Aaron Alaniz
aa28877

Data Mining - Homework 2

1. All code related to this question can be found in colleges.r and is commented appropriately.
    a. College name is a categorical variable that will not contribute to any viable model.
    b. According to the R output below only 11 of the components are needed to capture 95% of the variance in the normalized data.

```
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4     Comp.5     Comp.6     Comp.7
Standard deviation     2.2724559 2.1403191 1.09721448 1.03136927 0.97495076 0.87191235 0.80241989
Proportion of Variance 0.3044143 0.2700419 0.07096712 0.06270505 0.05603243 0.04481463 0.03795574
Cumulative Proportion  0.3044143 0.5744562 0.64542335 0.70812840 0.76416083 0.80897546 0.84693120
                           Comp.8     Comp.9    Comp.10    Comp.11    Comp.12    Comp.13     Comp.14
Standard deviation     0.77196908 0.70241585 0.66152123 0.62721077 0.54915097 0.43784094 0.303572067
Proportion of Variance 0.03512966 0.02908458 0.02579655 0.02319002 0.01777697 0.01130074 0.005432475
Cumulative Proportion  0.88206086 0.91114543 0.93694199 0.96013201 0.97790898 0.98920972 0.994642191
                          Comp.15     Comp.16    Comp.17
Standard deviation     0.19981045 0.174093987 0.14372351
Proportion of Variance 0.00235348 0.001786659 0.00121767
Cumulative Proportion  0.99699567 0.998782330 1.00000000
```

    c. The data must be normalized beforehand because the components are not guaranteed to be in the same scale. If components are not on same scale then some components will skew the results of PCA. Also seen in the R output below, is that out of state tuition contributes the most to principal component 1, new students enrolled contributes the most to principal component 2, and estimated book costs contributes the most to principal component 3.

```
Loadings:
                          Comp.1 Comp.2 Comp.3
apps.received                    -0.420
apps.accepted                    -0.434
new.stud.enrolled                -0.446
X..new.stud.from.top.10. -0.354          0.120
X..new.stud.from.top.25. -0.340 -0.118   0.143
num.FT.undergrad                 -0.444
num.PT.undergrad          0.106 -0.288 -0.266
in.state.tuition         -0.379  0.150
out.of.state.tuition     -0.403
room                     -0.273        -0.251
board                    -0.290        -0.252
add.fees                         -0.169  0.250
est.book.costs                           -0.652
est.personal.costs        0.145 -0.157 -0.404
X..fac.with.PHD          -0.254 -0.197  0.189
stud.fac.ratio            0.279 -0.101  0.188
graduation.rate          -0.325         0.182
```

2. Feature selection is a special case of feature extraction in that we are looking to optimize the processing of our input data based on some characteristic of a subset of features. Feature selection is still desired over extraction in cases where the input data is sparse relative to the number of features. In other words, if our input data sample is small with many features it is useful to evaluate different subsets of the features with our model.

3. The derivative of the modified equation is the same as provide with with the corresponding weighting factor
   a. $\sum_{n=1}^{N} r_n[-t_n\varphi(x_n) + \varphi(x_n)\varphi(x_n)^T w]$ If we set this equal to zero then we can minimize the error.
   b. $\sum_{n=1}^{N} r_n[-t_n\varphi(x_n) + \varphi(x_n)\varphi(x_n)^T w] = 0$
   c. $\sum_{n=1}^{N} r_n\varphi(x_n)\varphi(x_n)^T w = \sum_{n=1}^{N} r_n t_n\varphi(x_n)$
   d. $w^* = \sum_{n=1}^{N} r_n t_n\varphi(x_n) / \sum_{n=1}^{N} r_n\varphi(x_n)\varphi(x_n)^T$ I don't know how to reduce this further into more readable terms but I'm pretty sure this is algebraically correct. I'm sure it would be rewritten in terms of some psuedo-inverse form but i'm unsure of how to do with r_n in the equation. For data dependent noise the derivative would be slightly different but still on the first order. The final solution would have an additional vector added at the end.

