}$:

$$
\begin{aligned}
var(\hat{\beta}) &= \begin{bmatrix} Var(\hat{\beta}_1) & Cov(\hat{\beta}_1, \hat{\beta}_2) & ... & Cov(\hat{\beta}_1, \hat{\beta}_p) \\ Cov(\hat{\beta}_2, \hat{\beta}_1) & Var(\hat{\beta}_2) & ... & Cov(\hat{\beta}_2, \hat{\beta}_p) \\ ... & ... & ... & ... \\ Cov(\hat{\beta}_p, \hat{\beta}_1) & Cov(\hat{\beta}_p, \hat{\beta}_2) & ... & Var(\hat{\beta}_p) \end{bmatrix} \\
&= E[(\hat{\beta} - \beta)^T(\hat{\beta} - \beta)] \\
&= E[((X^TX)^{-1}X^TY - \beta)^T((X^TX)^{-1}X^TY - \beta)] \\
&= E[((X^TX)^{-1}X^T(X\beta + e) - \beta)^T((X^TX)^{-1}X^T(X\beta + e) - \beta)] \\
&= E[(\beta + (X^TX)^{-1}X^Te - \beta)^T(\beta + (X^TX)^{-1}X^Te - \beta)] \\
&= E[((X^TX)^{-1}X^Te)^T((X^TX)^{-1}X^Te)] \\
&= E[(X^TX)^{-1}X^Te^TeX(X^TX)^{-1}] \qquad \text{\# Since } (AB)^T = B^TA^T \\
&= (X^TX)^{-1}X^TE[eTe]X(X^TX)^{-1} \\
&= E[e^Te](X^TX)^{-1}
\end{aligned}
\tag{11}
$$

According to The Gauss-Markov assumptions, we have $E[e^Te] = \sigma^2 I$, Thus,

$$
var(\hat{\beta}) = (X^TX)^{-1}\sigma^2 \tag{12}
$$

We estimate σ^2 with $\hat{\sigma^2}$, where:

$$
\begin{aligned}
\hat{\sigma^2} &= \frac{1}{N - P - 1}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \\
&= \frac{\hat{e}^T\hat{e}}{N - P - 1}
\end{aligned}
\tag{13}
$$

, where $N - P - 1$ is the degree of freedom of $\hat{\sigma^2}$, which make $\hat{\sigma^2}$ is the unbiased estimator.

1.4 Objective function for OLS

For $\hat{\beta} = (X^TX)^{-1}X^TY$, if X and Y are standardized, so we have $var(X) = S^2$, where S^2 is the sample variance-covariance matrix(like the one from equation(10)). Thus, for j, $k = 1, ..., P$:

$$
cov(x_j, x_k) = \frac{1}{n-1}\sum_{i=1}^{N}[(x_{ij} - \bar{x})(x_{ik} - \bar{x})] \tag{14}
$$

, since X is standardized:

$$
\begin{aligned}
cor(x_j, x_k) &= \frac{cov(x_j, x_k)}{\sigma_x\sigma_y} \\
&= \frac{1}{n-1}x_j^Tx_k \qquad \text{\# Since } \sigma_x = \sigma_y = 1
\end{aligned}
\tag{15}
$$

So consider X^TX,we have:

$$
X^TX = (n-1)\rho_{(X,X)} \tag{16}
$$

Similarly, for X^TY, we can have:

$$X^TY = (n-1)\rho_{(X,Y)} \tag{17}$$

2 Shrinkage Method

If we want to do feature subset selection, also we do not want to get rid of any feature directly, we can apply some shrinkage methods. Shrinkage methods generally are continuous and do not have a high variability(compares to subset selection, where feature either is used or not used).

2.1 Ridge Regression

For X and Y, where $X : N \times P$ and $Y : N \times 1$, so $x_i = (x_{i1}, ..., x_{iP})$ and $Y = (y_1, ..., y_N)^T$, we add a L2 penalized factor to estimate β for ridge regression:

$$\begin{aligned} \hat{\beta}^{ridge} &= arg_\beta min\{RSS(Y) + L2\} \\ &= arg_\beta min\{\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{P} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{P}\beta_j^2\} \end{aligned} \tag{18}$$

Here, $\lambda \geq 0$ is the parameter that controls the shrinkage level, if λ goes higher, means β is more shrinkage so that $\hat{\beta}$ is overall smaller than $\hat{\beta}(OLS)$.
If we let $\beta = (\beta_0, \beta_1, ..., \beta_P)$, from equation (17), we have:

$$\hat{\beta}^{ridge} = arg_\beta min\{(Y - X\beta)^T(Y - X\beta) + \lambda\beta^T\beta\} \tag{19}$$

Similar to OLS objective function, we differentiate with respect to β, and let it be 0 to get the minimum, get:

$$\hat{\beta}^{ridge} = (X^TX + \lambda I)^{-1}X^TY \tag{20}$$

Here, this shows that it added a positive constant λ at the diagonal of X^TX, so that we do not need to restrict X^TX to be full rank(no need to be linearly independent).
However, if all the columns(features) in X are linearly independent, or saying the inputs are orthogonal, so there is no single multi-collinearity, the estimated β from ridge regression have the same eigenvector with the one from OLS, and there relation is:

$$\hat{\beta}^{ridge} = \frac{\hat{\beta}}{1 + \lambda} \tag{21}$$

, this is because for orthogonal inputs: $X^TX = I_p$.
We can also get similar results from Bayesian Regression, recalling some properties:

$$P(\theta|D) = \frac{P(D|\theta) \cdot P(\theta)}{P(D)} \tag{22}$$

- $P(\theta)$: prior distribution
- $P(\theta|D)$: posterior distribution

- $P(D|\theta)$: likelihood
- $P(D)$: evidence
- If prior and posterior belong to the same distribution, then we call them conjugate priors. e.g. $X \sim Binomial(n, p)$ and $p \sim Beta$.

