Econ 142 PSET 10

Sofia Guo

4/22/2019

```r
#load libraries
library(dplyr)
library(magrittr)
library(ggplot2)
library(glmnet)
library(stargazer)
library(reshape2)
library(kableExtra)
library(qpcR) #for RMSE function
library(psych)   # for function tr() to compute trace of a matrix
```

I. Data Exploration & Preparation

```r
#read in dataset
rehosp <- read.csv("/Users/sofia/Box/Cal (sofiaguo@berkeley.edu)/2018-19/Spring 2019/Econ 142/PSETS/PSE
```

We examine the dataset first with summary statistics, checking for any incomplete variables (next page).

```r
#descriptive statistics table
stargazer(rehosp, type='latex', header = F, title='Summary statistics', flip=T, digits = 2)
```

Table 1: Summary statistics

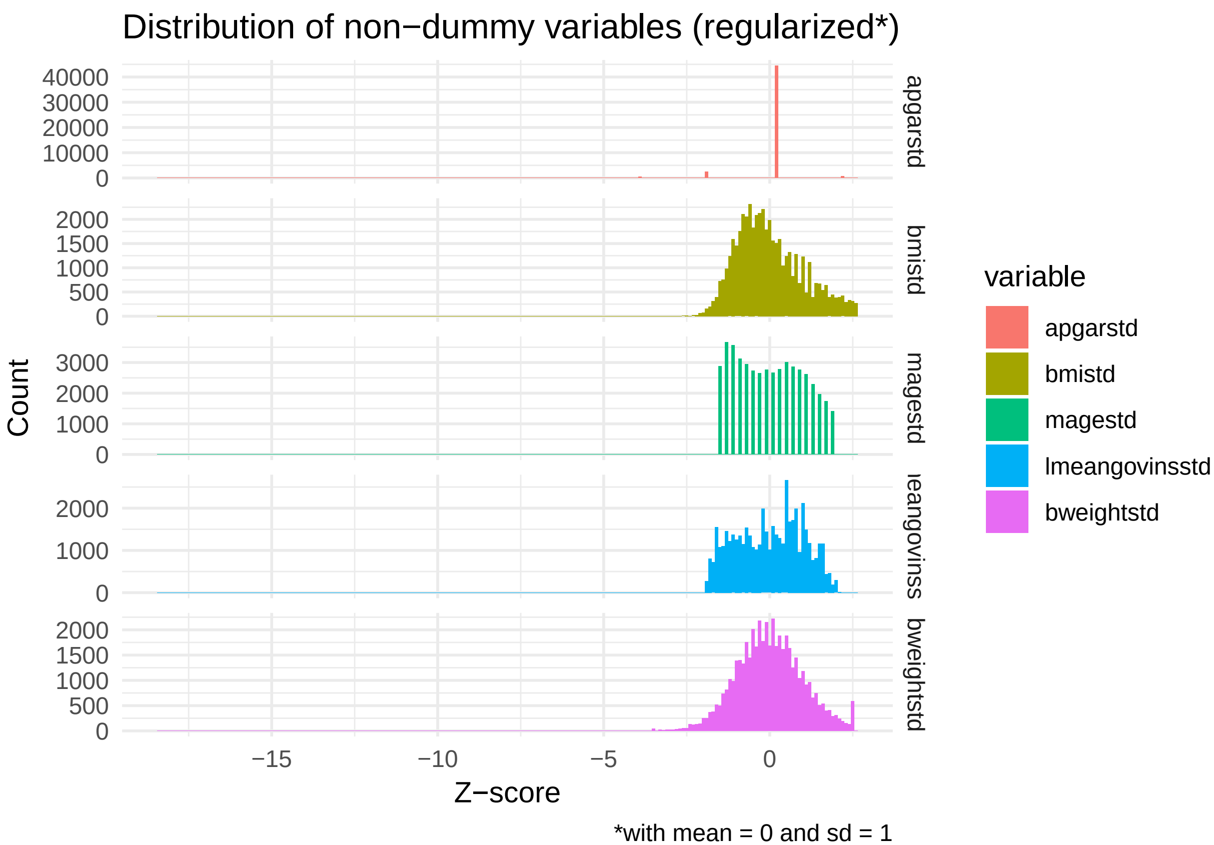
Statistic	apgar5	mage	lmean_govins	mom_dropout	mom_hs	mom_somecoll	mom_college	bweight	hospital	bmi
N	48,470	48,871	48,871	48,871	48,871	48,871	48,871	48,871	48,871	48,871
Mean	8.91	25.63	0.45	0.14	0.27	0.26	0.33	3,347.04	0.08	23.62
St. Dev.	0.49	4.98	0.23	0.35	0.44	0.44	0.47	434.37	0.27	3.85
Min	0.00	18	0.00	0	0	0	0	1,840	0	11.35
Pctl(25)	9.00	21	0.26	0	0	0	0	3,060	0	20.78
Pctl(75)	9.00	30	0.63	0	1	1	1	3,629	0	25.97
Max	10.00	35	1.00	1	1	1	1	4,441	1	33.83