4. The bias will decrease and the variance will increase as k increases. As we increase the

complexity of our model we are overfitting the model to the sample data and introducing increased variance because it is not aware of the rest of the values of X. Conversely, the bias is increased and the variance is decreased with a less complex model due to an underfitting of the sample data provided. The critical point in this exercise is the where the value of k minimizes the total error introduced by our model. This identifies the value of k that establishes the regions of overfitting and underfitting.

5. All code for this question can be found in runs_per_game.r
    a. Below is the output of the ANOVA table from each linear model. OBP has the smallest mean squared error therefore I say it best explains RPG. This means that predicting RPG from OBP has the least amount compared to other predictors, therefore RPG is best explained by OBP

OBP

```
Analysis of Variance Table

Response: RPG
          Df Sum Sq Mean Sq F value    Pr(>F)
OBP        1 6.0909  6.0909  215.05 1.147e-14 ***
Residuals 28 0.7930  0.0283
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

SLG

```
Analysis of Variance Table

Response: RPG
          Df Sum Sq Mean Sq F value    Pr(>F)
SLG        1 5.2380  5.2380  89.109 3.383e-10 ***
Residuals 28 1.6459  0.0588
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

AVG

```
> anova(rpg_model_avg)
Analysis of Variance Table

Response: RPG
          Df Sum Sq Mean Sq F value    Pr(>F)
AVG        1 4.3891  4.3891  49.262 1.239e-07 ***
Residuals 28 2.4947  0.0891
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

b. Given that we have added OBP to our model then if we compare the two remaining combinations we find that OBP + SLG minimizes the squared the most. Therefore I would add SLG to my model over AVG. The two tables below show this.

OBP + AVG

```
Analysis of Variance Table

Response: RPG
          Df Sum Sq Mean Sq  F value    Pr(>F)
OBP        1 6.0909  6.0909 236.4389 7.051e-15 ***
AVG        1 0.0975  0.0975   3.7849    0.0622 .
Residuals 27 0.6955  0.0258
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

OBP + SLG

```
Analysis of Variance Table

Response: RPG
          Df Sum Sq Mean Sq  F value    Pr(>F)
OBP        1 6.0909  6.0909 275.7307 1.072e-15 ***
SLG        1 0.1966  0.1966   8.9008  0.005984 **
Residuals 27 0.5964  0.0221
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

6. All code related to this question can be found in glmnet_regression.r (see outputs below)
   a. As lambda goes down the coefficients go up.
   b. As lambda goes down the coefficients go up.
   c. ….don’t have a clue on this...be gentle.

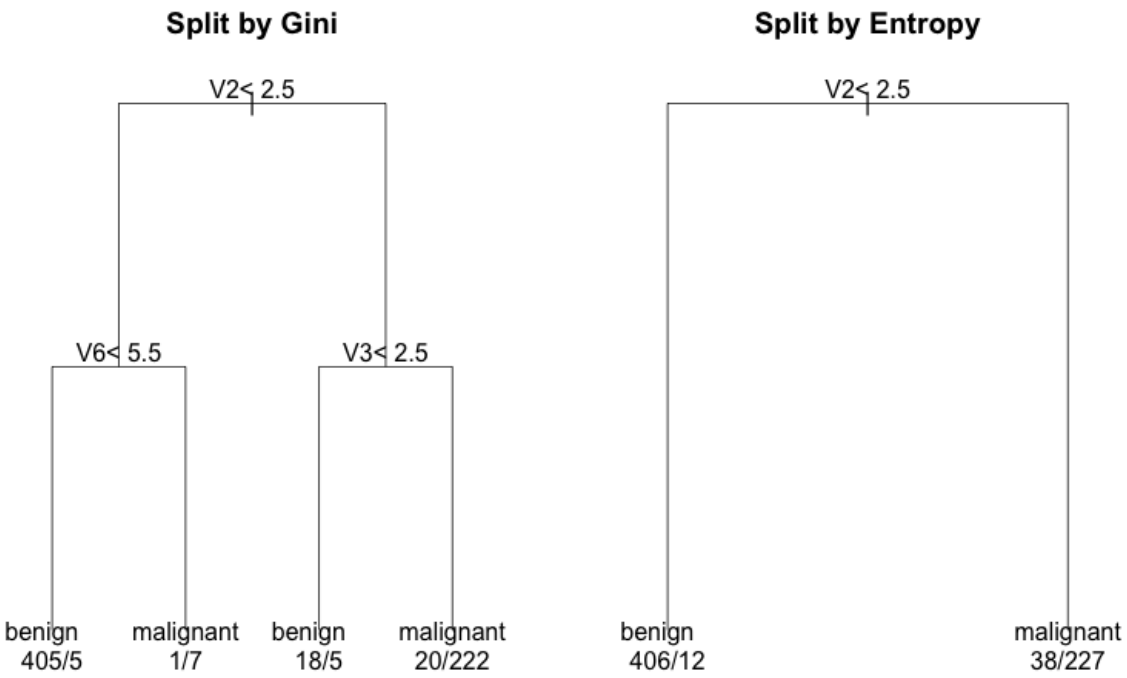
Lasso Regression