For ridge regression, assuming normal prior and normal likelihood, we have:

$$\begin{aligned} Y|\beta &\sim N_N(X\beta, \sigma^2 I) \\ \beta &\sim N_P(0, \tau^2 I) \end{aligned} \tag{23}$$

, where $\beta = (\beta_0, ..., \beta_P)$. We can have:

$$\begin{aligned} P(\beta|Y) &\propto P(\beta) \cdot P(Y|\beta) \\ &\propto e^{-\frac{1}{2}(\beta-0)^T \tau^{-2} I(\beta-0)} \cdot e^{-\frac{1}{2}(Y-X\beta)^T \sigma^{-2} I(Y-X\beta)} \\ &\propto e^{-\frac{1}{2\sigma^2}(Y-X\beta)^T(Y-X\beta) - \frac{1}{2\tau^2}||\beta||_2^2} \end{aligned} \tag{24}$$

We want to get the maximum $P(\beta|Y)$, so:

$$\begin{aligned} \text{posterior mode } \hat{\beta} &= arg_\beta max\{-\frac{1}{2\sigma^2}(Y-X\beta)^T(Y-X\beta) - \frac{1}{2\tau^2}||\beta||_2^2\} \\ &= arg_\beta min\{(Y-X\beta)^T(Y-X\beta) + \frac{\sigma^2}{\tau^2}||\beta||_2^2\} \\ &= arg_\beta min\{(Y-X\beta)^T(Y-X\beta) + \lambda\beta^T\beta\} \end{aligned} \tag{25}$$

, where $\lambda = \frac{\sigma^2}{\tau^2}$. This is the same as the equation (18).

2.2 Lasso Regression

For X and Y, where $X : N \times P$ and $Y : N \times 1$, so $x_i = (x_{i1}, ..., x_{iP})$ and $Y = (y_1, ..., y_N)^T$, we add a L1 penalized factor to estimate β for lasso regression:

$$\begin{aligned} \hat{\beta}^{lasso} &= arg_\beta min\{RSS(Y) + L1\} \\ &= arg_\beta min\{\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{P} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{P}|\beta_j|\} \end{aligned} \tag{26}$$

Since the absolute term cannot be differentiated directly, we can use sub-differentiate and the coordinate descent method for lasso when Y, X and λ are given.

$$\begin{aligned}\frac{\partial}{\partial\beta_k}\hat{\beta}^{lasso} &= \frac{\partial}{\partial\beta_k}(\sum_{i=1}^{N}(y_i-\sum_{j=0}^{P}x_{ij}\beta_j)^2+\frac{\partial}{\partial\beta_k}\lambda\sum_{j=1}^{P}|\beta_j|)\\ &= \begin{cases}-2\cdot\sum_{i=1}^{N}(y_i-\sum_{j=0}^{P}x_{ij}\beta_j)x_{ik}+(-1)\cdot\lambda & \text{,where } \beta_k<0\\ -2\cdot\sum_{i=1}^{N}(y_i-\sum_{j=0}^{P}x_{ij}\beta_j)x_{ik}+[-1,1]\cdot\lambda & \text{,where } \beta_k=0\\ -2\cdot\sum_{i=1}^{N}(y_i-\sum_{j=0}^{P}x_{ij}\beta_j)x_{ik}+1\cdot\lambda & \text{,where } \beta_k>0\end{cases}\\ &= \begin{cases}-2\cdot\sum_{i=1}^{N}(y_i-\sum_{(j=0\&j\neq k)}^{P}x_{ij}\beta_j)x_{ik}+2\cdot\beta_k\sum_{i=1}^{N}x_{ik}^2+(-1)\cdot\lambda & \text{,where } \beta_k<0\\ -2\cdot\sum_{i=1}^{N}(y_i-\sum_{(j=0\&j\neq k)}^{P}x_{ij}\beta_j)x_{ik}+2\cdot\beta_k\sum_{i=1}^{N}x_{ik}^2+[-1,1]\cdot\lambda & \text{,where } \beta_k=0\\ -2\cdot\sum_{i=1}^{N}(y_i-\sum_{(j=0\&j\neq k)}^{P}x_{ij}\beta_j)x_{ik}+2\cdot\beta_k\sum_{i=1}^{N}x_{ik}^2+1\cdot\lambda & \text{,where } \beta_k>0\end{cases}\end{aligned} \tag{27}$$

Let $h_k = 2\cdot\sum_{i=1}^{N}(y_i-\sum_{(j=0\&j\neq k)}^{P}x_{ij}\beta_j)x_{ik}$, we can conclude 3 conditions:

- $\beta_k<0$: we have

$$-h_k+2\cdot\beta_k\sum_{i=1}^{N}x_{ik}^2-\lambda=0 \tag{28}$$

  Therefore,

$$\beta_k=\frac{h_k+\lambda}{2\cdot\sum_{i=1}^{N}x_{ik}^2},\text{ where } h_k<-\lambda \tag{29}$$

- $\beta_k>0$: we have

$$-h_k+2\cdot\beta_k\sum_{i=1}^{N}x_{ik}^2+\lambda=0 \tag{30}$$

  Therefore,

$$\beta_k=\frac{h_k-\lambda}{2\cdot\sum_{i=1}^{N}x_{ik}^2},\text{ where } h_k>\lambda \tag{31}$$

- $\beta_k=0$: we have

$$0\in-h_k+\lambda\cdot[-1,1] \tag{32}$$

We can write the pseudocode for the coordinate descent algorithm:

