ORFE 525: Statistical Learning and Nonparametric Estimation
Homework 1

Zachary Hervieux-Moore
Collaborator: Guillaume Martinet

Thursday 9th March, 2017

Exercise 1: Complete the following in R. The goal is to predict house price from a dataset.

1) Convert the `zipcode` column into factors, and discuss why this conversion is necessary if we want to use the `zipcode` column as an explanatory variable in linear regression. (Hint: Factors are how R represents indicator variables. You can do this with `as.factor()`.)

2) Build a linear model on the training data using `lm()` by regressing the housing price on these variables: `bedrooms`, `bathrooms`, `sqft_living`, and `sqft_lot`.

    a) What's the R^2 of the model on the training data? What's the R^2 on the testing data?

    b) What if we want to use the `zipcode` to explain/predict the housing price? Add zipcode in your linear model, does it improve the R^2 or prediction power?

3) Fit the model on the training data using the `glmnet` function to predict the house prices using all the 18 features (columns on the right of the `price` column).

    a) Plot the regularization paths of Lasso and Ridge (both have L1 norm as the x-axis). Based on the 2 graphs you just plotted, which features seem to be more important than others? (**Note:** You may choose your own criteria for how to identify important features. However, you must explain these choices clearly in your write-up.)

    b) For both Lasso and Ridge regression, use a 5-fold cross validation (via the `cv.glmnet` function) to determine the tuning parameter λ. Mark the corresponding L1-norm on your plot made in Part a). Also plot the cross validation (use $\log(\lambda)$ as the x-axis). (**Hint:** Use `model.matrix` to implement categorical variables in `glmnet`)

    c) Evaluate both models by using the testing data. Record mean squared test error.

4) Fit the model on the training data using the `randomForest` function to predict the house prices using the features `sqft_living`, `sqft_lot`, `bedrooms`, `bathrooms`, `floors`, and `zipcode`. Plot variable importance

for the variables (via `varImpPlot`). Use `partialPlot` to plot the partial dependence for the top three variables in variable importance. (**Note:** Random forest cannot handle a categorical variable with more than 32 categories. Please find a meaningful way to regroup the zipcode numbers.)

5) Guess the price of Donald Trump's house (using the data provided) using the most reasonable model you think above. Do you think the predicted price is reasonable?

Answer: The code used to generate all theses answers are appended at the end of the question.

1) It is necessary to convert the column because if you regressed on the `zipcode` column without doing so, it would fit β to the the actual zipcode number. That is, something like

$$price = \beta_0 + \beta_1 * zipcode + \dots$$

Obviously, one could do this, but the meaning is questionable. Houses are not linearly related to zipcode. That is, as one increases zipcodes, the price of houses could go up and down. That is Beverly Hills (90210) and New York City (10012) are certainly more expensive than than Topeka, Kansas (66621) but Topeka is between the two values. Using factors corrects this by turning zipcode into many binary indicators. Which makes the regression

$$price = \beta_0 + \beta_1 \cdot 1_{\{zipcode=44444\}} + \beta_2 \cdot 1_{\{zipcode=44445\}} + \dots$$

Thus, this models how zipcodes can add or subtract value depending on whether the house is in the zipcode or not.

2) a) The R^2 of the linear model on the training data is 0.5101. The R^2 for the testing data is 0.5051.

b) After adding the zipcode, the R^2 of the linear model on the training data is 0.7393. The R^2 for the testing data is 0.7380. Which shows that it improves the R^2 and prediction power by a signficant amount. Also, we never expect that adding more variables decreases the R^2 because, if it did, we could just set the new variable β parameter to be 0 and get the previous R^2.

3) a) The two regularization paths are below. Note, there are many lines because I turned the zipcodes into indicator variables as per question 1.1. In the regularization paths, we are looking for the first variables to get an active β parameter at low λ values. This indicates that it has the strongest effect. Thus, the Ridge path is not too useful as all the parameters seem to diverge at 0. However, the Lasso path is more sporadic. Lasso suggests that different zipcodes are significant which makes sense. Also, waterfront, square footage of living space, and grade are also significant by this criteria.

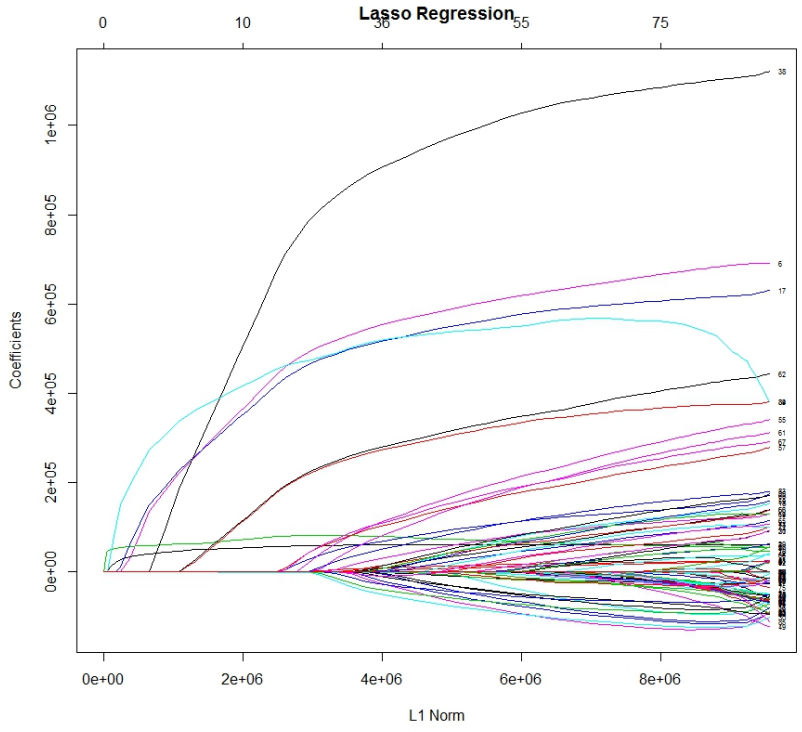

Figure 1: Lasso Regularization Path

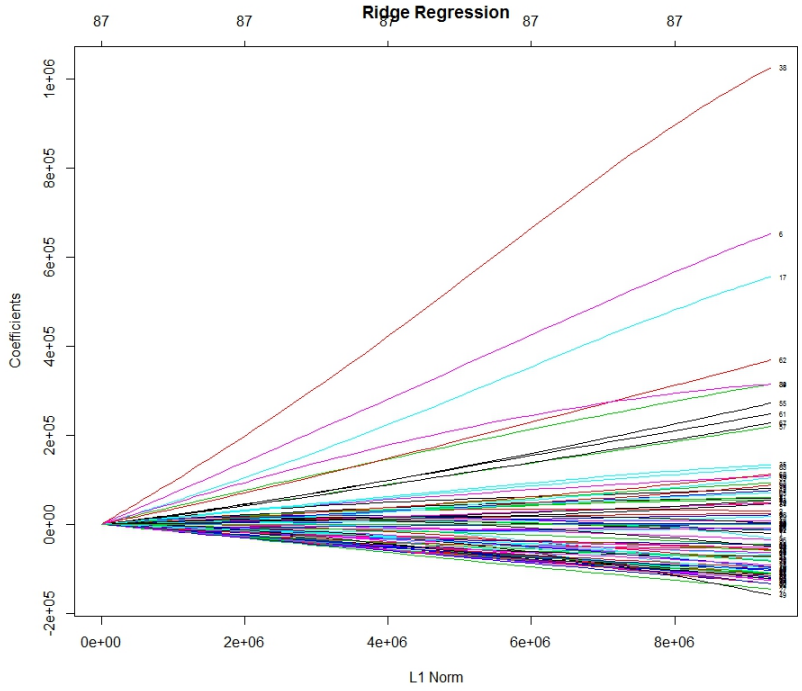

Figure 2: Ridge Regularization Path

b) After doing the 5-fold cross validation, the tuning parameter λ is 182.16 for Lasso and 28337.98 for Ridge. The corresponding norm value are shown as the solid black vertical lines in the first two figures below. The cross validation plots are shown after.