We see that apgar5 is missing 401 observations; this means our dataset is $N = 48470$ large. First, we drop all NA values, then take a look at the distribution of each non-dummy variable so we can get a sense of what our dataset looks like:

```r
#standardize all the non-dummy variables to mean 0 and SD 1
rehosp_std <- rehosp %>%
  na.omit() %>%
  mutate(apgarstd = (apgar5 - mean(apgar5))/sd(apgar5),
    bmistd = (bmi - mean(bmi))/sd(bmi),
         magestd = (mage - mean(mage))/sd(mage),
        lmeangovinsstd = (lmean_govins - mean(lmean_govins))/sd(lmean_govins),
        bweightstd = (bweight - mean(bweight))/sd(bweight)) %>%
  dplyr::select(hospital, mom_dropout, mom_hs, mom_somecoll, mom_college, apgarstd, bmistd, magestd, lmeangovinsstd, bweightstd)

rehosp_melt <- rehosp_std %>%
  dplyr::select(apgarstd, bmistd, magestd, lmeangovinsstd, bweightstd) %>%
  melt()

ggplot(rehosp_melt, aes(x=value, fill=variable)) +
  geom_histogram(binwidth = 0.1) +
  labs(y = 'Count', x = 'Z-score',
       title = 'Distribution of non-dummy variables (regularized*)',
       caption = "*with mean = 0 and sd = 1") +
  facet_grid(variable~., scales="free") +
  theme_minimal()
```



Immediately, we see that most of the variables are pretty normally distributed except the apgar scores (with a peak at a little above the mean, and several observations beyond -2 SD away, and more than a few actually a maximum -18 SD away due to the very high density at a high mean). This makes sense because the mean of the apgar scores would be very high given that these babies lived beyond their birth. Thus, it makes sense to take a shrinkage approach on our dataset, such that our predictive ability improves (we can use apgar as a predictor without worrying about its bias).

One way to deal with the unevenness across apgar scores is to shrink/reweight the dataset, using apgar scores as groups. From lecture, we know that the Ridge regression is one way to "reweight" data by group such that our predictions aren't biased towards the 8-9 apgar score outcomes. Or, we may find that the apgar scores are not good predictors because of this bias and exclude it from the regression. Either way, we will first examine the data for this trend, if any.

II. Preliminary model

First, we will run the OLS model with all regularized and dummy predictors, to see what the model deems significant:

```r
#kitchen sink OLS
OLS <- lm(hospital ~ magestd + mom_dropout + mom_hs + mom_somecoll + mom_college + lmeangovinsstd + bmis
```

```r
#OLS table
stargazer(OLS,
          type='latex', header = F, title='OLS', flip=T, digits = 2, multicolumn = F,
          font.size = "small",
          column.sep.width = '1pt')
```

Table 2: OLS

	Dependent variable:
	hospital
magestd	-0.01^{***}
	(0.002)
mom_dropout	0.02^{***}
	(0.005)
mom_hs	0.01^{*}
	(0.004)
mom_somecoll	-0.0002
	(0.004)
mom_college	
lmeangovinsstd	0.01^{***}
	(0.001)
bmistd	0.001
	(0.001)
apgarstd	-0.005^{***}
	(0.001)
bweightstd	-0.01^{***}
	(0.001)
Constant	0.08^{***}
	(0.003)
Observations	48,470
R^2	0.004
Adjusted R^2	0.004
Residual Std. Error	0.27 (df = 48461)
F Statistic	26.56^{***} (df = 8; 48461)
Note:	*p<0.1; **p<0.05; ***p<0.01