Algorithm 1 coordinate descent for lasso

Require: X, Y and λ are known

while reach MAX_ITER or $|\hat{\beta}_{previous} - \hat{\beta}| \leq 0.0001$ **do**

for $k = 0, 1, ..., P$ **do**

compute $h_k = \sum_{i=1}^{N}(y_i - \sum_{(j=0\&j\neq k)}^{P} x_{ij}\beta_j)x_{ik}$

update:

$$\beta_k = \begin{cases} \beta_k = \frac{h_k+\lambda}{\sum_{i=1}^{N} x_{ik}^2} = (h_k + \lambda) & \text{, where } h_k < -\lambda \\ \beta_k = \frac{h_k-\lambda}{\sum_{i=1}^{N} x_{ik}^2} = (h_k - \lambda) & \text{, where } h_k > \lambda \\ 0 & \text{, where } -\lambda \leq h_k \leq \lambda \end{cases} \tag{33}$$

end for

end while

- Question for Linear regression: If every covariate (column of X) and response (y) are standardized, do we need an intercept term (β_0)?
  Assume linear model $y_j = \sum_{i=1}^{P} x_{ij}\beta_i + \beta_0 + e_j$, j = 0,...,N, β_0 is the intercept, and we standardized all the variables. We have:

$$\begin{aligned} E[y] &= \hat{y} \\ &= \frac{1}{N}\sum_{j=1}^{N} y_j \\ &= \frac{1}{N}\sum_{j=1}^{N}(x_{ij}\beta_i + \beta_0 + e_j) \\ &= \frac{1}{N}\sum_{j=1}^{N}(\sum_{i=1}^{P} x_{ij}\beta_i) + \frac{1}{N}\cdot N \cdot \beta_0 + \frac{1}{N}\sum_{j=1}^{N} e_j \\ &= \beta_0 + \sum_{i=1}^{P}(\frac{1}{N}\sum_{j=1}^{N} x_{ij})\beta_i \\ &= \beta_0 \qquad \text{\# Since the average of each column of X equals to 0} \end{aligned} \tag{34}$$

  Also since we standardized y as well, so:

$$\hat{y} = \beta_0 = 0 \tag{35}$$

  In other way saying, the linear regression fitted line always goes through the point $(\bar{X}, \bar{y}$, when all the variables are standardized.

- RIDGE and LASSO standardizes all variables as default. Why?
  1. No need for an intercept(β_0).
  2. For L_N, we will take the summation of all the β power of N anyway, so we need all the beta from the same scalar, therefore we need all the x or column in X to be in the same scalar, which needs X to be standardized.

2.3 Elastic net(https://hastie.su.domains/Papers/B67.2%20(2005)%20301-320%20Zou%20&%20Hastie.pdf)

Since Ridge regression and Lasso regression are not perfect and all have drawbacks:

- when $P > N$, lasso select at most n variables.
- If the pairwise correlation between variables is too high, lasso tends to only randomly select 1 from the group.
- when $N > P$, if the correlation between predictors is high, ridge is far better than lasso.

If we have two tuning parameters λ_1 and λ_2 that correspond to ridge and LASSO penalties, how to get beta coefficients? We have the estimated beta for Elastic net:

$$\hat{\beta}(EN) = arg_{\beta} min\{(Y - X\beta)^2 + \lambda_2|\beta|_2^2 + \lambda_1|\beta|_1^1\} \tag{36}$$

We can create an artificial data set (X^*, Y^*), where:

$$\begin{aligned} X^*_{(N+P)\times P} &= \frac{1}{\sqrt{1+\lambda_2}} \begin{bmatrix} X \\ \sqrt{\lambda_2} \cdot I \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{11} & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{12} & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{13} & ... & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{1P} \\ \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{21} & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{22} & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{23} & ... & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{2P} \\ ... & ... & ... & ... & ... \\ \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{N1} & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{N2} & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{N3} & ... & \frac{1}{\sqrt{1+\lambda_2}} \cdot X_{NP} \\ \sqrt{\frac{\lambda_2}{1+\lambda_2}} & 0 & 0 & ... & 0 \\ 0 & \sqrt{\frac{\lambda_2}{1+\lambda_2}} & 0 & ... & 0 \\ 0 & 0 & \sqrt{\frac{\lambda_2}{1+\lambda_2}} & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & \sqrt{\frac{\lambda_2}{1+\lambda_2}} \end{bmatrix} \\ Y^*_{(N+P)\times 1} &= \begin{bmatrix} y_1 \\ y_2 \\ ... \\ y_N \\ 0 \end{bmatrix} \end{aligned} \tag{37}$$

Then, if we assume $\hat{\beta}^* = \sqrt{1+\lambda_2} \cdot \hat{\beta}$ and $\gamma = \frac{\lambda_1}{\sqrt{1+\lambda_2}}$ we can get the estimate β^* expressing as a lasso regression:

$$\hat{\beta}^* = arg_{\beta} min\{(Y^* - X^*\beta^*)^2 + \gamma||\beta^*||_1^1\} \tag{38}$$

, plugging in $X^*, Y^*, \hat{\beta}^*$ and γ:

$$\begin{aligned}
\hat{\beta}^* &= arg_\beta min\{Y^{*T}Y^* - 2Y^{*T}X^*\beta + \beta^{*T}X^{*T}X^*\beta^* + \gamma||\beta^*||_1^1\} \\
&= arg_\beta min\{Y^TY - 2\cdot\frac{1}{\sqrt{1+\lambda_2}}Y^TX\beta^* + \beta^*\cdot\frac{X^TX+\lambda_2 I}{1+\lambda_2}\beta^* + \frac{\lambda_1}{\sqrt{1+\lambda_2}}||\beta^*||_1^1\} \\
&= arg_\beta min\{-2\cdot\frac{1}{\sqrt{1+\lambda_2}}Y^TX\beta^* + \frac{\beta^{*T}X^TX\beta^*}{1+\lambda_2} + \frac{\lambda_2\beta^{*T}\beta^*}{1+\lambda_2} + \frac{\lambda_1}{\sqrt{1+\lambda_2}}||\beta^*||_1^1\} \\
&= arg_\beta min\{-2Y^TX\beta + \beta^TX^TX\beta + \lambda_2\beta^T\beta + \lambda_1||\beta||_1^1\} \\
&= arg_\beta min\{(Y-X\beta)^2 + \lambda_2|\beta|_2^2 + \lambda_1|\beta|_1^1\}
\end{aligned} \tag{39}$$

, thus, we can transform the elastic net regression into a lasso regression.