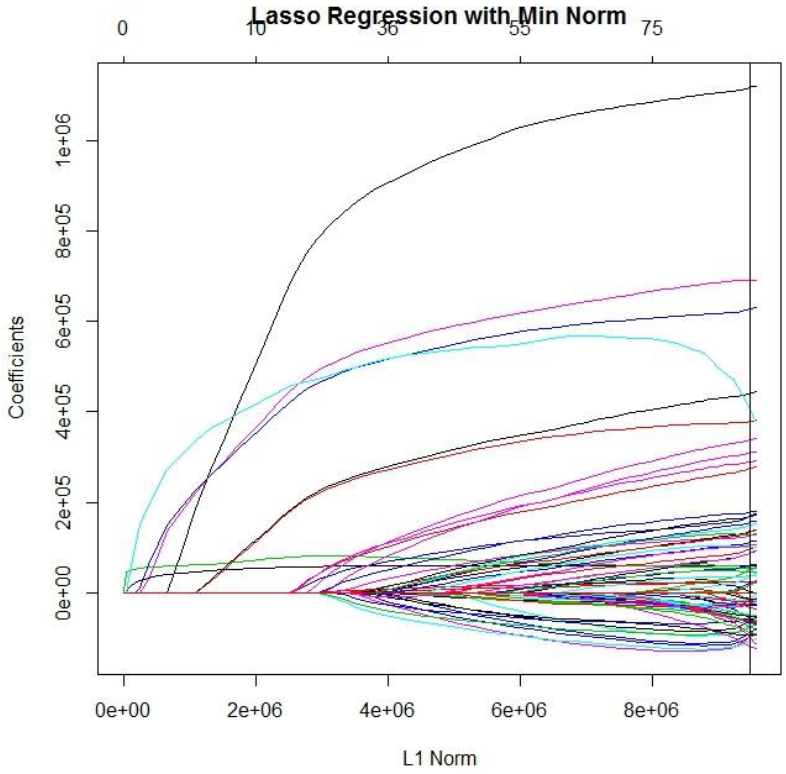

Figure 3: Lasso Regularization Path with Min Lambda Norm

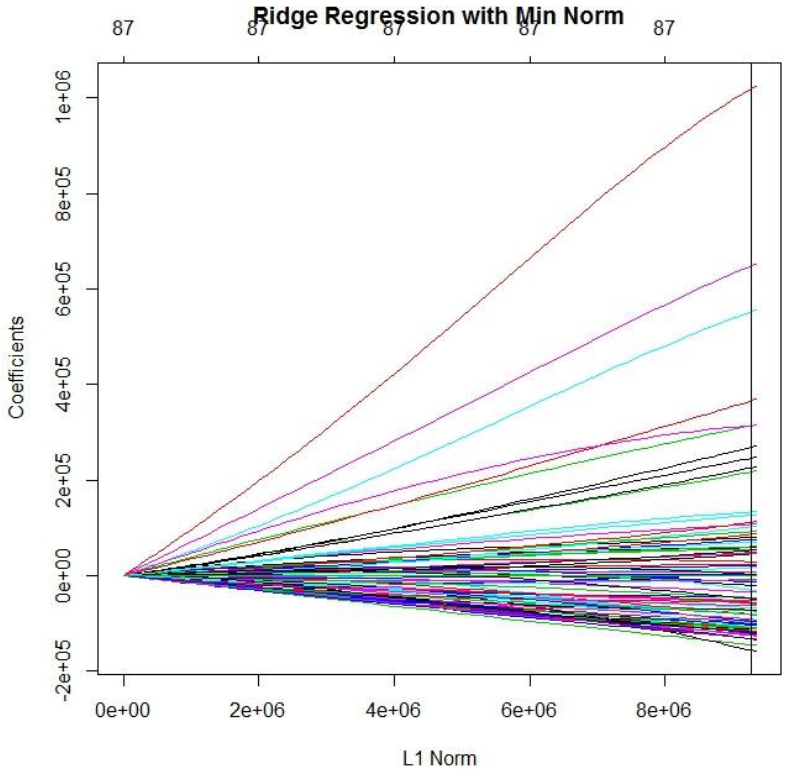

Figure 4: Ridge Regularization Path with Min Lambda Norm

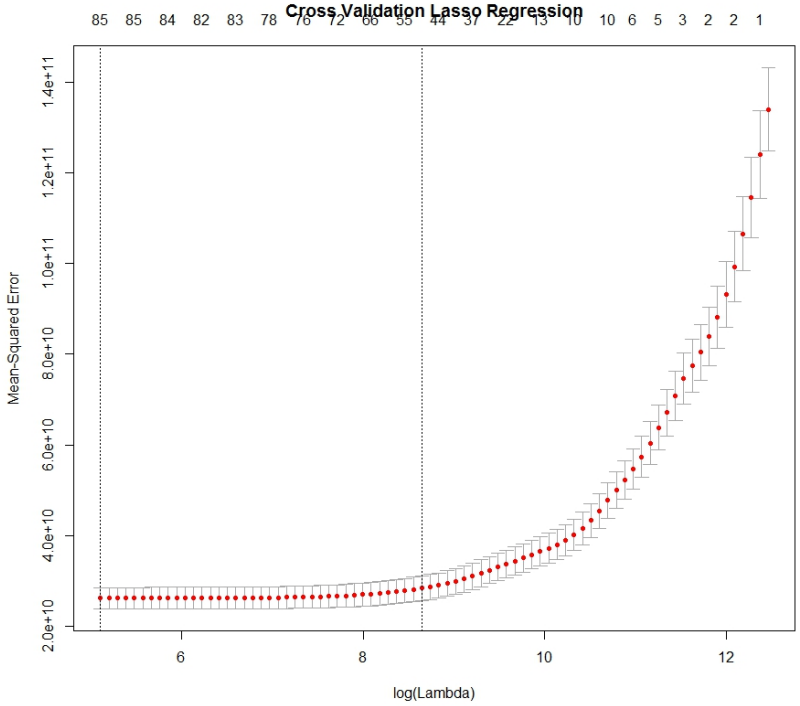

Figure 5: Lasso Cross Validation

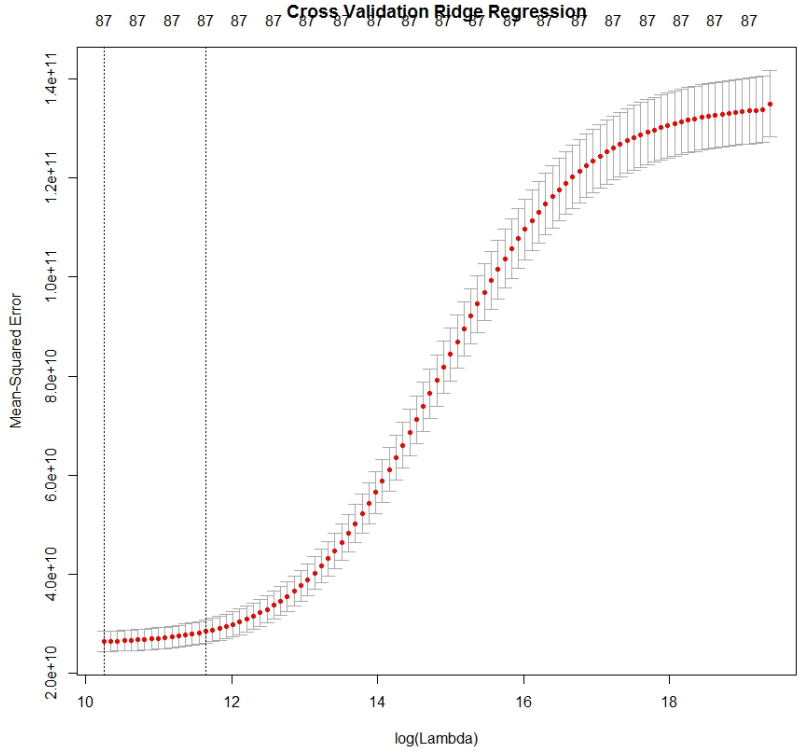

Figure 6: Ridge Cross Validation

c) Evaluating both models on the test data, the Lasso regression has an R^2 of 0.8002 and Ridge regression has an R^2 of 0.8004. The mean squared test error for Lasso is 26822892590 and the mean squared test error for Ridge is 26854501856. These numbers may seem high given that the R^2 is pretty good. But given the size of the data set, a few outliers (undervalued expensive houses) could cause that much error.

