USC ID: 9907399097

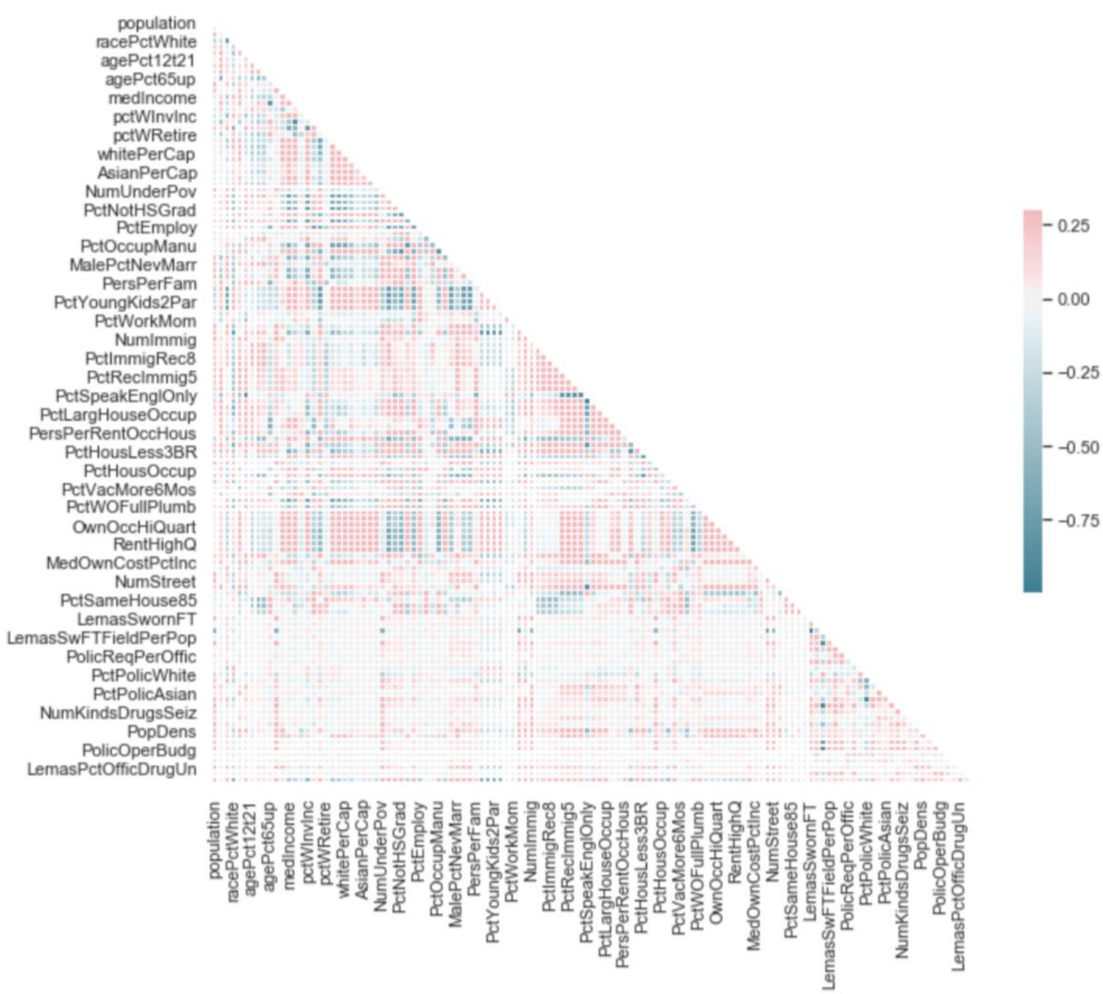
Name: Subhiksha Rani

Homework-4 Report

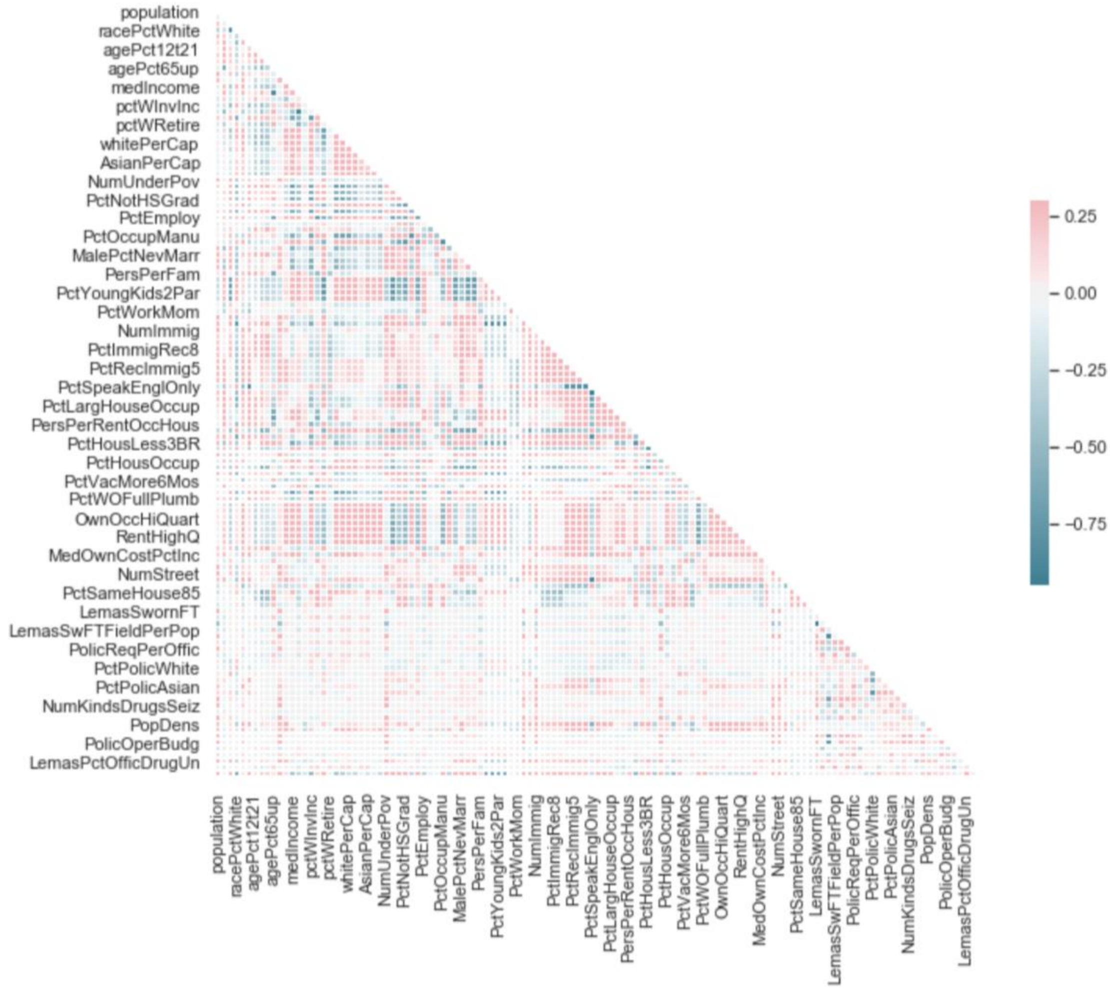
1) The LASSO and Boosting for Regression
   a) Downloading the dataset.

   b) Mean imputation method has been used to deal with missing values

   c) Correlation matrix for features in training set:



Correlation matrix for features in test set:



d) Coefficient of Variation

```
Coefficient of Variation CV for each feature in Training Set:
```

