Line H. Clemmensen
Section of Statistics and Data Analysis
DTU Compute

Exercises 02582
Module 3
Spring 2018

February 14, 2018

Topics: Least angle regression selection (LARS), Elastic net, Multiple testing

Resources for this exercise:

Listing 1: Resources in Matlab

```
sand.mat % sand data set (X: 59x2016)
center(y) % subtract mean from y
normalize(X) % subtract mean and divide by s.d. for each column of X
normalizetest(Xtst, m, s) % normalize using training mean m and s.d. s
elasticnet(X,y) % estimates elasticnet solution
lme = fitlme(T, 'Y~Var2'); % fit univariate model
lme.coefTest; % extract p-value for test of coefficient
FDR = mafdr(PValues, 'BHFDR', 'True'); % compute Benjamini Hochberg's
        % FDR rates
```

Listing 2: Resources in R

```
library(lars)
library(elasticnet)
library(cvTools)
library(R.matlab) # to be able to load .mat file in R
dat <- readMat(file.path('sand.mat')) # sand data set (X: 59x2016)
Ytrain = scale(Y,scale=F); # center y train (subtract mean from y train)
Xtrain = scale(X); # standardize (subtract mean and divide by s.d.)
Ytst = Ytst - mean(Ytr); # use the mean value of the training response
        # to center y test
Xtst = scale(Xtst, center=mean(Xtr)*matrix(1,dim(Xtst)[2]),
scale=sd(Xtr)*matrix(1,dim(Xtst)[2])); # normalize test using
        # training mean and s.d.
LAR <- lars(xtr, ytr, trace = F, type = "lar", normalize = F,
        intercept = F, use.Gram = F) # Build LARS model
ytsthat <- predict(LAR, xtst,s = index)$fit # fit LARS model
fit=cv.glmnet(X, Y,alpha=a,  type.measure = "mse", nfolds = 5,
        standardize = T, intercept = T) # elastic net model
fm1=lm(formula = Y ~ X[,j], data = YX) # fit linear model
p.adjust(p, method = "bonferroni", n = length(p)) # adjust p-values
        # using Bonferroni
p.adjust(p, method = "BH", n = length(p)) # adjust p-values
        # using Benjamini-Hochberg
```

Listing 3: Resources in Python

```
import scipy.io
import numpy as np
from sklearn import linear_model
from scipy import linalg
from sklearn import preprocessing
import matplotlib.pyplot as plt
import matplotlib.colors as colors
from scipy.stats import linregress
from statsmodels.sandbox.stats.multicomp import multipletests

# HelpFctsNormalize.py contains functions to
        # center, normalize and normalizetest

mat = scipy.io.loadmat(path + 'sand.mat') # sand data set (X: 59x2016)
reg = linear_model.Lars(n_nonzero_coefs=j, fit_path = False,
        fit_intercept = False, verbose = True) # LARS model
reg_elastic = linear_model.ElasticNet(alpha = _lambda,
        l1_ratio = ratio/10, fit_intercept = False)
                # Elastic net model
reg_elastic.fit(Xtrain, ytrain) # fit model
beta = reg_elastic.coef_ # extract betas
slope, intercept, r_value, PValues[j], std_err = linregress(Xsub, y)
        # linear regression
FDR = multipletests(PValues, alpha = 0.05, method = "fdr_bh")[1]
        # Computing Benjamini Hochberg's FDR
```

Line H. Clemmensen
Section of Statistics and Data Analysis
DTU Compute

Exercises 02582
Module 3
Spring 2018

February 14, 2018

Topics: Least angle regression selection (LARS), Elastic net, Multiple testing

Exercises:

1 Apply Least angle regression and selection (LARS) for the $p >> n$ sand data set ($\mathbf{X}$: data matrix with 59 observations and 2016 features, $\mathbf{y}$: the measured moisture content in percent for each sand sample). Find a suitable solution using:

  (a) The Cp statistic. Consider whether the C_p-statistic makes sense in this case ($p > n$). Why? Why not?

  - Hint: What happens to your estimate of the noise in the data?

  (b) Using Cross-validation. Remember to center $\mathbf{y}$ and normalize $\mathbf{X}$, but do it inside the cross validation!

  - In Matlab: help functions provided to this are: `center.m`, `normalize.m`, `normalizetest.m`

2 Find an elastic net solution for the sand data, with suitable choices of regression parameters using cross validation.

  (a) Use the coordinate descent algorithm.

  - Matlab: Use Matlab's `lasso` function.
  - R: Use R's `glmnet`.
  - Python: Use Python's `linear_model.ElasticNet`.

  (b) Investigate how different values of α affects the number of nonzero parameters in the coordinate descent algorithms.

(c) What are the pros and cons of the coordinate descent algorithm compared to using LARS?

3 Perform univariate feature selection for the sand data using:

(a) Bonferroni correction to control the family-wise error rate (FWER). Use $FWER = 0.05$.

(b) Benjamini-Hochberg's algorithm for FDR. Use an acceptable fraction of mistakes, $q = 0.15$.

Compare the solutions in terms of number of selected features and selected features. Hint: See the resources for implementations of Benjamini-Hochberg's algorithm.

End of exercise