HDS Exercise set 5.

YOUR NAME (STUDENT NUMBER)

Return by **10.15** o'clock on **Tuesday 3.12.2019** to the Moodle area of the course. Return the final file in pdf format with name "HDS5_yourname.pdf".

Hint: As running some solutions may take more time, you can use `{r, eval=F}` to skip running those solutions with every `Knit` while you are working on the other problems. But then remember to set `eval = T` before you compile your final solutions.

Problem 1.

Read in data "HDS_ex5_1.txt". There are two columns `x` and `y`, each having $n = 100$ values. We are interested in the mean of the product of `x` and `y`.

What is your estimate for the mean of the product of `x` and `y`?

Use bootstrap with $B = 1000$ replicates to estimate a standard error for your point estimate, as well as a 95% interval for the estimate.

Problem 2.

Read in "HDS_leukemia.txt". The data contain $p = 3571$ gene expression measurements on $n = 72$ leukemia patients, 47 of subtype ALL and 25 of subtype AML. These data are from study of Golub et al. (1999, Science) and were downloaded from https://web.stanford.edu/~hastie/CASI_files/DATA/leukemia.html.

The first column `y` tells the subtype (1=AML, 0=ALL). The task is to predict the subtype from gene expression. Therefore we use logistic regression.

(a)

Separate rows 71 and 72 as test data examples and use the samples 1,...,70 to train a ridge regression type of elastic net model (use $\alpha = 0.05$) and a LASSO type of elastic net model (use $\alpha = 0.95$) using `family="binomial"` to switch to logistic regression. Plot `cv.glmfit()` output for both models using the default measure (deviance) in CV. (Rely on the automatically chosen λ scale.)

Show the coefficient plots on the deviance scale. Find out from `cv.glmnet()` object's field `glmnet.fit` what is the deviance ratio corresponding to the chosen λ at `lambda.1se` and denote that ratio by a vertical line in the coefficient plots.

Make predictions for the two test data samples on probability scale. Would you classify them correctly?

(b)

Repeat the analysis of part (a) except use missclassification rate (`class`) as the error measure in CV instead of deviance. Do the results differ from part (a)? By looking at the CV plots, which measure (deviance or missclassification rate) do you expect to be more robust to small changes in data?

Problem 3.

Continue with "HDS_leukemia.txt" from Problem 2. The task is to assess uncertainty of the chosen model by bootstrap.

Use whole data of $n = 72$ individuals and $p = 3571$ genes to predict leukemia subtype `y` ($1 =$ AML, $0 =$ ALL), by using logistic regression with LASSO penalty ($\alpha = 1$) and `deviance` as the MSE criterion to choose the LASSO model at `lambda.1se`. Do bootstrap with $B = 100$ replications to evaluate the probabilities that each coefficient is non-zero. (You can interpret that a coefficient is zero if its absolute value is below 1e-10.)

Which coefficient are non-zero with a probability over 70%?

Problem 4.

We saw in lectures that ridge regression can remove unstable estimates that arise from highly correlated predictors.

(a) Write a function that generates a pair of correlated variables for a given sample size n and correlation $r \in [-1, 1]$. (You are free to use lecture material.) Check that your function works by using $n = 10,000$ and computing the correlation between the output variables for one application of the function for each of the values of $r = -0.9, -0.5, 0, 0.5, 0.9$. (Large n here is to make estimate precise whereas for small n a discrepancy from target r could be also because of a chance effect.)

(b) Using $n = 200$ and for each value of $r = 0.0, 0.99, 0.999$ generate $R = 50$ pairs of variables $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ with approximate correlation of r and for each pair generate an outcome variable $\boldsymbol{y} = \boldsymbol{x}_1 0.5 + \boldsymbol{\varepsilon}$ where $\varepsilon_i \sim \mathcal{N}(0, 1)$ for each individual i independently. Collect the effect size estimates of $\widehat{\beta}_1$ and $\widehat{\beta}_2$ when you fit the model $y = \beta_0 + x_1\beta_1 + x_2\beta_2 + \varepsilon$ using ridge regression and LASSO. Use `cv.glmnet()` and take the model at `lambda.min`. Make 3 scatter plots of $\widehat{\beta}_1$ vs. $\widehat{\beta}_2$ for each method (e.g. in 2x3 area), where one plot corresponds to one value of r.

Which method is more robust in the sense that estimates do not vary widely across data sets with same value of r? How do the methods estimate the effects compared to the true values?

Problem 5.

Let's see how sample size and sparsity affect LASSO.

Consider model $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where number of predictors (columns of $\boldsymbol{X}$) is $p = 100$. For each combination of $n \in \{100, 200, 500\}$ (rows of $\boldsymbol{X}$) and $p_0 \in \{1, 25, 50, 75, 99\}$ make $R = 10$ data sets where each predictor is sampled from $\mathcal{N}(0, 1)$ and the coefficients (β) for p_0 predictors are 0 and for the remaining $p - p_0$ predictors the coefficient is randomly sampled (50%:50% binomial sampling) from among two possibilities: $\{-1, 1\}$. Generate the errors ε from distribution $\mathcal{N}(0, p - p_0)$, i.e, the variance of errors is $p - p_0$ which is the same variance that is contributed by the predictors. (Remember that in R `rnorm()` function takes in SD not variance!) Consequently, the predictors explain 50% of the variance of the outcome Y. Finally, standardize both Y and all columns of X for easier interpretation.

For each of 3x5x10=150 data sets, apply `cv.glmnet()` with `alpha=1` (LASSO) and collect the following 4 values from the output `cvm`, `cvlo`, `cvup` and `nzero` at value `lambda = lambda.1se`. Average each of these 4 variables over R data sets for each combination of parameters, so in the end you will have 15 sets of the 4 variables. In all plots that follow, use the averaged values over R replicates, not the values from the individual replicates.

Figure 1: Plot proportion of zero coefficient from LASSO (you can derive this from the variable `nzero`) as a function of the true proportion of zero coefficients in data (p_0/p). Make 3 curves in the same plot corresponding to 3 values of n. How does LASSO's estimate of number of zero coefficients behave as n grows?

Figure 2: Plot 3 curves showing estimated CV'd MSE (`cvm`) as a function of true p_0/p and the different curves correspond different values of n. Add also the error bars (from `cvlo` to `cvup`) around the point estimates, for example, by using `arrows()`.

Hint: Try to avoid repeating code by collecting results e.g. to an `array` of dimension 3x5x4 (n x p0 x 4 averaged results), by using 3 nested for-loops to go through the combinations of parameters n and p_0, and $R = 10$ data sets.