4) The plots for the importance of the variables and the partial dependence for the random forest are shown below. To handle the zipcode issue, I kept the leading 3 digits of the zipcode. This reduced the number of factors but kept the meaningfulness of the zipcode since each zipcodes that are close to each other tend to be close to each other geographically.

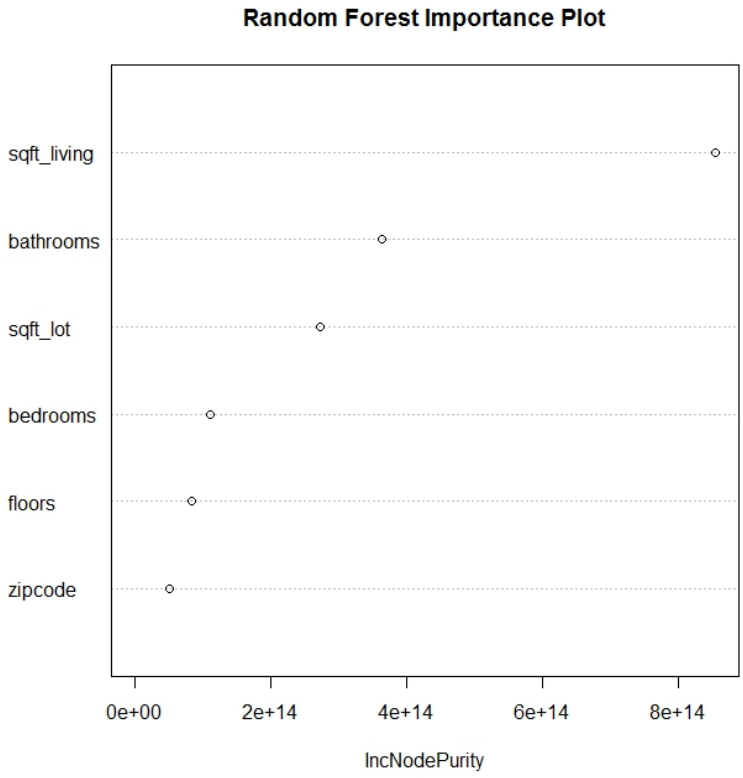

Figure 7: Variable Importance

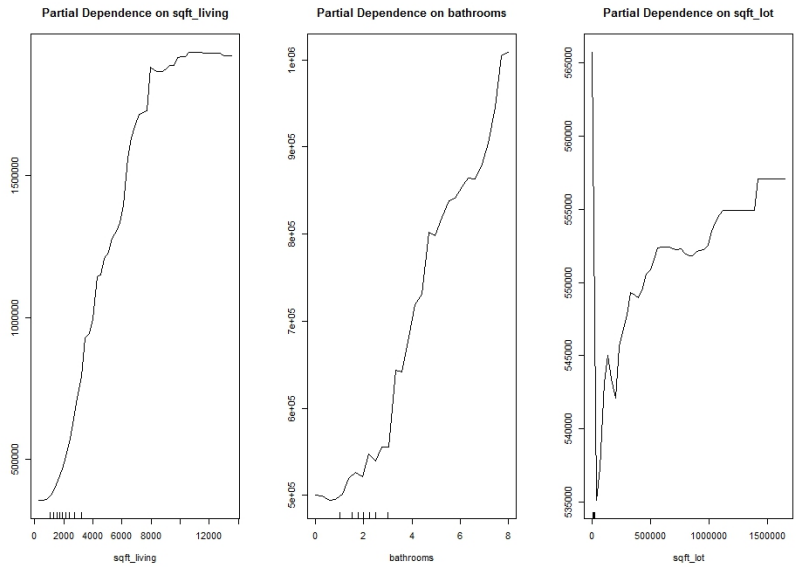

Figure 8: Partial Dependence for sqft_living, bathrooms, and sqft_lot

5) The most reasonable model developed was the Ridge regression as this had the highest R^2 value. But also, the random forest is not very good at extrapolating since it is based on making regions in the sample space. Thus, if you give a random forest a sample to estimate that it hasn't seen before, it will group it with the closest samples it has data on. Since Trump's house is no where near any house on the list, it is best not to use random forest. The Ridge regression predicts a house value of $11.3M. This is definitely in the right order of magnitude. However, given that the latitude and longitude point exactly to Bill Gates' house, I doubt that his house is only worth $11.3M.