It seems like mom_college and mom_somecoll are not very significant. This makes sense as both are sort of repetitive measures of educational attainment, given that high school completion and means tested benefits are probably more serious indicators of the mother's income. In addition, mothers' bmi is not significant, which after thinking about the probability of readmittance to the hospital for the infant is puzzling. A huge potential factor of readmittance may have to do with the health of the mother, such as ability to nuture or feed the child based on the mother's physical health. Our OLS results suggest that bmi is not a good measure of this health factor, and it seems that actually apgar scores do matter in addition to birthweight and mothers' age.

Let's try our ridge model, knowing this initial model outcome and see what happens.

III. Refining the model

We want to shrink the coefficients that are too large caused by the non-normal distribution of data as shown in the first graph. We use the following model and minimize using the ridge regression objective following it:

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_K x_{Ki} + \epsilon_i$$

$$min\beta \sum_i (y_i - \beta_0 - \beta_1 x_{1i} - \cdots - \beta_K x_{Ki})^2 + \lambda \sum_{j=1}^{K} \beta_j^2$$

To run the ridge regression, we try an example (using the tutorial from the url printed below):

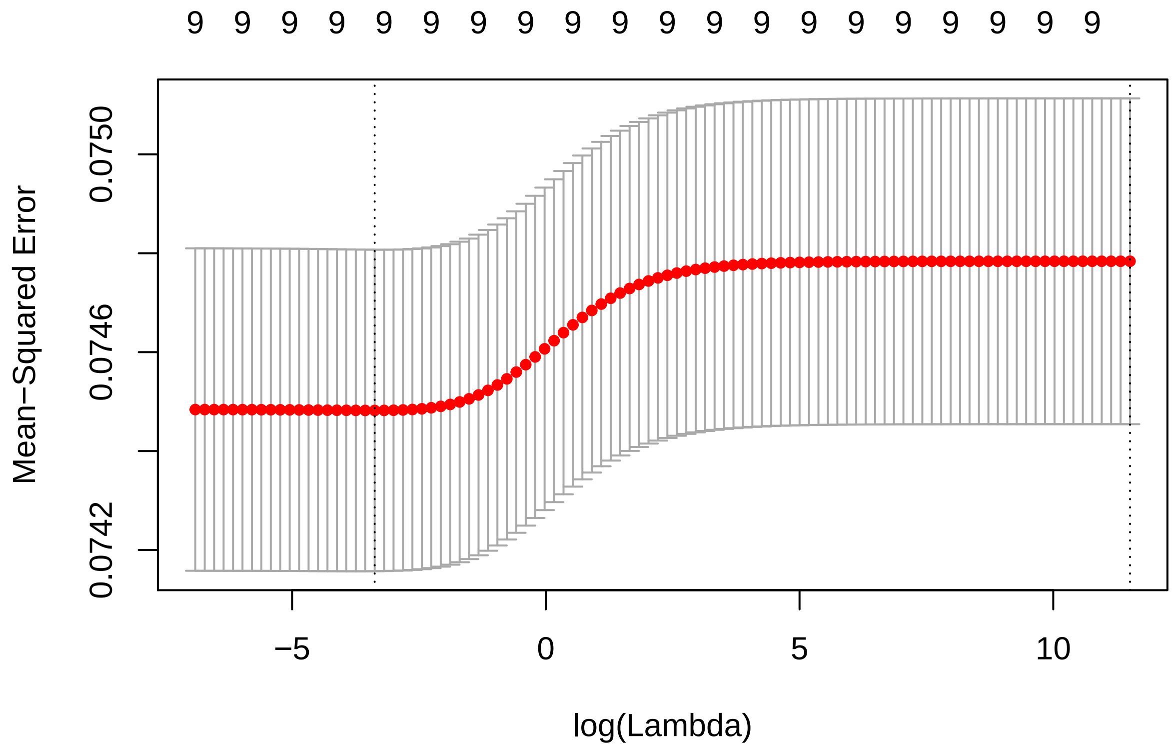
```r
#from https://www.datacamp.com/community/tutorials/tutorial-ridge-lasso-elastic-net

# Load libraries, get data & set seed for reproducibility ---------------------
set.seed(123)    # seef for reproducibility

# Center y, X will be standardized in the modelling function
y <- rehosp_std %>% dplyr::select(hospital) %>% as.matrix()
X <- rehosp_std %>% dplyr::select(-hospital) %>% as.matrix()


# Perform 5-fold cross-validation to select lambda ---------------------------
lambdas_to_try <- 10^seq(-3, 5, length.out = 100)
# Setting alpha = 0 implements ridge regression
ridge_cv <- cv.glmnet(X, y, alpha = 0, lambda = lambdas_to_try,
                      standardize = TRUE, nfolds = 5)
```

```r
# Plot cross-validation results
plot(ridge_cv)
```




```
# Best cross-validated lambda
lambda_cv <- ridge_cv$lambda.min
lambda_cv
```

```
## [1] 0.03430469
```