population	householdsize	racepctblack	racePctWhite	racePctAsian	racePctHisp	agePct12t21	agePct12t29	agePct16t24	agePct65up	...	LandArea	PopDens
2.240355	0.355681	1.428407	0.330102	1.358645	1.611552	0.36896	0.291217	0.500143	0.412638	...	1.644857	0.86421

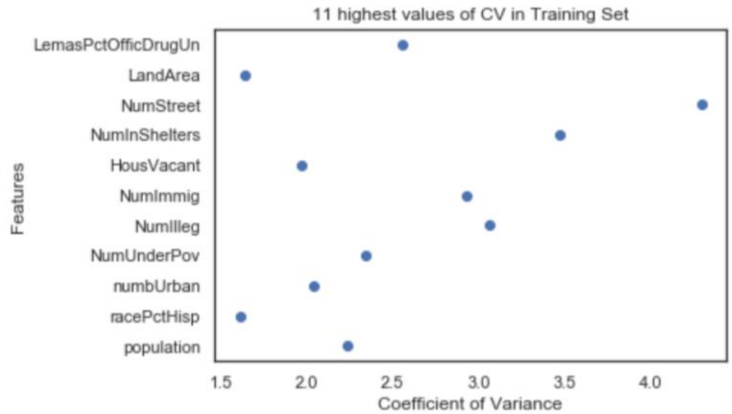
```
Coefficient of Variation CV for each feature in Test Set:
```

population	householdsize	racepctblack	racePctWhite	racePctAsian	racePctHisp	agePct12t21	agePct12t29	agePct16t24	agePct65up	...	LandArea	PopDens
0.24106	0.056078	0.329385	0.069985	0.284373	0.371586	0.054464	0.041658	0.078091	0.084183	...	0.212176	0.184859