Code Appendix

```
train <- read.csv('train.data.csv')
test <- read.csv('test.data.csv')

# Part 1.1 - Turn zipcode into factors
train$zipcode <- factor(train$zipcode)
test$zipcode <- factor(test$zipcode)

# Part 1.2a - Fit linear model
train.lm <- lm(price ~ bedrooms + bathrooms + sqft_living + sqft_lot,
    data=train)
print("R squared for training set:")
summary(train.lm)$r.squared
```

```
test.pred <- predict(train.lm, newdata=test)

## R squared for prediction (square of correlation)
print("R squared for test set:")
cor(test$price, test.pred)^2

# Part 1.2b - Fit linear model with zipcode
train.lm <- lm(price ~ bedrooms + bathrooms + sqft_living + sqft_lot +
    zipcode, data=train)
print("R squared for training set:")
summary(train.lm)$r.squared
test.pred <- predict(train.lm, newdata=test)

## R squared for prediction (square of correlation)
print("R squared for test set:")
cor(test$price, test.pred)^2

# Part 1.3a - Fit glmnet and plot regularization paths
library(glmnet)

## Lasso regression
train.glm.lasso <- glmnet(model.matrix(~ 0 + bedrooms + bathrooms + sqft
    _living + sqft_lot + floors + waterfront + view + condition + grade
    + sqft_above + sqft_basement + yr_built + yr_renovated + zipcode +
    lat + long + sqft_living15 + sqft_lot15, data=train), data.matrix(
    train$price), alpha=1)
plot(train.glm.lasso, label=TRUE)
title(main="Lasso Regression")

## Ridge regression
train.glm.ridge <- glmnet(model.matrix(~ 0 + bedrooms + bathrooms + sqft
    _living + sqft_lot + floors + waterfront + view + condition + grade
    + sqft_above + sqft_basement + yr_built + yr_renovated + zipcode +
    lat + long + sqft_living15 + sqft_lot15, data=train), data.matrix(
    train$price), alpha=0)
dev.new()
plot(train.glm.ridge, label=TRUE)
title(main="Ridge Regression")

# Part 1.3b - Cross validation
train.glm.lasso.cv <- cv.glmnet(model.matrix(~ 0 + bedrooms + bathrooms
    + sqft_living + sqft_lot + floors + waterfront + view + condition +
    grade + sqft_above + sqft_basement + yr_built + yr_renovated +
    zipcode + lat + long + sqft_living15 + sqft_lot15, data=train), data
    .matrix(train$price), alpha=1, nfolds=5)
print("Lambda min for lasso:")
train.glm.lasso.cv$lambda.min
dev.new()
plot(train.glm.lasso.cv)
title(main="Cross Validation Lasso Regression")

## Looked manually to figure out min lasso index
print("Beta min for lasso:")
train.glm.lasso.cv$glmnet.fit$beta[,"s67"]
print("Beta min l_1 norm:")
norm(as.matrix(train.glm.lasso.cv$glmnet.fit$beta[,"s67"]), type="1")
```

```
## Plot min norm
dev.new()
plot(train.glm.lasso.cv$glmnet.fit)
abline(v=train.glm.lasso.cv$lambda.min)
title(main="Lasso Regression with Min Norm")

train.glm.ridge.cv <- cv.glmnet(model.matrix(~ 0 + bedrooms + bathrooms
    + sqft_living + sqft_lot + floors + waterfront + view + condition +
    grade + sqft_above + sqft_basement + yr_built + yr_renovated +
    zipcode + lat + long + sqft_living15 + sqft_lot15, data=train), data
    .matrix(train$price), alpha=0, nfolds=5)
print("Lambda min for ridge:")
train.glm.ridge.cv$lambda.min
dev.new()
plot(train.glm.ridge.cv)
title(main="Cross Validation Ridge Regression")

## Looked manually to figure out min ridge index
print("Beta min for ridge:")
train.glm.ridge.cv$glmnet.fit$beta["s99"]
print("Beta min l_1 norm:")
norm(as.matrix(train.glm.ridge.cv$glmnet.fit$beta["s99"]), type="1")

## Plot min norm
dev.new()
plot(train.glm.ridge.cv$glmnet.fit)
abline(v=train.glm.ridge.cv$lambda.min)
title(main="Ridge Regression with Min Norm")

# Part 1.3c - Evaluate models
test.glm.lasso.pred <- predict(train.glm.lasso.cv, newx=model.matrix(~ 0
    + bedrooms + bathrooms + sqft_living + sqft_lot + floors +
    waterfront + view + condition + grade + sqft_above + sqft_basement +
    yr_built + yr_renovated + zipcode + lat + long + sqft_living15 +
    sqft_lot15, data=test), s="lambda.min")
print("Mean squared error of Lasso prediction:")
sum((test$price - test.glm.lasso.pred)^2)/length(test.glm.lasso.pred)
print("R squared for test set:")
cor(test$price, test.glm.lasso.pred)^2

test.glm.ridge.pred <- predict(train.glm.ridge.cv, newx=model.matrix(~ 0
    + bedrooms + bathrooms + sqft_living + sqft_lot + floors +
    waterfront + view + condition + grade + sqft_above + sqft_basement +
    yr_built + yr_renovated + zipcode + lat + long + sqft_living15 +
    sqft_lot15, data=test), s="lambda.min")
print("Mean squared error of Ridge prediction:")
sum((test$price - test.glm.ridge.pred)^2)/length(test.glm.ridge.pred)
print("R squared for test set:")
cor(test$price, test.glm.ridge.pred)^2

# Part 1.4 - Random Forest
library(randomForest)

## zipcode needs to be changed, leading digits highly meaningful to
    specific region of US
train$zipcode <- substr(train$zipcode,1,3)
test$zipcode <- substr(test$zipcode,1,3)
```

```
train.randomforest <- randomForest(price ~ sqft_living + sqft_lot +
    bedrooms + bathrooms + floors + zipcode, data=train)
dev.new()
varImpPlot(train.randomforest, main="Random Forest Importance Plot")

## Plot partial dependence of top 3 important variables
dev.new()
train.randomforest.imp <- importance(train.randomforest)
train.randomforest.imp <- train.randomforest.imp[order(train.
    randomforest.imp, decreasing=TRUE),]
train.randomforest.imp <- train.randomforest.imp[1:3]
train.randomforest.impvar <- names(train.randomforest.imp)
op <- par(mfrow=c(1, 3))
for (i in seq_along(train.randomforest.impvar)) {
  partialPlot(train.randomforest, train, train.randomforest.impvar[i],
      xlab=train.randomforest.impvar[i], main=paste("Partial Dependence
      on", train.randomforest.impvar[i]))
}
par(op)

## Calculate R squared
test.randomforest.pred <- predict(train.randomforest, newdata=test)
print("R squared for test set:")
cor(test$price, test.randomforest.pred)^2

# Part 1.5 - Trump

## Reload data to ensure integrity from zipcode change

train <- read.csv('train.data.csv')
test <- read.csv('test.data.csv')

train$zipcode <- factor(train$zipcode)
test$zipcode <- factor(test$zipcode)

donald_trump = data.frame(X=1, id=1, date='20141013T000000', price
    =14000000,
  bedrooms=8, bathrooms=25, sqft_living=50000,
  sqft_lot=225000, floors=4, zipcode=factor(c('98039')), condition=10,
      grade=10,
  waterfront=1, view=4, sqft_above=37500, sqft_basement=12500, yr_built
      =1994,
  yr_renovated=2010, lat=47.627606, long=-122.242054, sqft_living15
      =5000,
  sqft_lot15=40000)
## Make large test set for model.matrix to generate right matrix for
    Trump
donald_trump_matrix <- rbind(test, donald_trump)

predict(train.lm, newdata=donald_trump)
predict(train.glm.lasso.cv, newx=model.matrix(~ 0 + bedrooms + bathrooms
    + sqft_living + sqft_lot + floors + waterfront + view + condition +
    grade + sqft_above + sqft_basement + yr_built + yr_renovated +
    zipcode + lat + long + sqft_living15 + sqft_lot15, data=donald_trump
    _matrix), s="lambda.min")[length(donald_trump_matrix[,1]),]
predict(train.glm.ridge.cv, newx=model.matrix(~ 0 + bedrooms + bathrooms
    + sqft_living + sqft_lot + floors + waterfront + view + condition +
```

```
    grade + sqft_above + sqft_basement + yr_built + yr_renovated +
    zipcode + lat + long + sqft_living15 + sqft_lot15, data=donald_trump
    _matrix), s="lambda.min")[length(donald_trump_matrix[,1]),]
donald_trump$zipcode <- substr(donald_trump$zipcode,1,3)
predict(train.randomforest, newdata=donald_trump)
```

Exercise 2: The Lasso usually does not have an explicit formula for its solution. However, in this problem, you will be lucky enough as we are considering a very special case.

Let $\theta \in \mathbb{R}^d$ and $Y \in \mathbb{R}^d$. For any $\tau > 0$, we define the hard thresholding $\hat{\theta}^{HRD}$ and the soft thresholding $\hat{\theta}^{SFT}$ as

$$\hat{\theta}_j^{HRD} = \begin{cases} Y_j & \text{if } |Y_j| > 2\tau \\ 0 & \text{if } |Y_j| \leq 2\tau \end{cases} \text{ and } \hat{\theta}_j^{SFT} = \begin{cases} Y_j - 2\tau & \text{if } Y_j > 2\tau \\ 0 & \text{if } |Y_j| \leq 2\tau \\ Y_j + 2\tau & \text{if } Y_j < -2\tau \end{cases}$$

for $j = 1, \ldots, d$. Define $\|\theta\|_0 = |\{j : \theta_j \neq 0\}|$, where for any set S, $|S|$ is the cardinality of S. Proce that

$$\hat{\theta}_j^{HRD} = \arg\min_{\theta \in \mathbb{R}^d} \{\|Y - \theta\|_2^2 + 4\tau^2 \|\theta\|_0\},$$
$$\hat{\theta}_j^{SFT} = \arg\min_{\theta \in \mathbb{R}^d} \{\|Y - \theta\|_2^2 + 4\tau \|\theta\|_1\}$$

Answer: First, let us work with $\hat{\theta}_j^{HRD}$. Expand the norm,

$$\arg\min_{\theta \in \mathbb{R}^d} \|Y - \theta\|_2^2 + 4\tau^2 \|\theta\|_0$$
$$= \arg\min_{\theta \in \mathbb{R}^d} Y^T Y - 2Y^T \theta + \theta^T \theta + 4\tau^2 \|\theta\|_0$$

We drop the term that has no θ dependence

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y^T \theta + \|\theta\|_2^2 + 4\tau^2 \|\theta\|_0$$
$$= \arg\min_{\theta \in \mathbb{R}^d} \sum_{j=1}^{d} -2Y_j^T \theta_j + \theta_j^2 + 4\tau^2 \cdot 1_{\{\theta_j \neq 0\}}$$

Thus, we can just minimize each component individually.