2.4 Degree of Freedom

Recall the OLS, we have the estimated β for X and Y where X is (N×P) and Y is (N×1):

$$\beta(OLS) = (X^TX)^{-1}X^TY \tag{40}$$

Therefore, we can get the estimated Y, or we can call it Y-Hat:

$$\begin{aligned}
\hat{Y} &= X\beta \\
&= X(X^TX)^{-1}X^TY \\
&= S\cdot Y
\end{aligned} \tag{41}$$

Here, we let $S = X(X^TX)^{-1}X^T$, matrix S is (N×N), where it put on a "Hat" for Y. We assume that y are independent, then we can get the effective degree of freedom for OLS:

$$\begin{aligned}
df(\mu) &= \sum_{i=1}^{N}\frac{\partial\hat{y}_i}{\partial y_i} \\
&= \sum_{i=1}^{N}h_{ii} \\
&= tr(S) \\
&= tr(X(X^TX)^{-1}X^T) \\
&= tr((X^TX)^{-1}X^TX) \\
&= tr(I_P)
\end{aligned} \tag{42}$$

, where the h_{ii} is the diagonal term in the matrix. The sum of the diagonals of a matrix is called the Trace of the matrix.
As for ridge regression, we also have the closed-form solution for estimated β:

$$\beta(RIDGE) = (X^TX - \lambda I)^{-1}X^TY \tag{43}$$

We then compute the fitted values and get the hat matrix S:

$$\begin{aligned}\hat{Y} &= X\beta \\ &= X(X^TX + \lambda I)^{-1}X^TY \\ &= S \cdot Y\end{aligned} \tag{44}$$

Here, we need to use singular value decomposition to choose bases for the vector spaces, and we can write X as a diagonal matrix to compute S:

$$\begin{aligned}S &= X(X^TX + \lambda I)^{-1}X^T \\ &= \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \cdots & \\ & & & d_p \end{bmatrix} \begin{bmatrix} d_1^2 + \lambda & & & \\ & d_2^2 + \lambda & & \\ & & \cdots & \\ & & & dp^2 + \lambda \end{bmatrix}^{-1} \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \cdots & \\ & & & d_p \end{bmatrix} \\ &= \begin{bmatrix} \frac{d_1^2}{d_1^2+\lambda} & & & & & & \\ & \frac{d_2^2}{d_2^2+\lambda} & & & & & \\ & & \cdots & & & & \\ & & & \frac{d_p^2}{d_p^2+\lambda} & & & \\ & & & & 0 & & \\ & & & & & \cdots & \\ & & & & & & 0 \end{bmatrix}\end{aligned} \tag{45}$$

Thus,

$$\begin{aligned}df(\lambda) &= tr(S) \\ &= \sum_{i=1}^{P} \frac{d_i^2}{d_i^2 + \lambda}\end{aligned} \tag{46}$$

3 Model Selection

Main purpose: avoid over-fitting.

Model Selection $\begin{cases} \text{sample re-use: cross-validation, bootstrap} \\ \text{analytically: AIC, BIC, MDL, SRM} \end{cases}$

3.1 Cross-validation

The main idea is to divide data into training set and validation set, after fitting the model, we use the validation set to the generated model, and see the error rate. Error rate usually is mean-square error(MSE) $= \frac{1}{n}\sum_{i=1}^{1}(Y_I - \hat{Y}_i)^2$.

3.1.1 Leave-one-out cross-validation(LOOCV)

We can have some improvement on simple cross-validation, every time we took one from the set, and use the rest (n-1) to be the training set. We repeat this n times, therefore:

$$CV_N = \frac{1}{n}\sum_{i=1}^{n} MSE_i \text{ , where } MSE_i = (y_i - \hat{y_i})^2 \tag{47}$$

In such a way, we can have less bias, or to say less error rate.

3.1.2 k-fold cross-validation

If we equally separate the dataset into k sets, we take one set as the validation set. Notice that if k=n, then it is LOOCV, so LOOCV is a special case of k-fold cross-validation. K-fold validation generally has a much smaller calculation than LOOCV.

3.1.3 Bias-Variance Trade-off

If we do not consider the calculation effort, then why are we just not using LOOCV for all the cases, it has less bias(error rate)?
Quote from the ELS: "The mean of many highly correlated quantities has higher variance than does the mean of many quantities that are not as high correlated." So if the bias of LOOCV is less, but the variance of the LOOCV test error is greater than k-fold cross-validation.

3.2 Akaike information criterion(AIC)

$$AIC = -\frac{2}{N} \cdot loglik + 2 \cdot \frac{d}{N} \tag{48}$$

, where $loglik = \sum_{i=1}^{N} logPr_{\hat{\theta}}(y_i)$, this is the maximum log-likelihood, and $\hat{\theta}$ is the maximum likelihood estimation of θ, d is the number of parameter.

- In general, the model is better when AIC is smaller.
- Use AIC when models that we are comparing have similar N(rows).
- When model complexity(d) is greater, then likelihood(L) is greater, therefore AIC will drop. However, if complexity(d) keeps increasing, L will increase less than before, so AIC will eventually increase.

3.3 Bayesian information criterion(BIC)

$$BIC = -\frac{2}{N} \cdot loglik + \frac{log(N)}{N} \cdot d \tag{49}$$

- Compares to AIC, BIC penalizes complex models(large N) more heavily since $log(N)$, so it would prefer simpler model.

3.4 Selection criteria on penalized regression

According to the result from `https://arxiv.org/abs/0712.0881`, we have the adaptive model selection criteria, where:

$$\begin{aligned} AIC(\hat{\mu}) &= \frac{||y-\hat{\mu}||^2}{n\sigma^2} + \frac{2}{n}\hat{df}(\hat{\mu}) \\ BIC(\hat{\mu}) &= \frac{||y-\hat{\mu}||^2}{n\sigma^2} + \frac{\log(n)}{n}\hat{df}(\hat{\mu}) \end{aligned} \tag{50}$$

, n is the observation number(row amount from X), σ^2 is the true regression Variance (assume $\vec{y} \sim MVN(X\beta, \sigma^2 I_N)$), and $\hat{\mu}$ is the estimated regression mean value.