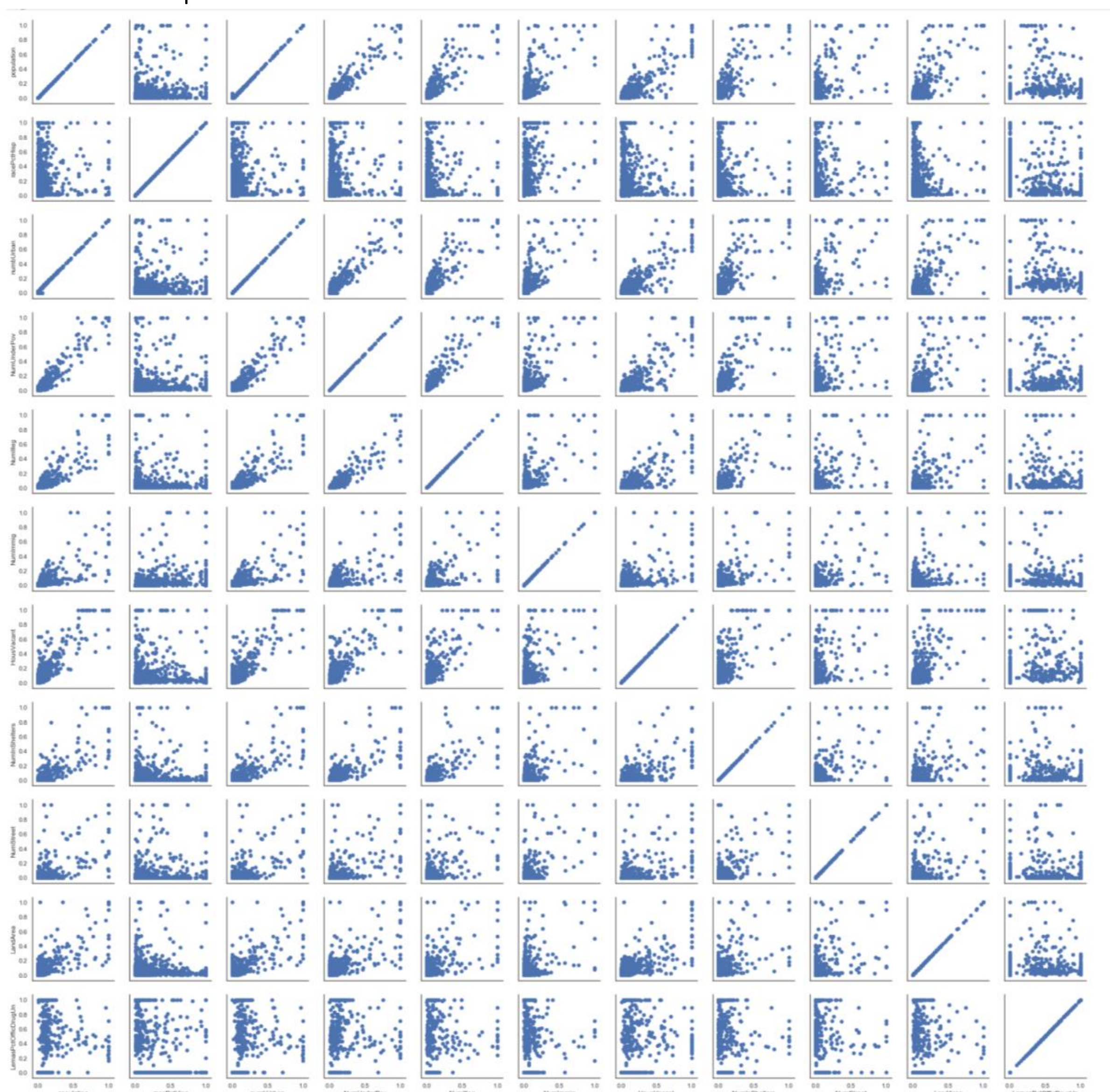
e) 11 items with highest CV in Training set:

```
11 items with highest CV in Training Set:  {'population': 2.240354965879167, 'racePctHisp': 1.6115517541662485, 'numbUrban': 2.037779618174568,
'NumUnderPov': 2.341659559375309, 'NumIlleg': 3.0579411110718357, 'NumImmig': 2.925656274779147, 'HousVacant': 1.9678085888226329, 'NumInShelter
s': 3.4697910919591677, 'NumStreet': 4.291486989120396, 'LandArea': 1.644857464527919, 'LemasPctOfficDrugUn': 2.552091540968875}
```

