NORTHWESTERN
UNIVERSITY

SCHOOL OF
PROFESSIONAL
STUDIES

COURSE DETAILS

Instructor Name:	Dr. Nathan Bastian (Prof. B)
Program Name:	Master of Science in Predictive Analytics (MSPA)
Course Name:	Practical Machine Learning
Course/Section Number:	PREDICT 422-DL-56

PROJECT DESCRIPTION

In this individual project, you will use the diabetes data in Efron et al. (2003) to examine the effects of ten baseline predictor variables [age, sex, body mass index (bmi), average blood pressure (map), and six blood serum measurements (tc, ldl, hdl, tch, ltg, glu)] on a quantitative measure of disease progression one year after baseline. There are 442 diabetes patients in this data set. The data are available in the R package "lars." You must employ several machine learning techniques using the diabetes data to fit linear regression, ridge regression and lasso models. You must also incorporate best subset selection and cross-validation techniques.

```
# Load the diabetes data
library(lars)
data(diabetes)
data.all <- data.frame(cbind(diabetes$x, y = diabetes$y))

# Partition the patients into two groups: training (75%) and test (25%)
n <- dim(data.all)[1]                 # sample size = 442
set.seed(1306)                        # set random number generator seed to enable
                                      # repeatability of results
test <- sample(n, round(n/4))         # randomly sample 25% test
data.train <- data.all[-test,]
data.test <- data.all[test,]
x <- model.matrix(y ~ ., data = data.all)[,-1] # define predictor matrix
                                               # excl intercept col of 1s
x.train <- x[-test,]                           # define training predictor matrix
x.test <- x[test,]                             # define test predictor matrix
y <- data.all$y                                # define response variable
y.train <- y[-test]                            # define training response variable
y.test <- y[test]                              # define test response variable
n.train <- dim(data.train)[1]                  # training sample size = 332
n.test <- dim(data.test)[1]                    # test sample size = 110
```

PROJECT REQUIREMENTS:

Fit the following models to the *training* set. For each model extract the model coefficient estimates from training set (not from re-running on the full data set), predict the responses for the *test* set, and calculate the "mean prediction error" (and its standard error) in the *test* set.

1. Least squares regression model using all ten predictors (R function lm).
2. Apply best subset selection using *BIC* to select the number of predictors (R function regsubsets in package leaps).
3. Apply best subset selection using *10-fold cross-validation* to select the number of predictors (R function regsubsets in package leaps). [Use a random number seed of 1306 before entering the command: folds <- sample(1:k, nrow(data.train), replace = TRUE).]
4. Ridge regression modeling using *10-fold cross-validation* to select the largest value of λ such that the cross-validation error is within 1 standard error of the minimum (R functions glmnet and cv.glmnet in package glmnet). [Use a random number seed of 1306 immediately before entering the command: cv.out <- cv.glmnet(x.train, y.train, alpha = 0).]
5. Lasso model using *10-fold cross-validation* to select the largest value of λ such that the cross-validation error is within 1 standard error of the minimum (R functions glmnet and cv.glmnet in package glmnet). [Use a random number seed of 1306 immediately before entering the command: cv.out <- cv.glmnet(x.train, y.train, alpha = 1).]

PROJECT REPORT:

- The report should be no more than 4 single-spaced pages long.
- It should include coefficient estimates for each model, the test data mean prediction errors (i.e., test MSE), and the standard errors of the mean prediction errors.
- Include any other details from your analysis that you feel are worthy of mention.
- The report should have sections (e.g., Introduction, Analysis, Results, Conclusion) and provide sufficient details that anyone with a reasonable statistics background could understand what you've done.
- Consider using tables and figures to enhance your report.
- *Do not* embed R code in the body of your report, but instead attach the code in an appendix. The appendix does *not* count towards the page limit.

GRADING CRITERIA:

- You can receive up to **60 points** for this assignment.
- 50 points for fulfilling the project requirements and matching the instructor's results exactly (this is why you have to use the specific random number generator seeds above).
- 10 points for the quality of the report (including: clarity of writing, organization, layout, etc.)