4 Summary Statistics

4.1 Covariance Regression Model

Consider:

- N is the observation amount and P is the feature amount
- $\vec{Y} = [y_1, ..., y_N]^T$ and $X = \begin{bmatrix} x_{(1,1)} & ... & x_{(1,P)} \\ ... & ... & ... \\ x_{(N,1)} & ... & x_{(N,P)} \end{bmatrix} = [X^{(1)}, ..., X^{(P)}]$
- Let $W(X^{(p)}) = (w(x_{(N_1,p)}, x_{(N_2,p)}))_{N\times N} \in \mathbb{R}^{N\times N}$, here $X^{(p)} = [x_{(1,P)}, ..., x_{(N,P)}]^T$. Specifically,

$$W(X^{(p)}) = \begin{bmatrix} 1 & w(x_{(1,p)}, x_{(2,p)}) & ... & w(x_{(1,p)}, x_{(N,p)}) \\ w(x_{(2,p)}, x_{(1,p)}) & ... & ... & ... \\ ... & ... & ... & w(x_{(N-1,p)}, x_{(N,p)}) \\ w(x_{(N,p)}, x_{(1,p)}) & ... & w(x_{(N,p)}, x_{(N-1,p)}) & 1 \end{bmatrix} \tag{51}$$

- We let $W(X^{(p)}) = (w(x_{(N_1,p)}, x_{(N_2,p)}))_{N\times N} \in \mathbb{R}^{N\times N}$, here $X^{(p)} = [x_{(1,P)}, ..., x_{(N,P)}]^T$, we call $W(X^{(p)})$ the similarity matrix. Specifically,

$$\begin{aligned} W(X^{(p)}) &= \begin{bmatrix} w(x_{(1,p)}, x_{(1,p)}) & ... & w(x_{(1,p)}, x_{(N,p)}) \\ ... & ... & ... \\ w(x_{(N,p)}, x_{(1,p)}) & ... & w(x_{(N,p)}, x_{(N,p)}) \end{bmatrix} \\ &= \begin{bmatrix} 1 & w(x_{(1,p)}, x_{(2,p)}) & ... & w(x_{(1,p)}, x_{(N,p)}) \\ w(x_{(2,p)}, x_{(1,p)}) & ... & ... & ... \\ ... & ... & ... & w(x_{(N-1,p)}, x_{(N,p)}) \\ w(x_{(N,p)}, x_{(1,p)}) & ... & w(x_{(N,p)}, x_{(N-1,p)}) & 1 \end{bmatrix} \end{aligned} \tag{52}$$

We introduce the covariance regression model:

$$YY^T = \beta_0 I_N + \beta_1 W(X^{(1)}) + ... + \beta_P W(X^{(P)}) + e \tag{53}$$

Let $W_P = W(X^{(P)})$, we define

$$\begin{aligned}\Sigma(\beta) &= E(YY^T) \\ &= \beta_0 I_N + \Sigma_{p=1}^{p} \beta_p W_p\end{aligned} \tag{54}$$

Here, YY^T is the sample covariance and $\Sigma(\beta)$ is the model-based covariance.

4.2 Model selection criteria on Summary Statistics

In our case, we assume that for summary statistics, we only have the property of X^TX and X^TY, where their dimensions are $P \times P$ and $P \times 1$.

Here comes the question: what if we want to apply the model selection criteria to a set of summary data with penalized regression?

According to equation(50), n and $\hat{df}(\hat{\mu})$ are known, but how to estimate σ^2 and $||y - \hat{\mu}||^2$?

For $||y - \hat{\mu}||^2$:

- we can write as $(y - X\hat{\beta})^T(y - X\hat{\beta}) = y^Ty - 2y^TX\hat{\beta} + \hat{\beta}^TX^TX\hat{\beta}$
- For y^Ty, since y is standardized, where $var(y_i) = 1$ and $\bar{y_i} = 0$, i.e.

$$\begin{aligned}var(y_i) &= \frac{\sum_{i=1}^{N}(y_i - \bar{y})^2}{N-1} \\ &= \frac{\sum_{i=1}^{N} y_i^2}{N-1} \\ &= 1\end{aligned} \tag{55}$$

  therefore, $y^Ty = \sum_{i=1}^{N} y_i^2 = (N-1) \cdot var(y_i) = N - 1$

- We have y^TX and X^TX from summary data.

For σ^2:

- From section 2.1, we know that for $\vec{y} = X\beta + \epsilon$, we can use:

$$\begin{aligned}\vec{\epsilon} &\sim N(\vec{0}, \tau^2 I_N) \\ \vec{\beta} &\sim N(\vec{0}, \theta^2 I_P)\end{aligned} \tag{56}$$

  to represent Ridge regression.

- Question: How can we obtain $\hat{\theta}^2$ and $\hat{\tau}^2$ from summary statistics?

- Method 1:
  According to `https://pubmed.ncbi.nlm.nih.gov/33145600/`, we can use a method-of-moments estimator of the variance by all observations:

$$\hat{S}^2 = \hat{m}_{eff}(\bar{z}^2 - 1) \tag{57}$$

where $\bar{z}^2$ is the mean of the squared z-statistics of the OLS regression of X, and $\hat{m}_{eff}$ is the estimated effective number of observations, where it equals N divided by the second spectral moment of the correlation matrix $X^T X$.
Then,

$$\begin{aligned} \hat{\theta}^2 &= N \cdot \hat{S}^2 \\ \hat{\tau}^2 &= (1 - \hat{S}^2) \end{aligned} \tag{58}$$

By equation (25), use the close form solution of Bayesian regression, we can plug $\hat{\theta}$ and $\hat{\tau}$ in:

$$\hat{\beta} = (X^T X + \frac{\hat{\tau}^2}{\hat{\theta}^2} I_P)^{-1} X^T y \tag{59}$$

- Method 2:
  According to `https://www.tandfonline.com/doi/full/10.1080/01621459.2015.1131699`. We first want to get the distribution for y. Here, we have:

$$\begin{aligned} E[\vec{y}] &= E[X\beta + \epsilon] \\ &= X \cdot E[\beta] + E[\epsilon] \\ &= 0 \end{aligned} \tag{60}$$

  Also, for the variance-covariance of $\vec{y}$, we have:

$$\begin{aligned} cov(\vec{y}) &= cov(X\beta + \epsilon) \\ &= X \cdot cov(\beta) \cdot X^T + cov(\epsilon) \end{aligned} \tag{61}$$