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y_j \theta_j + \theta_j^2 + 4\tau^2 \cdot 1_{\{\theta_j \neq 0\}} \tag{1}$$

We lower bound this by taking the absolute values of Y_j and θ_j as this can only reduce the min.

$$\geq \arg\min_{\theta \in \mathbb{R}^d} -2|Y_j||\theta_j| + \theta_j^2 + 4\tau^2 \cdot 1_{\{\theta_j \neq 0\}}$$

Case 1, suppose that $|Y_j| \leq 2\tau$. Then we can lower bound even further

$$\geq \arg\min_{\theta \in \mathbb{R}^d} -4\tau|\theta_j| + \theta_j^2 + 4\tau^2 \cdot 1_{\{\theta_j \neq 0\}}$$

These are all positive terms, thus picking $\theta_j = 0$ minimizes this and the original problem achieves this lower bound. Thus, $\theta_j = 0$ is the argmin when $|Y_j| \leq 2\tau$. Case 2, assume that $|Y_j| > 2\tau$. Going back to equation (1)

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y_j\theta_j + \theta_j^2 + 4\tau^2 \cdot 1_{\{\theta_j \neq 0\}}$$

Suppose $\theta_j \neq 0$, then

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y_j\theta_j + \theta_j^2 + 4\tau^2$$

Solving this quadratic yields that $\theta_j = Y_j$. Note that the objective value is negative when $\theta_j = Y_j$ since $|Y_j| > 2\tau \implies 0 > 4\tau^2 - Y_j^2$. Thus, $\theta_j = 0$ is not a better solution. We conclude that

$$\hat{\theta}_j^{HRD} = \begin{cases} Y_j & \text{if } |Y_j| > 2\tau \\ 0 & \text{if } |Y_j| \leq 2\tau \end{cases}$$

Now for $\hat{\theta}_j^{SFT}$, we proceed similarly as before up to equation (1) to get

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y_j\theta_j + \theta_j^2 + 4\tau|\theta_j| \tag{2}$$

Case 1, assume $|Y_j| \leq 2\tau$ and use the same series of inequalities

$$\begin{aligned} &\geq \arg\min_{\theta \in \mathbb{R}^d} -2|Y_j||\theta_j| + \theta_j^2 + 4\tau|\theta_j| \\ &\geq \arg\min_{\theta \in \mathbb{R}^d} -4\tau|\theta_j| + \theta_j^2 + 4\tau|\theta_j| \\ &= \theta_j^2 \end{aligned}$$

Which is clearly minimized at $\theta_j = 0$. Case 2, suppose that $\theta_j > 0$, by equation (2)

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y_j\theta_j + \theta_j^2 + 4\tau\theta_j$$

Solving yields $\theta_j = Y_j - 2\tau$. This is positive when $Y_j > 2\tau$. Thus, $\theta_j = Y_j - 2\tau$ when $Y_j > 2\tau$. Case 3, suppose that $\theta_j < 0$, again by equation (2)

$$= \arg\min_{\theta \in \mathbb{R}^d} -2Y_j\theta_j + \theta_j^2 - 4\tau\theta_j$$

Solving yields $\theta_j = Y_j + 2\tau$. This is negative when $Y_j < -2\tau$. Thus, $\theta_j = Y_j - 2\tau$ when $Y_j > 2\tau$. We conclude that

$$\hat{\theta}_j^{SFT} = \begin{cases} Y_j - 2\tau & \text{if } Y_j > 2\tau \\ 0 & \text{if } |Y_j| \leq 2\tau \\ Y_j + 2\tau & \text{if } Y_j < -2\tau \end{cases}$$

Exercise 3: Suppose the singular values of a matrix $X \in \mathbb{R}^{d_1 \times d_2}$ are $\sigma_1(X) \geq \ldots \geq \sigma_d(X)$, where $d = \min(d_1, d_2)$. Define the nuclear norm $\|X\|_* = \sum_{k=1}^{d} \sigma_k(X)$.

1) Given a 2×2 symmetric matrix

$$M(x, y, z) = \begin{pmatrix} x & y \\ y & z \end{pmatrix}$$

what are the shapes of the two sets

$$\{(x, \sqrt{2}y, z) \in \mathbb{R}^3 : \text{Rank}(M(x, y, z)) = 1, \|M(x, y, z)\|_{op} = 1\} \text{ and}$$
$$\{(x, \sqrt{2}y, z) \in \mathbb{R}^3 : \|M(x, y, z)\|_* \leq 1\}$$

where $\|\cdot\|_{op}$ is the matrix operator norm. PLot these two sets in $\mathbb{R}^3$.

2) a) Prove that the dual of the operator norm is the nuclear norm. In specific, prove that

$$\|Y\|_* = \max_{X:\|X\|_{op} \leq 1} \langle Y, X \rangle$$

b) Show that the nuclear norm $\|\cdot\|_*$ isa convex function.

c) Prove the convex hull of bounded nonsymmetrix rank-1 matrices is the nuclear norm ball, i.e.,

$$\text{conv}\{uv^T : \|uv^T\|_{op} \leq 1, u \in \mathbb{R}^{d_1}, v \in \mathbb{R}^{d_2}\} = \{X \in \mathbb{R}^{d_1 \times d_2} : \|X\|_* \leq 1\}$$

3) Let X be a square matrix. Prove the optimal solution to the unconstrained optimization

$$\min_Z \|Z\|_* + \frac{\tau}{2}\|X - XZ\|_F^2$$

is unique and has the formulation

$$\widehat{Z} = V \max\left(0, I - \frac{1}{\tau}\Lambda^{-2}\right) V^T$$

where $\|\cdot\|_F$ is the Frobenius norm and $X = U\Lambda V^T$ is the SVD of X.

(**Hint 1:** Nuclear norm and Frobenius norm are invariant under unitary transforms.

Hint 2: You may need to show $\|A\|_* \geq \|\text{diag}(A)\|_*$ for any square matrix

Hint 3: To prove uniqueness, you should explain what if V is not uniquely defined when singular values have multiplicity.)

Answer:

1) Since $M(x, y, z)$ is symmetric, we have that $\text{Rank}(M(x, y, z)) = 1$ implies that $M(x, y, z)$ has one eigenvalue equal to 0 and the other is ± 1. We also have that $\det(M) = xz - y^2 = 0$ because of the 0 eigenvalue. Also, we have $\text{trace}(M) = x + z = \pm 1$ since it is the sum of eigenvalues. Thus, we have

$$S_1 = \{(x, \sqrt{2}y, z) : x + z = \pm 1, xz = y^2\}$$

Equivalently

$$S_1 = \{(x, y, z) : x + z = \pm 1, xz = y^2/2\}$$

It is not very obvious what this is. Thus, consider the new variables $x' = \frac{1}{\sqrt{2}}x + \frac{1}{\sqrt{2}}z$, $y' = -\frac{1}{\sqrt{2}}x + \frac{1}{\sqrt{2}}z$, $z' = y$. Then we can write the above set as

$$S_1 = \{(x, y, z) : \sqrt{2}x' = \pm 1, x'^2 = z'^2 + y'^2\}$$

Thus, in this new coordinate system, it is two circles or radius $\frac{1}{\sqrt{2}}$. One is in the plane $x + z = 1$ centered at (1/2,0,1/2) and the other in the plane $x + z = -1$ centered at (-1/2,0,-1/2). The plot is shown below

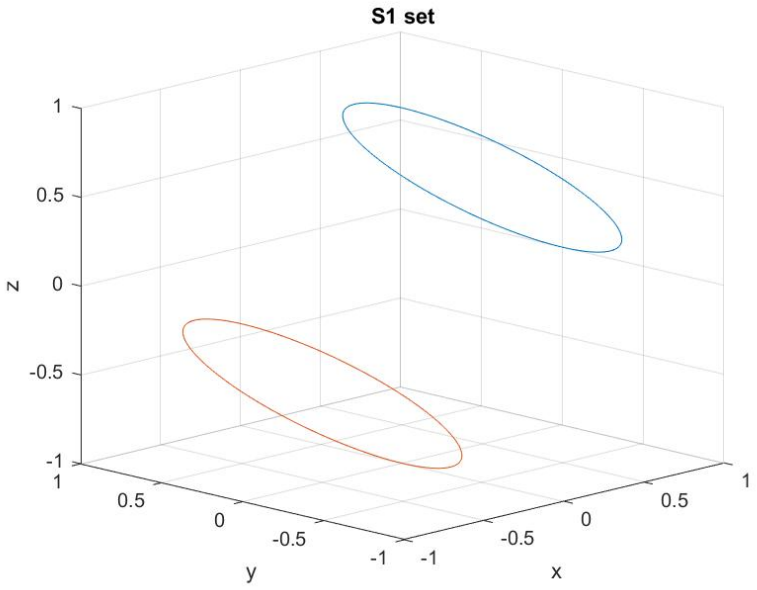

Figure 9: Plot of S_1

Now for the other set S_2. Since M is symmetric, we have that the singular values are precisely the absolute values of the eigenvalues. Thus, $\|M\|_* \leq 1 \implies |\lambda_1| + |\lambda_2| \leq 1$. This gives us four inequalities, $\pm\lambda_1 + \pm\lambda_2 \leq 1$. We also have that

$$\det(M(x, y/\sqrt{2}, z) - \lambda I) = \lambda^2 - (x+z)\lambda + xz - y^2/2$$

Which means that the roots are

$$\lambda = \frac{x + z \pm \sqrt{(x-z)^2 + 2y^2}}{2}$$

Combining these roots with the four inequalities before, we get the following inequalities

$$-1 \leq x + z \leq 1$$
$$(x-z)^2 + 2y^2 \leq 1$$

Using the same change of variable as before, these are equivalent to

$$-1 \leq \sqrt{2}x' \leq 1$$
$$y'^2 + z'^2 \leq 1/2$$

Thus, this is a cylinder with axis (1,0,1) passing through the origin whose top and bottom line on the plane $x + z = \pm 1$ and radius $1/\sqrt{2}$. The plot is below.