We use our chosen lambda using 5 fold cross validation to fit the final ridge model:

```
# Fit final model, get its sum of squared residuals and multiple R-squared
model_cv <- glmnet(X, y, alpha = 0, lambda = lambda_cv, standardize = TRUE)
y_hat_cv <- predict(model_cv, X)
RMSE_cv <- sqrt(mean((y_hat_cv - y)^2))
```

```
#look at the coefficients
coef(model_cv)
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                             s0
## (Intercept)      0.0814295469
## mom_dropout      0.0148418714
## mom_hs           0.0030050841
## mom_somecoll    -0.0045951582
## mom_college     -0.0053335238
## apgarstd        -0.0041792702
## bmistd           0.0007779401
## magestd         -0.0056134160
## lmeangovinsstd   0.0061620432
## bweightstd      -0.0063080169
```

```
#compare the RMSE's
RMSE_cv
```

```
## [1] 0.2728714
```

```r
RMSE(OLS)
```

```
## [1] 0.272868
```

Overall it looks pretty good with comparable RMSE to the OLS model (albeit a little bit smaller). Thus, let's run the smaller RMSE model (Ridge) on the test set to see the predictive ability.

IV. Testing the model

```r
#read in the test set
test <- read.csv("/Users/sofia/Box/Cal (sofiaguo@berkeley.edu)/2018-19/Spring 2019/Econ 142/PSETS/PSET
```

```r
#descriptive statistics table
stargazer(test, type='latex', header = F, title='Test Set Summary Statistics', flip=T, digits = 2)
```