  For $cov(\beta)$ and $cov(\epsilon)$, by equation (52), we have:

$$\begin{aligned} \epsilon_i &\overset{\text{iid}}{\sim} N(\vec{0}, \tau^2 I_N) \\ \beta_j &\overset{\text{iid}}{\sim} N(\vec{0}, \theta^2 I_P) \end{aligned} \tag{62}$$

, where i = 1,..., N and j = 1,..., P. Therefore,

$$
\begin{aligned}
cov(\beta) &= \begin{bmatrix} var(\beta_1) & ... & cov(\beta_P, \beta_1) \\ ... & ... & ... \\ cov(\beta_1, \beta_P) & ... & var(\beta_P) \end{bmatrix} \\
&= \begin{bmatrix} \theta^2 & 0 & ... & 0) \\ 0 & \theta^2 & ... & ... \\ ... & ... & \theta^2 & 0 \\ 0 & ... & 0 & \theta^2 \end{bmatrix} \\
&= \theta^2 I_P \\
cov(\epsilon) &= \begin{bmatrix} var(\epsilon_1) & ... & cov(\epsilon_N, \epsilon_N) \\ ... & ... & ... \\ cov(\epsilon_1, \epsilon_N) & ... & var(\epsilon_N) \end{bmatrix} \\
&= \begin{bmatrix} \tau^2 & 0 & ... & 0 \\ 0 & \tau^2 & ... & ... \\ ... & ... & \tau^2 & 0 \\ 0 & ... & 0 & \tau^2 \end{bmatrix} \\
&= \tau^2 I_N
\end{aligned} \tag{63}
$$

Thus, $cov(\vec{y}) = \theta^2 X X^T + \tau^2 I_N$

Therefore we can conclude that $\vec{y} \sim MVN(\vec{0}, \theta^2 X X^T + \tau^2 I_N)$.

According to the paper or equation (52), we can get the estimated beta by minimizing the Frobenius Norm of the difference between sample covariance and model-based covariance, which is:

$$||yy^T - \Sigma(\beta)||_F^2 \tag{64}$$

, by the y distribution from the Bayesian regression, we can plug in the model-based covariance, so we want to get the arg-min of:

$$||yy^T - (\theta^2 X X^T + \tau^2 I_N)||_F^2 \tag{65}$$

where sample variance yy^T and model-based covariance are both N×N matrix. The parameters for this arg-min is θ and τ.

From equation (53), we use the model-based variance equation, we have:

$$
\begin{aligned}
\Sigma(\beta) &= E(YY^T) \\
&= \beta_0 I_N + \Sigma_{p=1}^{p} \beta_p W_p \\
&= \beta_0 I_N + \beta_1 W_1 \\
&= \tau^2 I_N + \theta^2 X X^T
\end{aligned} \tag{66}
$$

From the paper, we have the closed-form solution for equation (64), where we use

matrix derivative with respect to θ and τ, therefore:

$$\begin{aligned}
\hat{\beta}(\tau^2, \theta^2) &= (tr(W_k W_l))^{-1}_{(K+1)\times(K+1)} (Y^T W_k Y)_{(K+1)\times 1} \\
&= (tr(W_k W_l))^{-1}_{2\times 2} (Y^T W_k Y)_{2\times 1} \\
&= \begin{bmatrix} tr(W_0 W_0) & tr(W_0 W_1) \\ tr(W_1 W_0) & tr(W_1 W_1) \end{bmatrix}^{-1} \begin{bmatrix} Y^T W_0 Y \\ Y^T W_1 Y \end{bmatrix} \\
&= \begin{bmatrix} tr(I_N I_N) & tr(I_N X X^T) \\ tr(X X^T I_N) & tr(X X^T X X^T) \end{bmatrix}^{-1} \begin{bmatrix} Y^T I_N Y \\ Y^T X X^T Y \end{bmatrix} \\
&= \begin{bmatrix} N & tr(X^T X) \\ tr(X^T X) & tr(X^T X X^T X) \end{bmatrix}^{-1} \begin{bmatrix} Y^T Y \\ (X^T Y)^T (X^T Y) \end{bmatrix} \qquad \textbf{\# tr(ab) = tr(ba)}
\end{aligned} \tag{67}$$

In this case, we can calculate beta hat from summary statistics.

4.3 Bias and Variance for the estimators

4.3.1 Ridge estimator

Consider $y = X\beta + e$, where $E[e|X] = 0$ and $Var[e|X] = \sigma^2 I_N$.
The bias of the Ridge estimator is:

$$E[\hat{\beta}_\lambda | X] - \beta \tag{68}$$

For $E[\hat{\beta}_\lambda | X]$, we have:

$$\begin{aligned}
E[\hat{\beta}_\lambda | X] &= E[(X^T X + \lambda I)^{-1} X^T y] \\
&= E[(X^T X + \lambda I)^{-1} X^T (X\beta + e)] \\
&= (X^T X + \lambda I)^{-1} X^T X \beta + (X^T X + \lambda T)^{-1} X^T E[e|X] \\
&= (X^T X + \lambda I)^{-1} X^T X \beta
\end{aligned} \tag{69}$$

Thus, the ridge estimator is unbiased if and only if $(X^T X + \lambda I)^{-1} X^T X = I$.
i.e., when $\lambda = 0$, the ridge estimator is the same as the OLS estimator, and those estimators are unbiased.

Recall from equation (12), we know that the variance-covariance of OLS estimator is $Var[\hat{\beta}_{OLS}|X] = (X^T X)^{-1}\sigma^2$. We can write the ridge estimator using OLS estimator:

$$\begin{aligned}
\hat{\beta}_{ridge} &= (X^T X + \lambda I)^{-1} X^T y \\
&= (X^T X + \lambda I)^{-1} X^T X \hat{\beta}_{OLS}
\end{aligned} \tag{70}$$

Therefore,

$$\begin{aligned} Var[\hat{\beta}_{ridge}|X] &= Var[(X^TX+\lambda I)^{-1}X^TX\hat{\beta}_{OLS}] \\ &= (X^TX+\lambda I)^{-1}X^TX \cdot Var[\hat{\beta}_{OLS}|X] \cdot [(X^TX+\lambda I)^{-1}X^TX]^T \\ &= (X^TX+\lambda I)^{-1}X^TX \cdot (X^TX)^{-1}\sigma^2 \cdot X^TX(X^TX+\lambda I)^{-1} \\ &= \sigma^2(X^TX+\lambda I)^{-1}X^TX(X^TX+\lambda I)^{-1} \end{aligned} \tag{71}$$