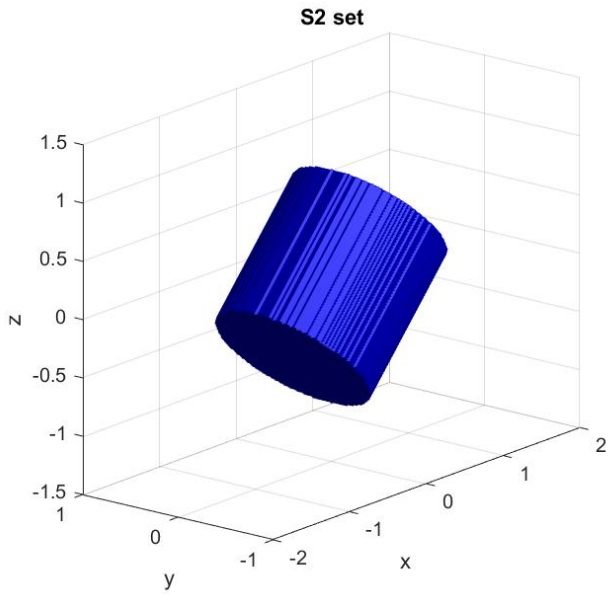

Figure 10: Plot of S_2

2) a) We show that both sides are less than or equal to each other to show equality. Let $U\Sigma V^T$ be the SVD of Y. Then, let $X = UV^T$. Their inner product is

$$\begin{aligned} \langle Y, X \rangle &= \text{trace}(V\Sigma U^T U V^T) \\ &= \text{trace}(V\Sigma I V^T) = \text{trace}(\Sigma V^T V) \\ &= \text{trace}(\Sigma) = \|Y\|_* \end{aligned}$$

We also have that $\|X\|_{op} = 1$ since it is the largest singular value of X. Thus, we have showed $\|Y\|_* \leq \max_{X:\|X\|_{op}\leq 1} \langle Y, X \rangle$
Now suppose that $\|X\|_{op} \leq 1$. Then

$$\langle Y, X \rangle = \text{trace}(V\Sigma U^T X) = \text{trace}(U^T X V \Sigma)$$

We note that each operator norm for the matrices U^T and V are less than 1 since they are orthonormal and X by assumption.

$$\|U^T X V\|_{op} \leq \|U^T\|_{op}\|X\|_{op}\|V\|_{op} \leq 1$$

Now let $X' = U^T X V$, which makes the trace of $\langle Y, X\rangle$ equal to

$$= \text{trace}(X'\Sigma) = \sum_{i=1}^{n} \sigma_i x'_{ii} \leq \sum_{i=1}^{n} \sigma_i |x'_{ii}|$$

But we just showed that $\|X'\|_{op} \leq 1$. Hence, we have

$$= \text{trace}(X'\Sigma) \leq \sum_{i=1}^{n} \sigma_i = \|Y\|_*$$

So, $\max_{X:\|X\|_{op}\leq 1}\langle Y, X\rangle \leq \|Y\|_*$ and we conclude that

$$\max_{X:\|X\|_{op}\leq 1} \langle Y, X\rangle = \|Y\|_*$$

b) Let X, Y be matrices and $\lambda \in (0, 1)$. Then

$$\begin{aligned}\|\lambda X + (1-\lambda)Y\|_* &= \max_{Z:\|Z\|_{op}\leq 1} \lambda\langle X, Z\rangle + (1-\lambda)\langle Y, Z\rangle \\ &\leq \lambda \max_{Z:\|Z\|_{op}\leq 1} \langle X, Z\rangle + (1-\lambda) \max_{Z:\|Z\|_{op}\leq 1} \langle Y, Z\rangle \\ &= \lambda\|X\|_* + (1-\lambda)\|Y\|_*\end{aligned}$$

Therefore $\|\cdot\|_*$ is convex.

c) Denote the sets by $A = \text{conv}\{uv^T : \|uv^T\|_{op} \leq 1, u \in \mathbb{R}^{d_1}, v \in \mathbb{R}^{d_2}\}$ and $B = \{X \in \mathbb{R}^{d_1 \times d_2} : \|X\|_* \leq 1\}$. We first show that $A \subseteq B$. From the previous part, $\|\cdot\|_*$ is convex, therefore B is convex at it is a sublevel set of a convex function. Thus, we need not worry about the convex hull part and only need to show that $\forall uv^T$ with $\|uv^T\|_{op} \leq 1$, then $uv^T \in B$.

We have that

$$uv^T = \frac{u}{\|u\|}\|u\|\|v\|\frac{v^t}{\|v^T\|}$$

Where we assumed $u, v^T \neq 0$ because if they did, they are trivially in B. Thus, $\|u\|\|v\|$ is a non-zero singular calue of uv^T as one can view the above as the SVD of uv^T. This gives us $\|uv^T\|_* = \|u\|\|v\|$. We then have

$$\|uv^T\|_{op} = \max_i \sigma_i(uv^T) = \|u\|\|v^T\| = \|uv^T\|_*$$

However, we by assumption that the operator norm is bounded by 1and so the nuclear norm is bounded by 1, so we conclude that $uv^T \in B$.

Now we show the other inclusion. First, define the SVD of $X = U\Sigma V^T$. If $X = 0$, then it is trivially in A. Assume $X \neq 0$, so that $\|X\|_* = \sum_{i=1}^d \sigma_i(X) > 0$. We first show that the i^{th} columns of U and V, $u_i v_i^T$, is in A. By Cauchy-Schwarz, for all $x \in \mathbb{R}^{d_2}$

$$\|u_i v_i^T x\| \leq \|u_i\|\|v_i^T\|\|x\| = \|x\|$$

Thus, we have that $u_i v_i^T \in A$. Now, we rewrite X as follow which is the SVD in summation form

$$X = \sum_{i=1}^d \frac{\sigma_i}{\sum_{i=1}^d \sigma_i}(u_i v_i^T \sum_{i=1}^d \sigma_i)$$

But, since $\sum_{i=1}^d \sigma_i$ by assumption, we have $(u_i v_i^T \sum_{i=1}^d \sigma_i) \in A$. Then the outer sum is a convex combination of element in A, thus it is also in A. So we showed that $X \in A$ and $B \subseteq A$. We conclude that $A = B$.

3) Starting with the objective, we first do the SVD decomposition for X. Then, we use the hint and use the fact that the Frobenius norm is invariant under unitary transforms.

$$\begin{aligned}
&\|Z\|_* + \frac{\tau}{2}\|X - XZ\|_F^2 \\
&= \|Z\|_* + \frac{\tau}{2}\|U\Lambda V^T - U\Lambda V^T Z\|_F^2 \\
&= \|Z\|_* + \frac{\tau}{2}\|\Lambda V^T - \Lambda V^T Z\|_F^2 \\
&= \|Z\|_* + \frac{\tau}{2}\|\Lambda - \Lambda V^T Z V\|_F^2
\end{aligned}$$

Since V is unitary, then there is a one-to-one correspondence between Z and $V^T ZV$ and we can focus on solving

$$\min_Z \|Z\|_* + \frac{\tau}{2}\|\Lambda - \Lambda Z\|_F^2$$

We first handle the case when Z is diagonal. Let $Z = \Lambda_Z = \text{diag}(\lambda_{z_i})$. Then the objective becomes

$$\|\Lambda_Z\|_* + \frac{\tau}{2}\|\Lambda - \Lambda\Lambda_Z\|_F^2$$
$$\sum_{i=1}^{d} |\lambda_{z_i}| + \frac{\tau}{2}\sum_{i=1}^{d} \lambda_i^2(1-\lambda_{z_i})^2$$