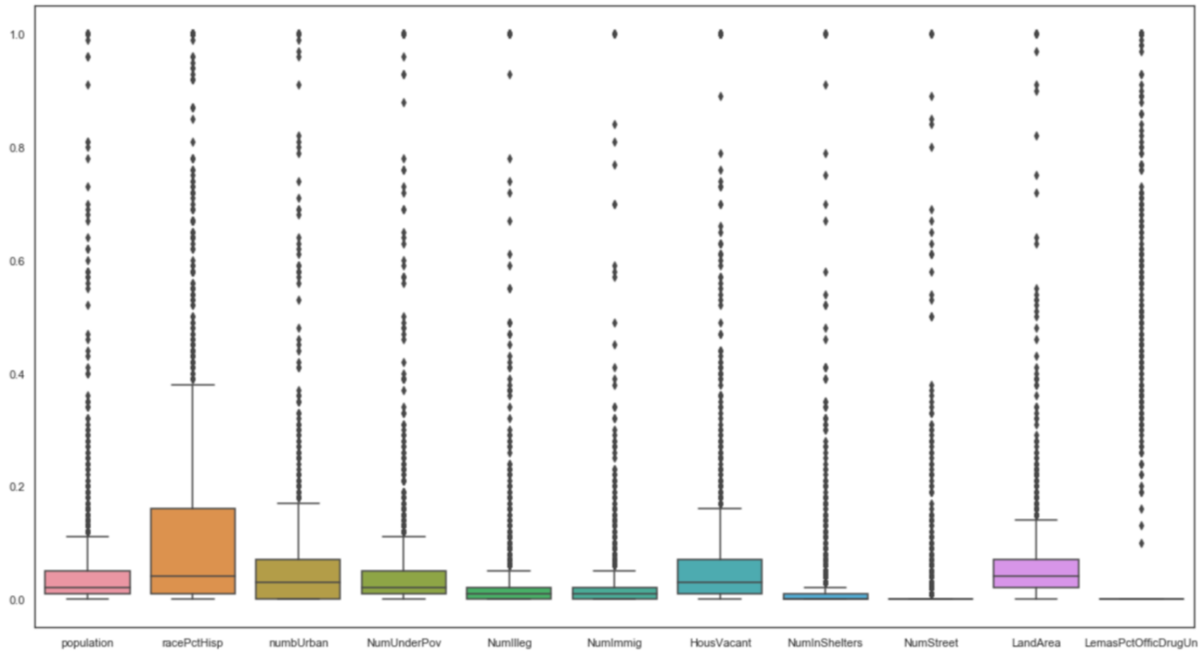
Scatter plots for the above features and their CV values:



Pairwise scatter plots for the above selected features & all their values:



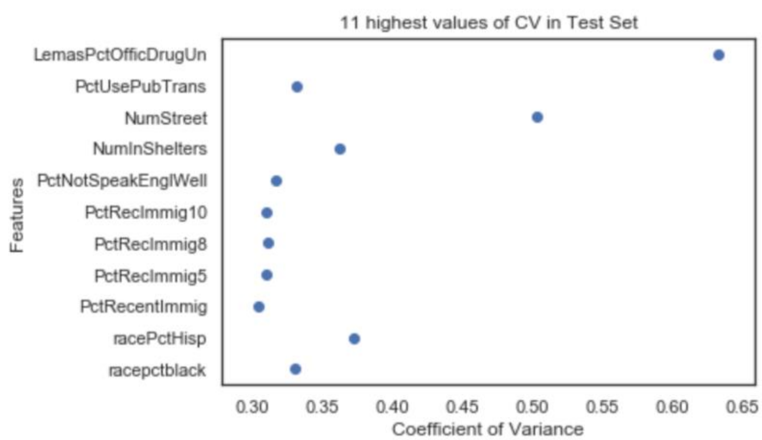
Boxplots for the above selected features & all their values:



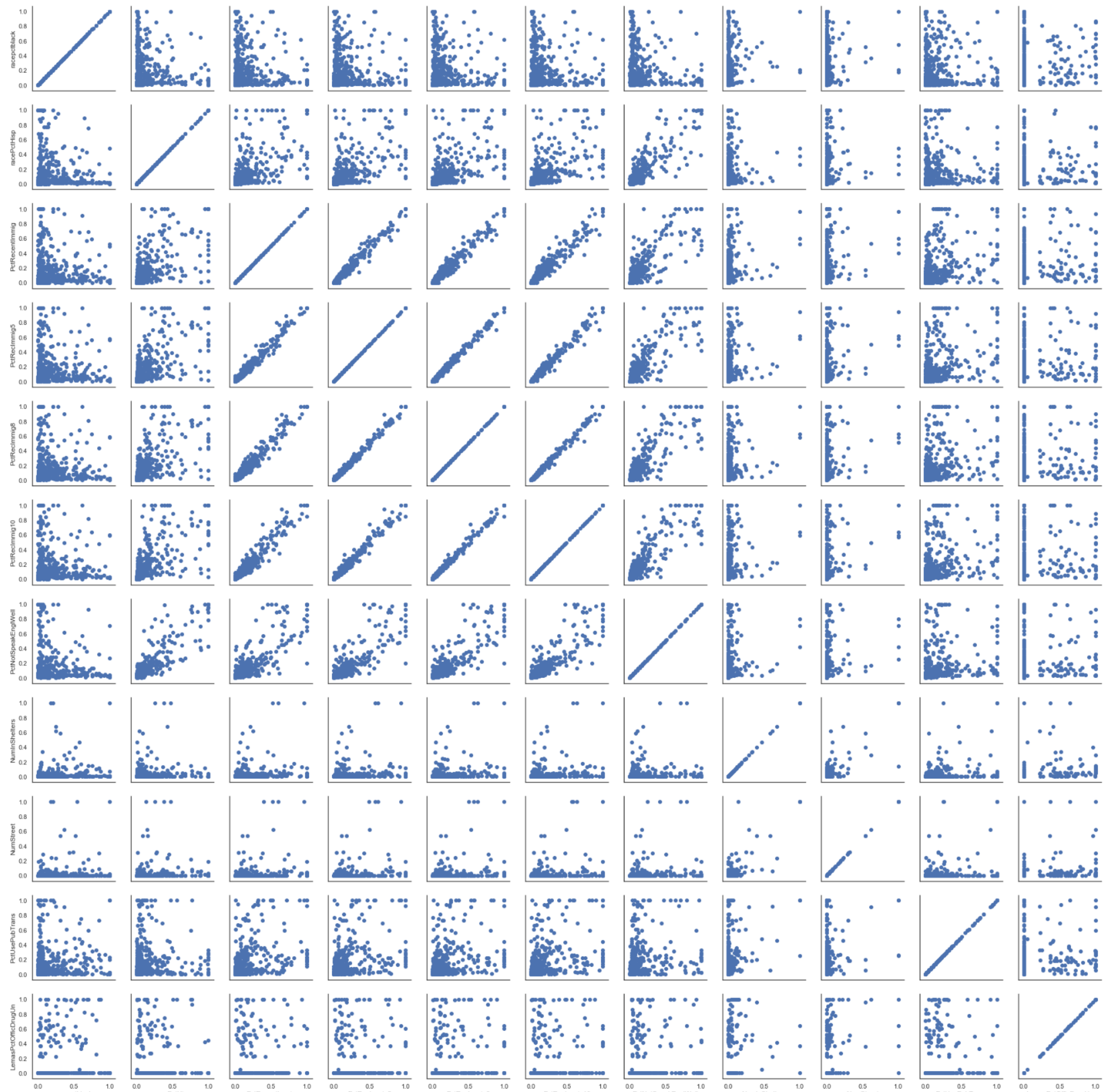
11 items with highest CV in Test set:

```
11 items with highest CV in Test Set:  {'racepctblack': 0.3293848884116853, 'racePctHisp': 0.3715859691961144, 'PctRecentImmig': 0.30419962604981443, 'PctRecImmig5':
0.3090746231725096, 'PctRecImmig8': 0.31089242049302973, 'PctRecImmig10': 0.30931650812816325, 'PctNotSpeakEnglWell': 0.31591256159729636, 'NumInShelters': 0.3615451
7030058014, 'NumStreet': 0.5024230012273304, 'PctUsePubTrans': 0.33088616430457335, 'LemasPctOfficDrugUn': 0.6326651872840013}
```

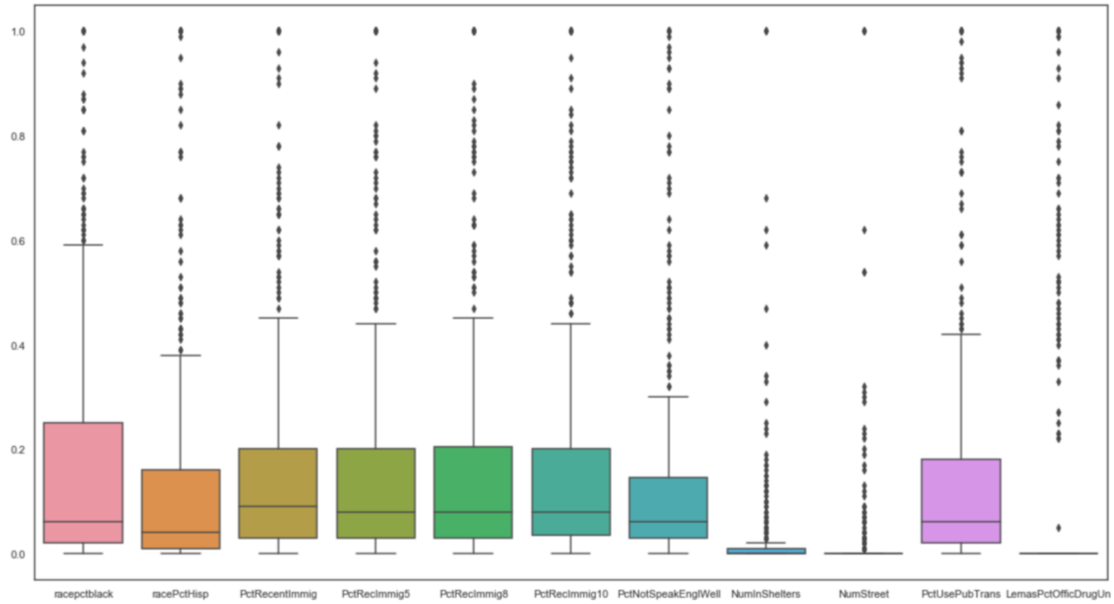
Scatter plots for the above features and their CV values:



Pairwise scatter plots for the above selected features & all their values:



Boxplots for the above selected features & all their values:



From the above scatter plots, we can see that while most of the values follow a linear pattern, others are spread across throughout the graph. Also, as per definition of Coefficient of Variation, higher the coefficient of variation, the greater the level of dispersion around the mean.

f) Mean squared error obtained by fitting a linear model using least squares:
0.7897255274361263

g) Lambda chosen by ridge cross validation: 0.0466301673441609
Mean squared error obtained by fitting a Ridge Regression Model using the lambda value obtained by ridge cross validation: 0.017562554133114135

h) Mean squared error obtained by fitting a LASSO model on the training set, with lambda chosen by LASSO cross-validation: 0.017774613570741355

```
List of variables selected by the model:
array([-0.        ,  0.        ,  0.27249895, -0.        , -0.01665789,
        0.        ,  0.07109539, -0.27784678,  0.        ,  0.0011751 ,
       -0.        ,  0.04073145,  0.        , -0.08891917,  0.02771806,
       -0.10354325,  0.05391932,  0.04011279, -0.10917987,  0.        ,
       -0.        , -0.06268167, -0.0217659 , -0.03149086,  0.02913329,
        0.02311615,  0.02483343, -0.        , -0.14530918, -0.01971014,
        0.        ,  0.00743498, -0.        ,  0.14574506, -0.01741799,
        0.        ,  0.01310179,  0.        ,  0.12481517,  0.14171105,
       -0.08644452, -0.        ,  0.        , -0.        , -0.19722105,
       -0.03685726, -0.0107591 ,  0.        , -0.13195846, -0.06160317,
        0.09706191, -0.0750557 , -0.        , -0.        , -0.        ,
       -0.00535548, -0.        ,  0.        ,  0.        ,  0.03847502,
        0.        , -0.07598984, -0.        , -0.07663781,  0.05692306,
       -0.        , -0.        , -0.06082072,  0.22506971,  0.06096321,
        0.00468589,  0.12739235, -0.05984836,  0.        ,  0.07417148,
       -0.06134435, -0.01607479,  0.02542797, -0.02501277, -0.01552814,
       -0.        , -0.        , -0.1619326 , -0.        ,  0.        ,
        0.1670111 ,  0.09050384, -0.01572268, -0.07872944,  0.05915423,
        0.16826345,  0.05057059, -0.00066031, -0.        ,  0.02370337,
       -0.        , -0.        ,  0.        ,  0.        ,  0.07703976,
       -0.03093294,  0.        ,  0.09245849,  0.        , -0.01682801,
       -0.        , -0.02987966,  0.00189325,  0.0065833 ,  0.        ,
       -0.00505628, -0.03130874, -0.01443093,  0.03645794,  0.01123726,
       -0.04554653,  0.        , -0.        ,  0.02395635,  0.04084537,
        0.00131602,  0.        ])
```

Mean squared error obtained by fitting a LASSO model on standardized features, with lambda chosen by LASSO cross-validation: 0.01743236193299663

```
List of variables selected by the model:
array([ 0.00000000e+00, -0.00000000e+00,  2.01426606e-01, -1.65122575e-02,
        0.00000000e+00,  0.00000000e+00, -0.00000000e+00, -6.21841232e-02,
       -0.00000000e+00,  0.00000000e+00,  0.00000000e+00,  3.15010662e-02,
        0.00000000e+00, -0.00000000e+00,  0.00000000e+00, -1.36885581e-02,
        0.00000000e+00,  4.51117999e-03, -3.05682124e-02,  0.00000000e+00,
        0.00000000e+00,  0.00000000e+00, -0.00000000e+00, -4.26998799e-06,
        1.01466223e-02,  0.00000000e+00,  0.00000000e+00,  0.00000000e+00,
       -0.00000000e+00, -0.00000000e+00,  0.00000000e+00, -0.00000000e+00,
        0.00000000e+00,  0.00000000e+00, -0.00000000e+00, -0.00000000e+00,
        0.00000000e+00, -0.00000000e+00,  1.18132334e-01,  0.00000000e+00,
        0.00000000e+00,  0.00000000e+00,  0.00000000e+00, -0.00000000e+00,
       -2.49089093e-01, -0.00000000e+00, -0.00000000e+00, -0.00000000e+00,
       -6.39776748e-02,  0.00000000e+00,  1.51575917e-01, -0.00000000e+00,
       -0.00000000e+00, -0.00000000e+00, -0.00000000e+00, -0.00000000e+00,
        0.00000000e+00,  0.00000000e+00,  0.00000000e+00,  0.00000000e+00,
       -0.00000000e+00,  0.00000000e+00,  0.00000000e+00,  0.00000000e+00,
        0.00000000e+00, -0.00000000e+00,  0.00000000e+00, -0.00000000e+00,
        1.47871465e-01,  2.70632274e-02, -0.00000000e+00,  9.49705053e-02,
       -3.85053873e-02, -0.00000000e+00,  3.87190610e-02, -7.57426922e-03,
       -0.00000000e+00,  0.00000000e+00, -0.00000000e+00,  0.00000000e+00,
        0.00000000e+00,  0.00000000e+00, -0.00000000e+00,  0.00000000e+00,
        0.00000000e+00,  0.00000000e+00,  4.36337260e-02,  0.00000000e+00,
       -2.45378175e-02,  0.00000000e+00,  1.44807999e-01,  2.60558995e-02,
       -0.00000000e+00, -0.00000000e+00,  3.59220259e-03,  0.00000000e+00,
       -0.00000000e+00,  0.00000000e+00,  0.00000000e+00,  0.00000000e+00,
       -0.00000000e+00,  1.40066957e-02,  4.02887357e-02,  0.00000000e+00,
       -0.00000000e+00,  0.00000000e+00, -0.00000000e+00, -0.00000000e+00,
        0.00000000e+00, -0.00000000e+00, -0.00000000e+00, -0.00000000e+00,
       -0.00000000e+00,  0.00000000e+00,  0.00000000e+00, -0.00000000e+00,
        0.00000000e+00, -0.00000000e+00,  1.82415348e-03,  1.96770336e-02,
        4.45632839e-03,  1.04475884e-02])
```