4.4 Summary for Linear Model on Summary Statistics

Suppose we have X^TX and X^Ty for summary data, we firstly estimate $\hat{\beta}$, tuning parameters(λ) is selected manually from a set of potential λs:

- Ridge: Use the closed-form solution for Rdige regression: $\hat{\beta} = (X^TX+\lambda I)^{-1}X^Ty$.
- Lasso: By Coordinate Descent Algorithm, for j = 1, 2, ..., P:

$$\begin{aligned} \hat{\beta}_j(OLS) =& X^Ty_{(j)} - [(X^TX)_{(j,1)}\beta_1 + (X^TX)_{(j,2)}\beta_2 + ... + (X^TX)_{(j,j-1)}\beta_{j-1} \\ &+ (X^TX)_{(j,j+1)}\beta_{j+1} + ...(X^TX)_{(j,P)}\beta_P] \\ =& X^Ty_{(j)} - \Sigma^P_{i=1\&i\neq j}[(X^TX)_{(j,i)}\beta_i] \\ =& X^Ty_{(j)} - h_j \end{aligned} \tag{72}$$

  Therefore,

$$\begin{aligned} \hat{\beta}_j(lasso) &= \frac{\partial}{\partial\beta_j}[RSS(\beta)+L1] \\ &= \begin{cases} X^Ty_{(j)} - h_j + \lambda & \text{, where } (X^Ty_{(j)} - h_j) < -\lambda \\ 0 & \text{, where } -\lambda \leq (X^Ty_{(j)} - h_j) \leq \lambda \\ X^Ty_{(j)} - h_j - \lambda & \text{, where } (X^Ty_{(j)} - h_j) > \lambda \end{cases} \end{aligned} \tag{73}$$

- Elastic Net: Similarly to Lasso,

$$\begin{aligned} \hat{\beta}_j(EN) =& \frac{\partial}{\partial\beta_j}[RSS(\beta)+L1+L2] \\ =& \begin{cases} \frac{1}{1+\lambda_2}\cdot(X^Ty_{(j)} - h_j + \lambda_1) & \text{, where } (X^Ty_{(j)} - h_j) < -\lambda_1 \\ 0 & \text{, where } -\lambda_1 \leq (X^Ty_{(j)} - h_j) \leq \lambda_1 \\ \frac{1}{1+\lambda_2}\cdot(X^Ty_{(j)} - h_j - \lambda_1) & \text{, where } (X^Ty_{(j)} - h_j) > \lambda_1 \end{cases} \end{aligned} \tag{74}$$

Next, we will select the better tuning parameters from a set of linear model, as an example, we use AIC and BIC model selection criteria:

$$\begin{aligned} AIC(\hat{\mu}) =& \frac{||y-\hat{\mu}||^2}{n\sigma^2} + \frac{2}{n}\hat{df}(\hat{\mu}) \\ BIC(\hat{\mu}) =& \frac{||y-\hat{\mu}||^2}{n\sigma^2} + \frac{\log(n)}{n}\hat{df}(\hat{\mu}) \end{aligned} \tag{75}$$

- Ridge

- $\hat{df}(\hat{\mu}) = \textbf{tr(Hat Matrix)} = \textbf{tr}(X(X^TX + \lambda I_P)^{-1}X^T)$
  $= \textbf{tr}((X^TX + \lambda I_P)^{-1}X^TX)$
- $||y - \hat{\mu}||^2 = y^Ty - 2y^TX\hat{\beta} + \hat{\beta}^TX^TX\hat{\beta}$
  $= (N-1) - 2(X^Ty)^T\hat{\beta} + \hat{\beta}^T(X^TX)\hat{\beta}$
- $\begin{bmatrix}\sigma^2 \\ \tau^2\end{bmatrix} = \begin{bmatrix}N & tr(X^TX) \\ tr(X^TX) & tr(X^TXX^TX)\end{bmatrix}^{-1}\begin{bmatrix}y^Ty \\ (X^Ty)^T(X^Ty)\end{bmatrix}$

- Lasso
  - $\hat{df}(\hat{\mu}) = \textbf{non-zero coefficient amount(non-zero value amount in beta hat)}$
  - $||y - \hat{\mu}||^2 = y^Ty - 2y^TX\hat{\beta} + \hat{\beta}^TX^TX\hat{\beta}$
    $= (N-1) - 2(X^Ty)^T\hat{\beta} + \hat{\beta}^T(X^TX)\hat{\beta}$
  - $\begin{bmatrix}\sigma^2 \\ \tau^2\end{bmatrix} = \begin{bmatrix}N & tr(X^TX) \\ tr(X^TX) & tr(X^TXX^TX)\end{bmatrix}^{-1}\begin{bmatrix}y^Ty \\ (X^Ty)^T(X^Ty)\end{bmatrix}$

- Elastic Net
  - $\hat{df}(\hat{\mu}) = \textbf{non-zero coefficient amount(non-zero value amount in beta hat)}$
  - $||y - \hat{\mu}||^2 = y^Ty - 2y^TX\hat{\beta} + \hat{\beta}^TX^TX\hat{\beta}$
    $= (N-1) - 2(X^Ty)^T\hat{\beta} + \hat{\beta}^T(X^TX)\hat{\beta}$
  - $\begin{bmatrix}\sigma^2 \\ \tau^2\end{bmatrix} = \begin{bmatrix}N & tr(X^TX) \\ tr(X^TX) & tr(X^TXX^TX)\end{bmatrix}^{-1}\begin{bmatrix}y^Ty \\ (X^Ty)^T(X^Ty)\end{bmatrix}$

4.5 Using SVD on Summary Data

Now, consider that if we only have Cor(X,y) and k column of Cor(X,X), is it possible to have an estimation of the real Cor(X,X) and so to apply the Liner Model in section 4.4? Recall the Singular Value Decomposition(SVD) of the centred input matrix X(we assume the summary data is standardized), so the SVD of the $N \times P$ matrix X has the form:

$$X = UDV^T \tag{76}$$

U is a $N \times P$ orthogonal matrix and V is a $P \times P$ orthogonal matrix, such that $U^TU = I_P$ and $V^TV = I_P$.