Thus, we get lower values when $\lambda_{z_i} \geq 0$, so we take $\Lambda \succeq 0$. Which turns the minimization problem into

$$\min_{\lambda_{z_i} \geq 0} \sum_{i=1}^{d} |\lambda_{z_i}| + \frac{\tau}{2}\sum_{i=1}^{d} \lambda_i^2(1-\lambda_{z_i})^2$$

This is a convex optimization problem. Slater's condition also guarantees that KKT conditions are necessary and sufficient. We then have that there exists $\mu \geq 0$ s.t.

$$-\tau\lambda_i^2(1-\lambda_{z_i}) + 1 - \mu_i = 0$$
$$\iff \tau\lambda_i^2\lambda_{z_i} - \tau\lambda_i^2 + 1 = \mu_i \geq 0$$

and our slackness conditions

$$\mu_i \lambda_{z_i} = 0 \ \forall i$$

Now we handle a couple of cases. Suppose $\tau\lambda_i^2 - 1 > 0$. Then if $\mu_i > 0$, $\lambda_{z_i} = 0$ by slackness, and so $\mu_i = -\tau\lambda_i^2 + 1 < 0$ which is a contradiction. Thus, we have

$$\mu_i = 0$$
$$\lambda_{z_i} = 1 - \frac{1}{\tau}\lambda_i^{-2} > 0$$

If $\tau\lambda_i^2 - 1 \leq 0$, then $\lambda_{z_i} = 0$ since if it strictly positive, then $\mu_i > 0$ which fails slackness. Thus, $\lambda_{z_i} = 0$. Therefore, we have shown for a diagonal matrix Z, we have that the minimizer is $\max(0, I - \frac{1}{\tau}\Lambda^{-2})$.

Now, if we can show that the solution must be a diagonal, then we are done. We begin by showing that $|A\|_* \geq \|\text{diag}(A)\|_* \; \forall A$. We have that

$$\|\text{diag}(A)\|_* = \sum_{i=1}^{d} |a_{ii}| \text{ and}$$
$$\|A\|_* = \max_{X:\|X\|_{op}\leq 1} \langle A, X\rangle$$

Now construct X to be diagonal with $x_{ii} = \text{sign}(a_{ii})$. This yields

$$\|X\|_{op} \leq 1 \text{ and } \langle A, X\rangle = \|\text{diag}(A)\|_*$$

Thus we conclude that $|A\|_* \geq \|\text{diag}(A)\|_* \; \forall A$. Thus, if we find a solution Z, then we can decrease the nuclear norm of the minimization by simply taking the diag(Z) instead to be our solution. Lets see what happens to the Frobenius norm.

$$\begin{aligned}
&\|\Lambda - \Lambda Z\|_F^2 \\
&= \text{trace}((\Lambda - \Lambda Z)^T(\Lambda - \Lambda Z)) \\
&= \text{trace}(\Lambda^2 - Z^T\Lambda^2 - \Lambda^2 Z + Z^T\Lambda^2 Z) \\
&= \text{trace}(\Lambda^2) - 2 * \text{trace}(\Lambda^2 Z) + \text{trace}(\Lambda^2 ZZ^T) \\
&= \text{trace}(\Lambda^2) - 2 * \text{trace}(\Lambda^2 \text{diag}(Z)) + \|\Lambda Z\|_F^2 \\
&\geq \text{trace}(\Lambda^2) - 2 * \text{trace}(\Lambda^2 \text{diag}(Z)) + \|\Lambda \text{diag}(Z)\|_F^2
\end{aligned}$$

Thus, by taking Z to be the diag(Z), then we see that the Frobenius norm does not change since Λ^2 is diagonal. The first term has no Z, the middle term will be unchanged, and the last term is $\|\Lambda Z\|_F^2$ whose singular values do not increase since Λ^2 is positive and diagonal. That is, for every Z, we can find a diagonal matrix that is at least as good as it. Thus, we conclude that all solutions will be diagonal and so we have shown that

$$\widehat{Z} = V \max(0, I - \frac{1}{\tau}\Lambda^{-2})V^T$$

is the optimal solution to the optimization problem.

Exercise 4: Consider

$$\min_{\beta} \frac{1}{2}\|Y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

1) If $\widehat{\beta}_1$ and $\widehat{\beta}_2$ are both minimizers of the above problem, prove they have the same prediction $X\widehat{\beta}_1 = X\widehat{\beta}_2$. This result suggests that even the Lasso solution may not be unique, linear predictors based on them turn out to be unique!

   (**Hint:** Consider the vector $\alpha\widehat{\beta}_1 + (1-\alpha)\widehat{\beta}_2$ for $\alpha \in (0,1)$.)

2) Let $\widehat{\beta}$ be a minimizer for the above problem. Denote X_j to be the j^{th} column of X, prove that

$$\begin{cases} \lambda = X_j^T(Y - X\widehat{\beta}) & \text{if } \widehat{\beta}_j > 0 \\ \lambda = -X_j^T(Y - X\widehat{\beta}) & \text{if } \widehat{\beta}_j < 0 \\ \lambda \geq |X_j^T(Y - X\widehat{\beta})| & \text{if } \widehat{\beta}_j = 0 \end{cases}$$

   (**Hint:** If x^* minimizes a convex function $f(x)$, if and only if $0 \in \partial f(x^*)$. You can directly use this and the subgradient of ℓ_1 norm.)

3) If $\lambda > \|X^TY\|_\infty$, where $\|\cdot\|_\infty$ is the sup-norm, prove that the minimizer of the above minimization must be zero.

   (**Hint:** User Q4.1 and Q4.2.)

4) Suppose the problem above has a unique minimizer for all $\lambda > 0$, denoted by $\widehat{\beta}(\lambda)$. Given $[\lambda_0, \lambda_1]$, suppose the support and the signs of $\widehat{\beta}(\lambda)$ are unchanged for $\lambda_0 \leq \lambda \leq \lambda_1$. Show that there is a vector γ_0 such that

$$\widehat{\beta}(\lambda) = \widehat{\beta}(\lambda_0) - (\lambda - \lambda_0)\gamma_0$$

   This result proves that the Lasso regularization path is piecewise linear!! Compare the result with the Lasso regularization path in Q1.3 and explain why that plot is also piecewise linear (Note, the horizontal axis in that plot is $\|\beta\|_1$?)

   (**Hint:** Use Q4.2. You may need the Moore-Penrose pseudoinverse. Given a SVD of rank r matrix $A = U\Lambda V^T \in \mathbb{R}^{n\times d}$, where $U \in \mathbb{R}^{n\times r}$, $V \in \mathbb{R}^{d\times r}$. The pseudoinverse of A is $A^\dagger = V\Lambda^{-1}U^T$.)

5) Consider the design such that $X \in \mathbb{R}^{n \times d}$ with $\text{Rank}(X) = n$. We again suppose $\widehat{\beta}(\lambda)$ is the unique minimizer of the optimization problem above for all $\lambda > 0$. Prove that when λ goes to zero, the Lasso estimator converges to one of the compressed sensing estimator

$$\widehat{\beta}^{CS} = \arg\min_{\beta} \|\beta\|_1 \text{ s.t. } Y = X\beta$$

Namely, prove that there exists a solution of the above problem $\widehat{\beta}^{CS}$ such that

$$\lim_{\lambda \to 0^+} \widehat{\beta}\lambda = \widehat{\beta}^{CS}$$

(**Hint:** You may need to first prove the existence of the limitation by bounding the number of piecewise lines in Q4.4.)

Answer: 1) Note that $X\widehat{\beta}$ is a vector. Thus, consider the vector function $f(X) = \frac{1}{2}\|Y - X\|_2^2$. Then we have

$$\nabla f(X) = X - Y$$
$$\nabla^2 f(X) = I \succ 0$$

That is, $f(X)$ is strictly convex. Thus, $\frac{1}{2}\|Y - X\beta\|_2^2 + \lambda\|\beta\|_1$ is a strictly convex function plus a convex function. This results in a strictly convex function. Now, take any two optimal $X\widehat{\beta}_1$ and $X\widehat{\beta}_2$ and suppose they are not equal. We will show that we they cannot be optimal. Let $f(X)$ be as before and $g(X) = \|X\|_1$. Take a convex combination of these vectors,

$$f(\alpha X\widehat{\beta}_1 + (1-\alpha)X\widehat{\beta}_2) + g(\alpha X\widehat{\beta}_1 + (1-\alpha)X\widehat{\beta}_2)$$

By strict convexity of f and convexity of g

$$< \alpha f(X\widehat{\beta}_1) + (1-\alpha) f(X\widehat{\beta}_2) + \alpha g(X\widehat{\beta}_1) + (1-\alpha) g(X\widehat{\beta}_2)$$

However, this shows that $\alpha X\widehat{\beta}_1 + (1-\alpha)X\widehat{\beta}_2$ is more optimal than $X\widehat{\beta}_1$ and $X\widehat{\beta}_2$ which contradicts the assumption that they were optimal. Therefore, we must have that $X\widehat{\beta}_1 = X\widehat{\beta}_2$.

