Note for DS2023s (Machine Learning Section)

Topic 1: Linear Regression 一般线性回归

CASE: $n \gg p$

$$\hat{\beta}_{OLS} = (X'X)^{-1}X'y$$

- When $n \gg p$, $X'X \in \mathbb{R}^{p \times p}$, $X'y \in \mathbb{R}^{p \times 1}$, thus it is appliable to calculate in the memory
- The key is to **calculate the two matrix multiplication above （重点即为求解两矩阵乘法**，最后再求解一个线性方程组即可）
- Note that, to maximize the calculation performance, we can combine the calculation into one（为方便计算，常用如下方式整合矩阵）：
  $X'[X, y]$
- After obtaining the two matrix, at last we have to solve the linear equation to get $\hat{\beta}$ (Remember that we should always try to avoid calcualte the inverse of a matrix)
- Moreover, if we have to consider the intercept, we can transform the data matrix （若考虑截距项）：
  $X^* = [1, X]$

Note: *TRY TO REALIZE THE PROCESS USING PYTHON WITH RDD*

Topic 2: Regularization (Shrinkage Methods) 正则化（数据缩减技术）

CASE: $n < p$

- When $n < p$, $X'X$ is not reversible. In this case, OLS **DOES NOT HAVE UNIQUE SOLUTION**
  - Note that it does not mean that there is no solution! By contrast, there are infinite sets of solutions that can hold $\min S_c = 0$ (you can think of it as there are more variables than equations, so the variables might have many solutions)（此时OLS并非无解，而是有无数解）
- Such case acutally will also accures when $n \approx p$
- In this case, OLS is too **flexible**, and the solution though perfectly fits the modle, will perform badly in test sets. It is called **OVERFIT** （**过拟合**，模型在测试集的表现欠佳)

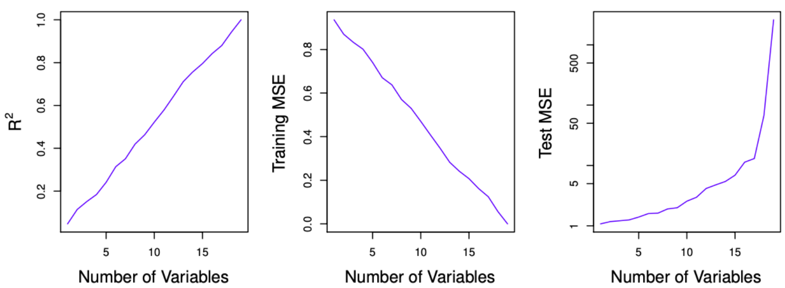

FIGURE 6.23. *On a simulated example with* $n = 20$ *training observations, features that are completely unrelated to the outcome are added to the model.* Left: *The* R^2 *increases to 1 as more features are included.* Center: *The training set MSE decreases to 0 as more features are included.* Right: *The test set MSE increases as more features are included.*

Method 1: Ridge Regression 岭回归

ESSENCE：ℓ_2 Regularization

- Recall in OLS, our loss function:
  $\text{Loss} = ||Y - X\beta||_2 = \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2$
- In Ridge Regression, we add an extra penalty:
  $\text{Loss}^* = ||Y - X\beta||_2 + \lambda||\beta||_2 = \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \sum_{j=1}^{p}\beta_j^2$
- We are going to find the optimum β that minimize that loss function
- Here $\lambda \geq 0$ is called the **Tunning Parameter**, $\lambda||\beta||_2$ is called **Shrinkage Penalty**
  - When $\lambda = 0$, ridge regression will degenereate into OLS （此时退化为普通最小二乘）; when $\lambda \to \infty$, the penalty term will squeeze out the β and approach to zero（此时各系数将趋近于0）
- The parameter estimation will relies on the selection of λ, i.e. $\hat{\beta}_\lambda^R$, and to choose a good λ is important. We can use methods such as Cross Valediction, etc.（可用交叉验证等手段选取一个最优的λ以得到较优的岭回归系数）
- Note that the ridge regression **DOES NOT INCLUDE** β_0（在岭回归中不会对截距项进行缩减）

