Assignment 1, STK-IN 4300

He GU

heg@mail.uio.no

Problem 1

By using R, both lasso and ridge regression models could be easily implemented. And we are using a 5-fold cross-validation here to contrast the ability of both models.

CODE:

```
library(glmnet)

#get data
load("C:/Users/he322/Desktop/Oblig/STK/data_o1.rdata")
data <- data.frame(X, y)

#split data
n_random = sample(1:110,110)
n = c(sort(n_random[1:22]), sort(n_random[23:44]), sort(n_random[45:66]),
      sort(n_random[67:88]), sort(n_random[89:110]))

#setting range of parameters
lambdas <- 10^seq(-5, 5, by = 0.1)

err_ridge = 0
err_lasso = 0

#5-fold cv
for(i in 1:5){
    a <- 22*i-21
    b <- 22*i

    #tune parameters by 10-fold cv
```

```r
    cv_ridge <- cv.glmnet(X[-n[a:b], ], y[-n[a:b]], alpha = 0, lambda = lambdas)
    cv_lasso <- cv.glmnet(X[-n[a:b], ], y[-n[a:b]], alpha = 1, lambda = lambdas)
    lambda_ridge <- cv_ridge$lambda.min
    lambda_lasso <- cv_lasso$lambda.min

    #fit model by optimal parameters
    ridge_reg <- glmnet(X[-n[a:b], ], y[-n[a:b]], alpha = 0, lambda = lambda_ridge)
    lasso_reg <- glmnet(X[-n[a:b], ], y[-n[a:b]], alpha = 1, lambda = lambda_lasso)
    err_ridge <- err_ridge+ sum((predict(ridge_reg, newx = X[n[a:b],], s = lambda_ridge)-y[n[a:b]])^2)/22
    err_lasso <- err_lasso+ sum((predict(lasso_reg, newx = X[n[a:b],], s = lambda_lasso)-y[n[a:b]])^2)/22
}

err_ridge <- err_ridge/5
err_lasso <- err_lasso/5
err_ridge
err_lasso
```

RESULT:

```
> err_ridge
[1] 410.399
> err_lasso
[1] 414.8218
```

It seems that the Ridge regression performs slightly better than the Lasso regression.

We know that the Lasso regression shall reduce some parameters into 0, while the ridge regression shall reduce them into tiny numbers, which, however, are usually not 0. Thus, the result implies that although the influence of some variables could be quite small, but the influence does exist.

Problem 2

We see that we want to find an ω which minimizes the linearised expression

$$\sum_{i=1}^{N} g'(\omega_{old}^T x_i)^2 \left(\frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)} + \omega_{old}^T x_i - \omega^T x_i \right)^2$$

Obviously, this is equivalent with finding an ω such that

$$\omega = \arg\min_{\omega}\left[\sum_{i=1}^{N} g'(\omega_{old}^T x_i)^2 \left(\frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)} + \omega_{old}^T x_i - \omega^T x_i\right)^2\right]$$

With respect to ω, we see that the other terms could be considered as fixed. And thus, by using w_i to represent $g'(\omega_{old}^T x_i)^2$ and r_i to represent $\left(\frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)} + \omega_{old}^T x_i\right)$, the expression above could be transformed into a more clear one:

$$\begin{aligned}\omega &= \arg\min_{\omega}\left[\sum_{i=1}^{N} g'(\omega_{old}^T x_i)^2 \left(\frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)} + \omega_{old}^T x_i - \omega^T x_i\right)^2\right] \\ &= \arg\min_{\omega}\left[\sum_{i=1}^{N} w_i (r_i - \omega^T x_i)^2\right]\end{aligned}$$

It seems that we need to calculate the derivative $\frac{\partial\left(\sum_{i=1}^{N} w_i (r_i - \omega^T x_i)^2\right)}{\partial \omega}$ in order to determine the ω. And But it could be pretty hard to get the derivative in such case, and thus, we shall further transform it into a "matrix form" such that we can apply matrix calculus to it:

$$\omega = \arg\min_{\omega}\left[\sum_{i=1}^{N} w_i (r_i - \omega^T x_i)^2\right] = \arg\min_{\omega}[(\boldsymbol{r} - X\omega)^T W (\boldsymbol{r} - X\omega)]$$

where W is a diagonal matrix filled by w_i, $\boldsymbol{r}$ is the vector $(r_1, r_2, \ldots, r_n)$ and X is the "full matrix" filled with every x_i.

Now, by using the matrix calculus, we find out that

$$\begin{aligned}\frac{\partial\left(\sum_{i=1}^{N} w_i (r_i - \omega^T x_i)^2\right)}{\partial \omega} &= \frac{\partial\left((\boldsymbol{r} - X\omega)^T W (\boldsymbol{r} - X\omega)\right)}{\partial \omega} \\ &= \frac{\partial\left((\boldsymbol{r}^{\boldsymbol{T}} - \omega^T X^T) W (\boldsymbol{r} - X\omega)\right)}{\partial \omega} \\ &= \frac{\partial(\boldsymbol{r}^{\boldsymbol{T}} W \boldsymbol{r} - \boldsymbol{r}^{\boldsymbol{T}} W X\omega - \omega^T X^T W \boldsymbol{r} + \omega^T X^T W X \omega)}{\partial \omega} \\ &= 0 + (\boldsymbol{r}^{\boldsymbol{T}} W X)^{\boldsymbol{T}} - X^T W \boldsymbol{r} + X^T W X \omega + (X^T W X)^T \omega \\ &= -X^T W^T \boldsymbol{r} - X^T W \boldsymbol{r} + X^T W X \omega + X^T W^T X \omega\end{aligned}$$

And since as we defined before, W is a diagonal matrix, thus, this expression could be further transformed into

$$\frac{\partial\left(\sum_{i=1}^{N} w_i (r_i - \omega^T x_i)^2\right)}{\partial \omega} = -X^T W^T \boldsymbol{r} - X^T W \boldsymbol{r} + X^T W X \omega + X^T W^T X \omega$$

$$= -X^T W \boldsymbol{r} - X^T W \boldsymbol{r} + X^T W X \omega + X^T W X \omega$$

$$= -2X^T W \boldsymbol{r} + 2X^T W X \omega$$

Then, by solving $\frac{\partial\left(\sum_{i=1}^{N} w_i \left(r_i - \omega^T x_i\right)^2\right)}{\partial \omega} = 0$, we find out that

$$\omega = \arg\min_{\omega} \left[\sum_{i=1}^{N} g'(\omega_{old}^T x_i)^2 \left(\frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)} + \omega_{old}^T x_i - \omega^T x_i \right)^2 \right]$$

$$= (X^T W X)^{-1} X^T W \boldsymbol{r}$$

Now, we could conclude that

$$\omega = (X^T W X)^{-1} X^T W \boldsymbol{r}$$

where W is a diagonal matrix filled by w_i, $\boldsymbol{r}$ is the vector $(r_1, r_2, \ldots, r_n)$ and X is the "full matrix" filled with every x_i, and besides, we have $\begin{cases} w_i = g'(\omega_{old}^T x_i)^2 \\ r_i = \left(\frac{y_i - g(\omega_{old}^T x_i)}{g'(\omega_{old}^T x_i)} + \omega_{old}^T x_i \right) \end{cases}$.

________END________