Comparing the test error obtained for standardized features & without standardization, we can see that the error is reduced a little for standardized features.

i) M value selected by cross-validation: 96 with a minimum score of 0.0194459259850635
Mean squared error obtained by fitting a PCR model, with M chosen by cross-validation: 0.018819644010020897

j) Alpha value obtained by cross-validation: 1
Mean squared error obtained by fitting a l1-penalized XGBoost regressor, with alpha chosen by cross-validation: 0.030402779110923702

2) Tree-Based Methods
   a) Downloading the dataset.

   b) Data Preparation
      i) Data Imputation Techniques:
         (1) <u>Mean imputation:</u> Simply calculate the mean of the observed values for that variable for all individuals who are non-missing.
         (2) <u>Substitution:</u> Impute the value from a new individual who was not selected to be in the sample.
         (3) <u>Hot deck imputation:</u> A randomly chosen value from an individual in the sample who has similar values on other variables.
         (4) <u>Cold deck imputation:</u> A systematically chosen value from an individual who has similar values on other variables.
         (5) <u>Regression imputation:</u> The predicted value obtained by regressing the missing variable on other variables.
         (6) <u>Stochastic regression imputation:</u> The predicted value from a regression plus a random residual value.
         (7) <u>Interpolation and extrapolation:</u> An estimated value from other observations from the same individual. It usually only works in longitudinal data.

         In this assignment, I have chosen Mean imputation to deal with the missing values.

      ii) Coefficient of variation:

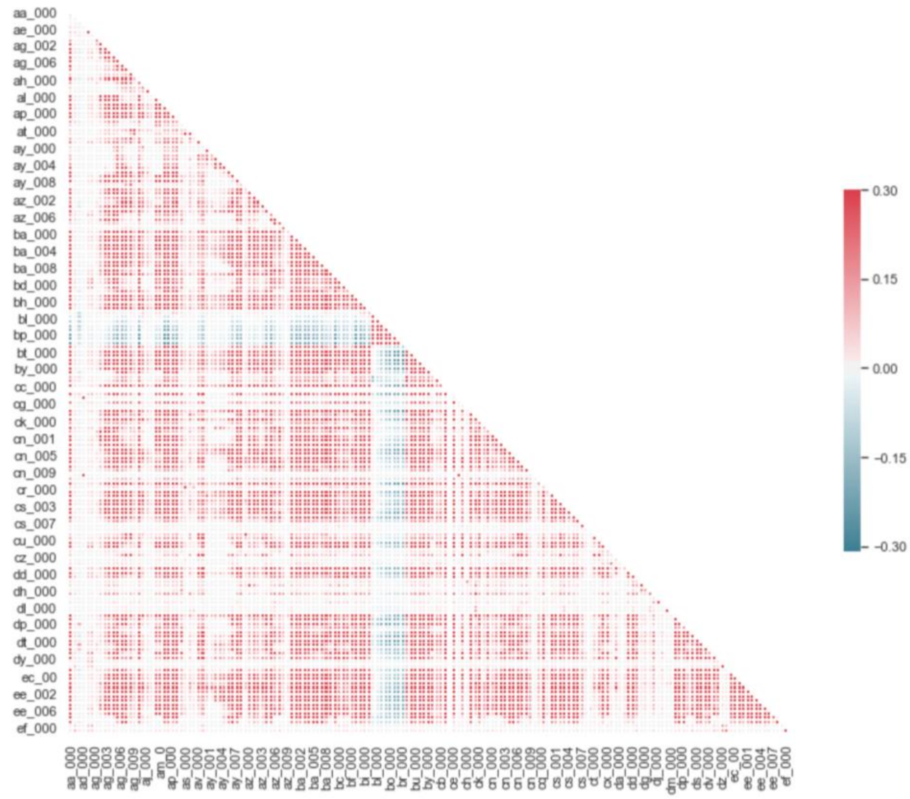
```
Coefficient of Variation CV for each feature in Training Set:
```

aa_000	ab_000	ac_000	ad_000	ae_000	af_000	ag_000	ag_001	ag_002	ag_003	...	ee_002	ee_003	ee_004	ee_005	ee_006	ee_007	ee_008	ee_009	ef_
2.450917	2.297616	2.169604	193.922717	23.202393	18.670817	91.976214	34.763409	17.344407	8.544525	...	2.579033	2.55869	2.606059	2.829163	3.191586	4.962235	3.22168	5.626661	47.25