```
15 x 4 sparse Matrix of class "dgCMatrix"
                    s0        s1        s2         s3
(Intercept) 2.4524138 1.2262069 0.2452414 0.02452414
CRIM        .         .         .         .
ZN          .         .         .         .
INDUS       .         .         .         .
CHAS        .         .         .         .
NOX         .         .         .         .
RM          .         .         .         .
AGE         .         .         .         .
DIS         .         .         .         .
RAD         .         .         .         .
TAX         .         .         .         .
PTRATIO     .         .         .         .
B           .         .         .         .
LSTAT       .         .         .         .
MEDV        0.8911625 0.9455813 0.9891163 0.99891163
```

Ridge Regression

```
15 x 4 sparse Matrix of class "dgCMatrix"
                    s0           s1            s2            s3
(Intercept)  6.728462425  4.736869440  1.414429e+00  2.346933e-01
CRIM        -0.021305644 -0.014782944 -4.307795e-03 -8.801680e-04
ZN           0.007535268  0.005566098  1.685095e-03  3.230587e-04
INDUS       -0.011598528 -0.005368946  4.177796e-04  3.916187e-04
CHAS         0.759522404  0.460520465  1.097905e-01  1.020967e-02
NOX         -2.688260829 -2.145838743 -7.119412e-01 -1.281546e-01
RM           1.047578128  0.637115960  1.548461e-01  1.356168e-02
AGE         -0.001323307 -0.000545534 -8.343331e-05  7.948191e-05
DIS         -0.259732898 -0.192391188 -5.699074e-02 -8.063546e-03
RAD          0.031804220  0.028393358  1.131490e-02  3.034526e-03
TAX         -0.001233685 -0.001029792 -4.295259e-04 -1.139676e-04
PTRATIO     -0.214747367 -0.139708952 -3.798119e-02 -4.998571e-03
B            0.002326684  0.001449357  3.704799e-04  3.969330e-05
LSTAT       -0.117761883 -0.076383843 -2.015065e-02 -2.516477e-03
MEDV         0.740239955  0.840775663  9.602620e-01  9.954839e-01
```

7. All code related to this question can be found in decision_trees.r (See plots below)
    a. The GINI tree correctly classifies 423 benign and 229 malignant tumors while misclassifying 21 benign as malignant and 10 malignant as benign. In contrast the entropy tree correctly classifies 406 benign and 227 malignant tumors while misclassifying.
    b. I'm not sure on this one...


Split by Gini
V2< 2.5
V6< 5.5
V3< 2.5
benign
405/5
malignant
1/7
benign
18/5
malignant
20/222
Split by Entropy
V2< 2.5
benign
406/12
malignant
38/227