Table 3: Test Set Summary Statistics

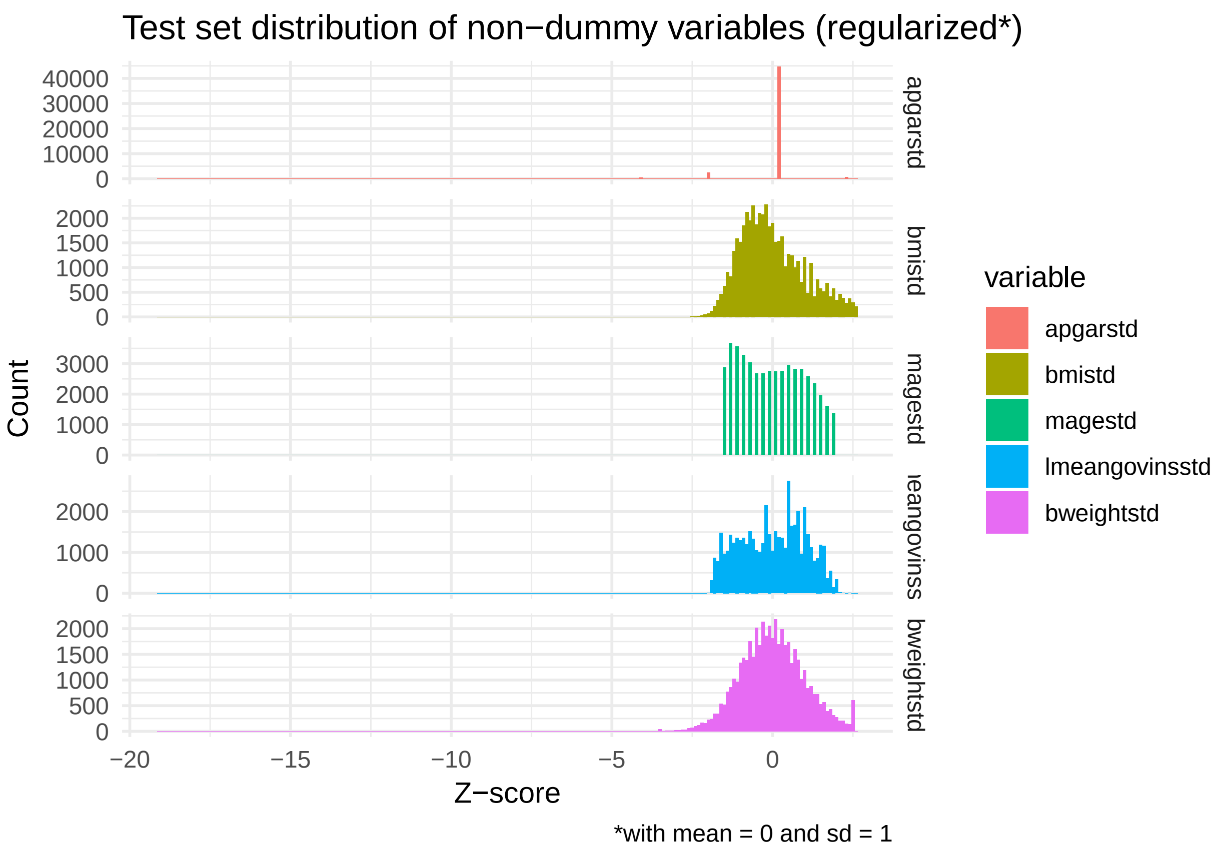
Statistic	apgar5	mage	lmean_govins	mom_dropout	mom_hs	mom_somecoll	mom_college	bweight	hospital	bmi
N	48,571	48,959	48,959	48,959	48,959	48,959	48,959	48,959	48,959	48,959
Mean	8.92	25.58	0.45	0.14	0.27	0.26	0.33	3,346.03	0.08	23.61
St. Dev.	0.47	4.97	0.23	0.35	0.45	0.44	0.47	432.64	0.28	3.88
Min	0.00	18	0.00	0	0	0	0	1,840	0	9.12
Pctl(25)	9.00	21	0.26	0	0	0	0	3,060	0	20.72
Pctl(75)	9.00	30	0.63	0	1	1	1	3,629	0	25.98
Max	10.00	35	1.00	1	1	1	1	4,441	1	33.83

We see that apgar5 is missing 388 observations; this means our dataset is $N = 48571$ large. First, we drop all NA values, then take a look at the distribution of each non-dummy variable so we can get a sense of what our dataset looks like:

```
#standardize all the non-dummy variables to mean 0 and SD 1
test_std <- test %>%
  na.omit() %>%
  mutate(apgarstd = (apgar5 - mean(apgar5))/sd(apgar5),
    bmistd = (bmi - mean(bmi))/sd(bmi),
         magestd = (mage - mean(mage))/sd(mage),
        lmeangovinsstd = (lmean_govins - mean(lmean_govins))/sd(lmean_govins),
        bweightstd = (bweight - mean(bweight))/sd(bweight)) %>%
  dplyr::select(hospital, mom_dropout, mom_hs, mom_somecoll, mom_college, apgarstd, bmistd, magestd, lme

test_melt <- test_std %>%
  dplyr::select(apgarstd, bmistd, magestd, lmeangovinsstd, bweightstd) %>%
  melt()

ggplot(test_melt, aes(x=value, fill=variable)) +
  geom_histogram(binwidth = 0.1) +
  labs(y = 'Count', x = 'Z-score',
       title = 'Test set distribution of non-dummy variables (regularized*)',
       caption = "*with mean = 0 and sd = 1") +
  facet_grid(variable~., scales="free") +
  theme_minimal()
```




```
#use predict function to
X_test <- test_std %>% dplyr::select(-hospital) %>% as.matrix()
```

```r
y_test <- test_std %>% dplyr::select(hospital) %>% as.matrix()

# Fit final model, get its sum of squared residuals and multiple R-squared
model_test <- glmnet(X_test, y_test, alpha = 0, lambda = lambda_cv, standardize = TRUE)
y_hat_test <- predict(model_test, X_test)
RMSE_test <- sqrt(mean((y_hat_test - y_test)^2))
```

```r
#compare all the coefficients
coef(model_cv)
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                          s0
## (Intercept)     0.0814295469
## mom_dropout     0.0148418714
## mom_hs          0.0030050841
## mom_somecoll   -0.0045951582
## mom_college    -0.0053335238
## apgarstd       -0.0041792702
## bmistd          0.0007779401
## magestd        -0.0056134160
## lmeangovinsstd  0.0061620432
## bweightstd     -0.0063080169
```

```r
coef(model_test)
```

```
## 10 x 1 sparse Matrix of class "dgCMatrix"
##                         s0
## (Intercept)     0.082676240
## mom_dropout     0.015592953
## mom_hs         -0.001230440
## mom_somecoll   -0.001220581
## mom_college    -0.004776844
## apgarstd       -0.005395657
## bmistd          0.003656075
## magestd        -0.003204487
## lmeangovinsstd  0.007900653
## bweightstd     -0.007883584
```

```r
#compare the RMSE's
RMSE_cv
```

```
## [1] 0.2728714
```

```r
RMSE_test
```

```
## [1] 0.2746867
```

```r
RMSE(OLS)
```

```
## [1] 0.272868
```