2) We have f and g as defined in the last part. The subgradient of the problem is

$$\partial(f+g) = -X^T(Y - X\beta) + \lambda \partial g$$

We have that $a \in \partial g(\beta)$ if for all γ

$$\|\gamma\|_1 - \|\beta\|_1 \geq a^T(\gamma - \beta)$$
$$\sum_{j=1}^{n} |\gamma_j| - |\beta_j| \geq a_j^T(\gamma_j - \beta_i)$$

That is, we require that $a_j \in \partial|\beta_j|$ for all j. This is much easier to see that

$$a_j \in \partial|\beta_j| = \begin{cases} \{1\} & \text{if } \beta_j > 0 \\ \{-1\} & \text{if } \beta_j < 0 \\ [-1,1] & \text{if } \beta_j = 0 \end{cases}$$

Thus, we have that $a \in \partial\|\beta\|_1 = g$ if for all j

$$a \in \partial|\beta_j| = \begin{cases} \{1\} & \text{if } \beta_j > 0 \\ \{-1\} & \text{if } \beta_j < 0 \\ [-1,1] & \text{if } \beta_j = 0 \end{cases}$$

This means that $0 \in \partial(f+g)$ iff $X^T(Y - X\beta) \in \lambda\partial g = \lambda\partial\|\beta\|_1$. Equivalently, for all j,

$$\begin{cases} \lambda = X_j^T(Y - X\beta) & \text{if } \beta_j > 0 \\ \lambda = -X_j^T(Y - X\beta) & \text{if } \beta_j < 0 \\ \lambda \geq |X_j^T(Y - X\beta)| & \text{if } \beta_j = 0 \end{cases}$$

3) We will show that $\lambda \geq |X_j^T(Y - X\beta)|$ to conclude that $\beta_j = 0$ by part Q4.2.

$$\lambda > \|X^TY\|_\infty \geq |X_j^TY| \ \forall j$$

However, note that if $\beta = 0$, we have that $X\beta = 0$, and so

$$|X_j^TY| = |X_j^T(Y - X\beta)| \ \forall j$$

That is, if $\beta = 0$,

$$\lambda > |X_j^T(Y - X\beta)| \ \forall j$$

And by Q4.2, we have that $\widehat{\beta} = 0$ is an optimal solution. By Q4.1, we have that two optimal solutions have the same prediction. That is $X\widehat{\beta}_1 = X\widehat{\beta}_2$. However, we showed $\widehat{\beta}_1 = 0$ is optimal and the optimal value is $\frac{1}{2}\|Y\|_2^2$. So, $X\widehat{\beta}_2 = 0$. This implies that

$$\frac{1}{2}\|Y - X\widehat{\beta}_2\|_2^2 + \lambda\|\widehat{\beta}_2\|_1 = \frac{1}{2}\|Y\|_2^2$$
$$\implies \lambda\|\widehat{\beta}_2 = 0$$

Since $\lambda > 0$, we conclude that $\widehat{\beta}_2 = 0$ and so $\widehat{\beta} = 0$ is unique.

4) A good guess for γ_0 is the linear interpolation between the end points $\widehat{\beta}(\lambda_1)$ and $\widehat{\beta}(\lambda_0)$. That is

$$\gamma_0 = \frac{\widehat{\beta}(\lambda_1) - \widehat{\beta}(\lambda_0)}{\lambda_0 - \lambda_1}$$

We then wish to show that

$$f(\lambda) = \widehat{\beta}(\lambda_0) - (\lambda - \lambda_0)\gamma_0$$
$$= \frac{\lambda_1 - \lambda}{\lambda_1 - \lambda_0}\widehat{\beta}(\lambda_0) + \frac{\lambda - \lambda_0}{\lambda_1 - \lambda_0}\widehat{\beta}(\lambda_1)$$

Satisfies the optimality conditions in Q4.2. Notice that we have

$$\begin{cases} \widehat{\beta}_i > 0 \Leftrightarrow f(\lambda)_i > 0 \\ \widehat{\beta}_i < 0 \Leftrightarrow f(\lambda)_i < 0 \\ \widehat{\beta}_i = 0 \Leftrightarrow f(\lambda)_i = 0 \end{cases}$$

Then let I_-, I_0, and I_+ denote the set of indices that are negative, zero, and positive respectively. Now, for $i \in I_+$, we have $f(\lambda)_i > 0$. Then we have

$$X_j^T(Y - Xf(\lambda)) = \frac{\lambda_1 - \lambda}{\lambda_1 - \lambda_0}X_j^T(Y - X\widehat{\beta}(\lambda_0)) + \frac{\lambda - \lambda_0}{\lambda_1 - \lambda_0}X_j^T(Y - X\widehat{\beta}(\lambda_1))$$

Now note that $\lambda_1 = X_j^T(Y - X\widehat{\beta}(\lambda_1))$ and $\lambda_0 = X_j^T(Y - X\widehat{\beta}(\lambda_0))$. So,

$$= \frac{\lambda_1 - \lambda}{\lambda_1 - \lambda_0}\lambda_0 + \frac{\lambda - \lambda_0}{\lambda_1 - \lambda_0}\lambda_1 = \lambda$$

Thus, $f(\lambda)$ is optimal in the set of indices I_+. A similar sequence of computation follows for I_-. The case I_0 is different. Thus, we have that $\lambda_1 \geq |X_j^T(Y - X\widehat{\beta}(\lambda_1))|$ and $\lambda_0 \geq |X_j^T(Y - X\widehat{\beta}(\lambda_0))|$. Working in the oppositive direction as I_+ we get

$$\begin{aligned}
\lambda &= \frac{\lambda_1 - \lambda}{\lambda_1 - \lambda_0}\lambda_0 + \frac{\lambda - \lambda_0}{\lambda_1 - \lambda_0}\lambda_1 \\
&\geq \frac{\lambda_1 - \lambda}{\lambda_1 - \lambda_0}|X_j^T(Y - X\widehat{\beta}(\lambda_0))| + \frac{\lambda - \lambda_0}{\lambda_1 - \lambda_0}|X_j^T(Y - X\widehat{\beta}(\lambda_1))| \\
&\geq |X_j^T(Y - X(\frac{\lambda - \lambda_0}{\lambda_1 - \lambda_0}\widehat{\beta}(\lambda_1) + \frac{\lambda_1 - \lambda}{\lambda_1 - \lambda_0}\widehat{\beta}(\lambda_0)))| \\
&= |X_j^T(Y - Xf(\lambda))|
\end{aligned}$$

Where we used the triangle inequality to combine the absolute values. Thus, $f(\lambda)$ is an optimal solution for all $\lambda \in [\lambda_0, \lambda_1]$. By assumption, there is a unique minimizer. Thus,

$$\widehat{\beta}(\lambda) = f(\lambda) = \widehat{\beta}(\lambda_0) - (\lambda - \lambda_0)\gamma_0$$

The Lasso regularization path in Q1.3 was also piecewise linear and has to do with the fact that if $\widehat{\beta}(\lambda)$ is piecewise linear, then we have that the L1 norm is a sum of piecewise linear functions which is piecewise linear.

5) One can argue that $\lim_{\lambda \to 0+} \widehat{\beta}(\lambda)$ converges because it is a sequence of piece wise linear functions. That is, there will be small interval $(0, \epsilon)$ for ϵ sufficiently small, that the $\widehat{\beta}(\lambda)$ will be linear on. This is justified by the fact that there is only a finite number of intervals that $\widehat{\beta}(\lambda)$ is linear on due to the fact that there is only a finite number of indices.