D is a $P \times P$ diagonal matrix,

$$\begin{bmatrix}d_1 & 0 & ... & 0 \\ 0 & d_2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & d_P\end{bmatrix} \tag{77}$$

where $d_1 \geq d_2 \geq ... \geq d_p \geq 0$ called the singular values of X. Thus, we can have the expression of Cor(X,X):

$$\begin{aligned} X^TX &= (UDV^T)^T(UDV^T) \\ &= VD^TU^TUDV^T \\ &= VD^2V^T \end{aligned} \tag{78}$$

For singular values, in most cases, the top amount of the value can occupy most of the sum of all singular values, so we can use the top k singular value and the corresponding portion of U and V to estimate the matrix X.

$$\begin{aligned} X_{N\times P} &= U_{N\times P} D_{P\times P} V^T_{P\times P} \\ &\approx U_{N\times k} D_{k\times k} V^T_{k\times P} \end{aligned} \tag{79}$$

So it will be similar to Cor(X,X):

$$\begin{aligned} (X^T X)_{P\times P} &= V_{P\times P} D^2_{P\times P} V^T_{P\times P} \\ &\approx V_{P\times k} D^2_{k\times k} V^T_{k\times P} \end{aligned} \tag{80}$$

Here,

$$D^2_{k\times k} = \begin{bmatrix} d_1^2 & 0 & ... & 0 \\ 0 & d_2^2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & d_k^2 \end{bmatrix} \tag{81}$$

But how could we know k?
Let's assume we want to estimate at least 90 percent of the real $X^T X$. We have:

$$\frac{\Sigma_{i=1}^k d_i^2}{\Sigma_{i=1}^P d_i^2} = 90\% \tag{82}$$

We want the sum of the top k of squared d can take up at least 90 percent of the sum for all the d in the original $D^2_{P\times P}$ matrix, this way, we are giving up the P-k least influenced components.

Here are two questions:

- How to get $\Sigma_{i=1}^P d_i^2$?

  We can use $(N-1)\times P$ to predict? (The sum of squared singular values $\Sigma_{i=1}^P d_i^2$ is the squared Frobenius norm for matrix D, or: $\Sigma_{i=1}^P d_i^2 = ||D||_F^2 = \Sigma_i^N \Sigma_j^P |d_{ij}|^2$)

- How to get $\Sigma_{i=1}^k d_i^2$?

  Since $V^T V = I_P$, we have:

$$\begin{aligned} \Sigma_{i=1}^k [d_i^2] &= \Sigma_{i=1}^k [V_i^T V_i d_i^2] \\ &= \Sigma_{i=1}^k [V_i^T d_i^2 V_i] \\ &= \Sigma_{i=1}^k [(V_i^T d_i)(d_i V_i)] \\ &= \Sigma_{i=1}^k [(d_i V_i)^T (d_i V_i)] \end{aligned} \tag{83}$$

  This means that we only need the first k column from the matrix $\begin{bmatrix} d_1\vec{V_1} & d_2\vec{V_2} & ... & d_P\vec{V_P} \end{bmatrix}$ to calculate the percentage we want.

Now, look back at equation (79), if we are estimating X^TX in this way, what matrix we are expected to provide?

$$\begin{aligned}
(X^TX)_{P\times P} &= V_{P\times P}D^2_{P\times P}V^T_{P\times P} \\
&\approx V_{P\times k}D^2_{k\times k}V^T_{k\times P} \\
&= V_{P\times k}\begin{bmatrix} d_1^2 & 0 & ... & 0 \\ 0 & d_2^2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & d_k^2 \end{bmatrix} V^T_{k\times P} \\
&= \begin{bmatrix} \vec{v_1} & \vec{v_2} & ... & \vec{v_k} \end{bmatrix} \begin{bmatrix} d_1^2 & 0 & ... & 0 \\ 0 & d_2^2 & ... & 0 \\ ... & ... & ... & ... \\ 0 & 0 & ... & d_k^2 \end{bmatrix} \begin{bmatrix} \vec{v_1} \\ \vec{v_2} \\ ... \\ \vec{v_k} \end{bmatrix} \\
&= \begin{bmatrix} (d_1\vec{v_1})^2 & (d_2\vec{v_2})^2 & ... & (d_k\vec{v_k})^2 \end{bmatrix}
\end{aligned} \tag{84}$$

This seems like we also need the first k column from $\begin{bmatrix} d_1\vec{V_1} & d_2\vec{V_2} & ... & d_P\vec{V_P} \end{bmatrix}$ to estimate X^TX.

4.6 Summary

So far, to apply a model estimation, the minimum information we are expected to have are:

- $\vec{r}$, or in another way, X^Ty, this is reported as summary statistics. We also can get the feature amount P from this.
- N, means the number of study objects, or observation number.
- $\begin{bmatrix} d_1\vec{V_1} & d_2\vec{V_2} & ... & d_k\vec{V_k} \end{bmatrix}$, this is a $P \times k$ matrix where we can estimate X^TX for a certain accuracy percentage, depending on the choice of k.

Something needs to pay attention to:

In real-life research of **brain-imaging or genetic-related studies**, the estimation of $r = X^Ty$, which represents the correlation between the observations and phenotype, is generally from summary statistics databases that are publicly available for major diseases/phenotypes.

However, for the estimation of $R = X^TX$, which is a matrix of correlations between features/genotypes, if we also get the estimation of R from another publicly available genotype database, the genotype **X** used to estimate **R** and **r** will, in general, be different. This is different from other **macroscopic** statistics linear relation studies: for example, if there are two studies all from the Asian research center, and there....

In this paper `https://pubmed.ncbi.nlm.nih.gov/28480976/`, it gave a solution that they used $R = X_r^TX_r$, and regularize it/ In particular, is letting $R_s = (1-s)X_r^TX_r + sI$, for some $0 < s < 1$.