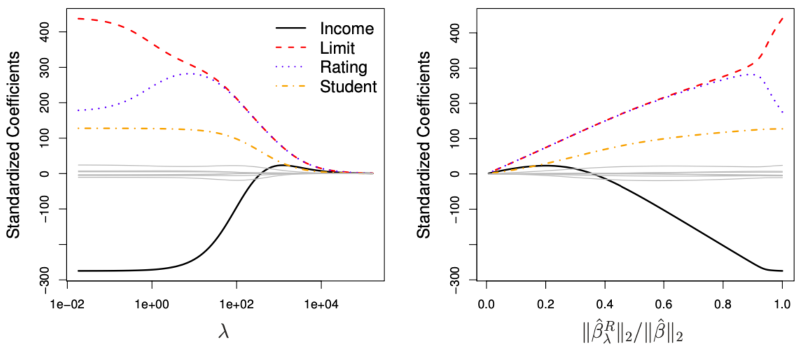

FIGURE 6.4. *The standardized ridge regression coefficients are displayed for the* Credit *data set, as a function of* λ *and* $\|\hat{\beta}_\lambda^R\|_2/\|\hat{\beta}\|_2$.

*As is shown in FIGURE 6.4, **Income, Limit, Rating & Student** are all variables' parameter estimation. From the lest graph, as λ increases, all estimation shrinks to zero. From the right graph, the ratio shows how different the ridge estimation and OLS estimation are (1 means no difference and 0 means utmost different)* （上图即展示了$lambda$对于参数估计的影响，其取值越大，参数约趋近于0，压缩越明显）

- In ridge regression, the final estimation of β is
  $\hat{\beta}_\lambda = (X^TX + \lambda I)^{-1}X^TY$
- It can be proved that $(X^TX + \lambda I)^{-1}$ must be inversible
  - First, it can be proved that for column full rank matrix X, X^TX is **positive semi-definite** (PSD)（首先证X^TX是半正定的）
    - $\forall a \in \mathbb{R}^{p\times 1}, y := Xa \neq 0$ (This is held by column full rank 由列满秩)
    - $a^TX^TXa := y^Ty = \sum y_i^2 \geq 0$
  - Second it can be proved that $X^TX + \lambda I$ is PD.
    - For a PSD, the eignalvalues $\alpha_i \geq 0$.
    - By adding λI whose eignvalues must be positive, the overall eignvalues $\Lambda > 0$, thus the matrix is PD （其次证$X^TX + \lambda I$是正定的）
  - Moreover, it can be proved that $X^TX + \lambda I$ is symmetric （再可证矩阵是对称的）

- $(X^TX+\lambda I)^T = \lambda I + X^TX$
  - Last, for a symmetric PD matrix, it is inversible （对称正定阵是可逆的）
    - Recall that the determinant of a matrix is equal to the product of the eignvalues
    - Since all eignvalues are greater than zero thus the determinant is greater than zero
- Also note that $(X^TX+\lambda I)^T$ though invertible, but in this case is a rahter large matrix which is hard to actually calculate the analytical value. In this case, we choose to use **Conjugate Gradient Method** （由于矩阵很大，求解逆矩阵的解析解是困难的，常常用共轭梯度法等求解数值解）

Method 2: LASSO

ESSENCE：ℓ_1 Regularization

- **Problem** of *Ridge Regression*:
  - Though ridge regression can squeeze the parameters to **CLOSE** to zero, it cannot set any to exact zero（岭回归永远无法真正将参数压缩至0）
  - It might be **acceptable for prediction accuracy**, but might cause **inconvenience in interpretation**, especially when p is large and we only need few of them
- In LASSO, the loss function is:
  $\text{Loss} = ||Y - X\beta||_2 + \lambda||\beta||_1$
- We can see that the only difference is the **penalty term**, Ridge regression uses a ℓ_2 norm penalty regularization, while LASSO uses a ℓ_1 norm penalty regularization （即通过1-范数和2-范数进行正则化）