We see that the RMSE for the ridge regression on the test set is slightly higher than both previous RMSE's; let's try OLS on the test set.

```r
#OLS on test set
OLS_test <- lm(hospital ~ magestd + mom_dropout + mom_hs + mom_somecoll + mom_college + lmeangovinsstd +
```

```r
#OLS table
stargazer(OLS_test,
```

```r
         type='latex', header = F, title='Test set OLS', flip=T, digits = 2, multicolumn = F,
         font.size = "small",
         column.sep.width = '1pt')
```

Table 4: Test set OLS

	Dependent variable:
	hospital
magestd	-0.003^{**}
	(0.002)
mom_dropout	0.02^{***}
	(0.005)
mom_hs	0.002
	(0.004)
mom_somecoll	0.003
	(0.004)
mom_college	
lmeangovinsstd	0.01^{***}
	(0.001)
bmistd	0.004^{***}
	(0.001)
apgarstd	-0.01^{***}
	(0.001)
bweightstd	-0.01^{***}
	(0.001)
Constant	0.08^{***}
	(0.003)
Observations	48,571
R^2	0.005
Adjusted R^2	0.005
Residual Std. Error	0.27 (df = 48562)
F Statistic	29.14^{***} (df = 8; 48562)
Note:	*p<0.1; **p<0.05; ***p<0.01

Let's compare all the RMSE's we have:

```r
#compare the RMSE's

RMSE_DF <- data.frame(Regression = c("Ridge", "Ridge", "OLS", "OLS"),
                      Dataset = c("Training", "Test", "Training", "Test"),
                      RMSE = c(RMSE_cv, RMSE_test, RMSE(OLS), RMSE(OLS_test)))
```

```r
#display table
kable(RMSE_DF,"latex", caption = "Comparison of RMSE for Ridge and OLS Regressions on Regularized Rehosp
```

Table 5: Comparison of RMSE for Ridge and OLS Regressions on Regularized Rehosp dataset

Regression	Dataset	RMSE
Ridge	Training	0.2728714
Ridge	Test	0.2746867
OLS	Training	0.2728680
OLS	Test	0.2746817

It seems like OLS on the standardized data is a better choice than the Ridge regression with the lowest $RMSE_{OLSTest} = 0.2746817$. Thus, our chosen model is:

$$hospital_i = \beta_0 + \beta_1 mage_i + \beta_2 momdropout_i + \beta_3 momhs_i + \beta_4 momsomecoll_i + \beta_5 momcollege_i + \\ \beta_6 lmeangovins_i + \beta_7 bmi_i + \beta_8 apgar5_i + \beta_1 bweight_i + \epsilon_i$$

where mage, lmeangovins, bmi, apgat5 and bweight are standardized to a mean of 0 and standard deviation of 1.