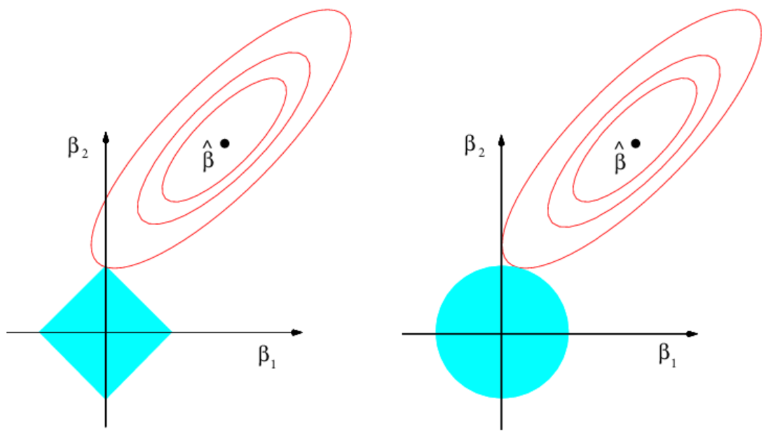

FIGURE 6.7. *Contours of the error and constraint functions for the lasso* (left) *and ridge regression* (right)*. The solid blue areas are the constraint regions, $|\beta_1| + |\beta_2| \leq s$ and $\beta_1^2 + \beta_2^2 \leq s$, while the red ellipses are the contours of the RSS.*

Topic 3: Logistic Regression

Essence: Modeling for the distribution of $Y \sim \text{Bernoulli}(\cdot)$

- In Logistic Regression, the final purpose is to answer a bunch of "YES or NO" "BELONGS TO or NOT BELONGS TO" problems. Such problems can be abstracted into a 0-1 questions, and there we have **Bernoulli Distribution**, which is extactly the one trying to demonstrate "how likely something will happen or not" （最终的目的是对Y的Bernoulli分布进行建模）

- For $Y_1, \ldots, Y_n$, they all follow some patterns of Bernoulli Distribution. But the parameters of Bernoulli distribution (i.e. p) are different for different Y's, which depends on the given information (variables X's)（Y服从Bernoulli分布是确定的，但并不是独立的；其分布的概率参数p依赖于给定的其他解释变量X的信息）
- We can describe such dependency as $p \propto f(X)$. To be more specific, here we assume that Y depends on the linear combinition of X, thus $p \propto X\beta$ （这种相依关系这里假设通过解释变量的线性组合进行刻画）
- Another problem is that $0 \leq p \leq 1$, but $X\beta$ literally can be any value, thus we need another bridging function to constrain the value to $[0, 1]$ （需要另外引入一个函数使得分布在区间范围内）
    - One possible solution: CDF
    - Another suggested solution: **Sigmoid** (it is the most commonly used, but not the only one) $\rho(x) = \frac{1}{1+e^{-x}}$
- Thus we have:
  $Y|x \sim \mathrm{Bernoulli}(\rho(\beta^T x))$
    - $\rho(\beta^T x)$ describes the possibility of $Y = 1$
- Here, the combinition patterns are still unknown. We want to have an optimum estimation to best fit the probabilities for all given Y's and X's. And here we have **Maximum Likelihood Estimation （通过极大似然函数以估计β）**
    - $P(Y_i = y) = p_i^y (1 - p_i)^y,\ y = 0 \text{ or } 1$
    - $l = \sum \log P(Y_i = y_i) = \sum [y_i \log p_i + (1 - y_i) \log(1 - p_i)]$, where $p_i = \rho(x_i \beta)$
    - Usually, set loss function as $L(\beta) = -l$ for minimization
    - Unfortunately, it is also hard to get the analytical solution. We also have to use iterations to calculate the numerical solution

Topic 4: Optimization Methods