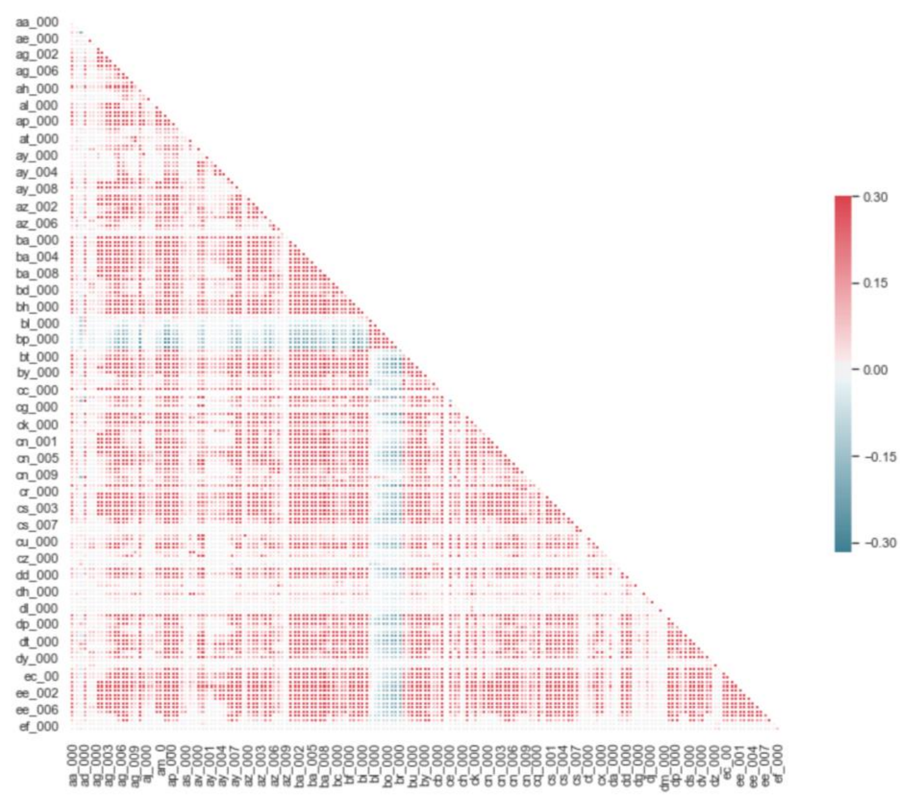
```
Coefficient of Variation CV for each feature in Test Set:
```

aa_000	ab_000	ac_000	ad_000	ae_000	af_000	ag_000	ag_001	ag_002	ag_003	...	ee_002	ee_003	ee_004	ee_005	ee_006	ee_007	ee_008	ee_009	ef_
7.405023	1.676951	2.160302	1.717723	17.946662	16.003112	52.679048	43.195742	17.057382	9.072105	...	2.684583	2.620214	2.687079	2.943193	3.351882	4.594078	3.699644	6.189823	49.948

iii) Correlation matrix for features in training set:



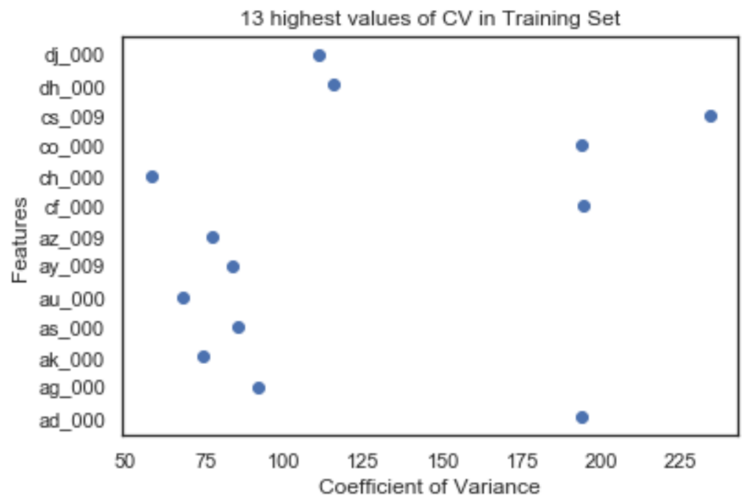
Correlation matrix for features in test set:



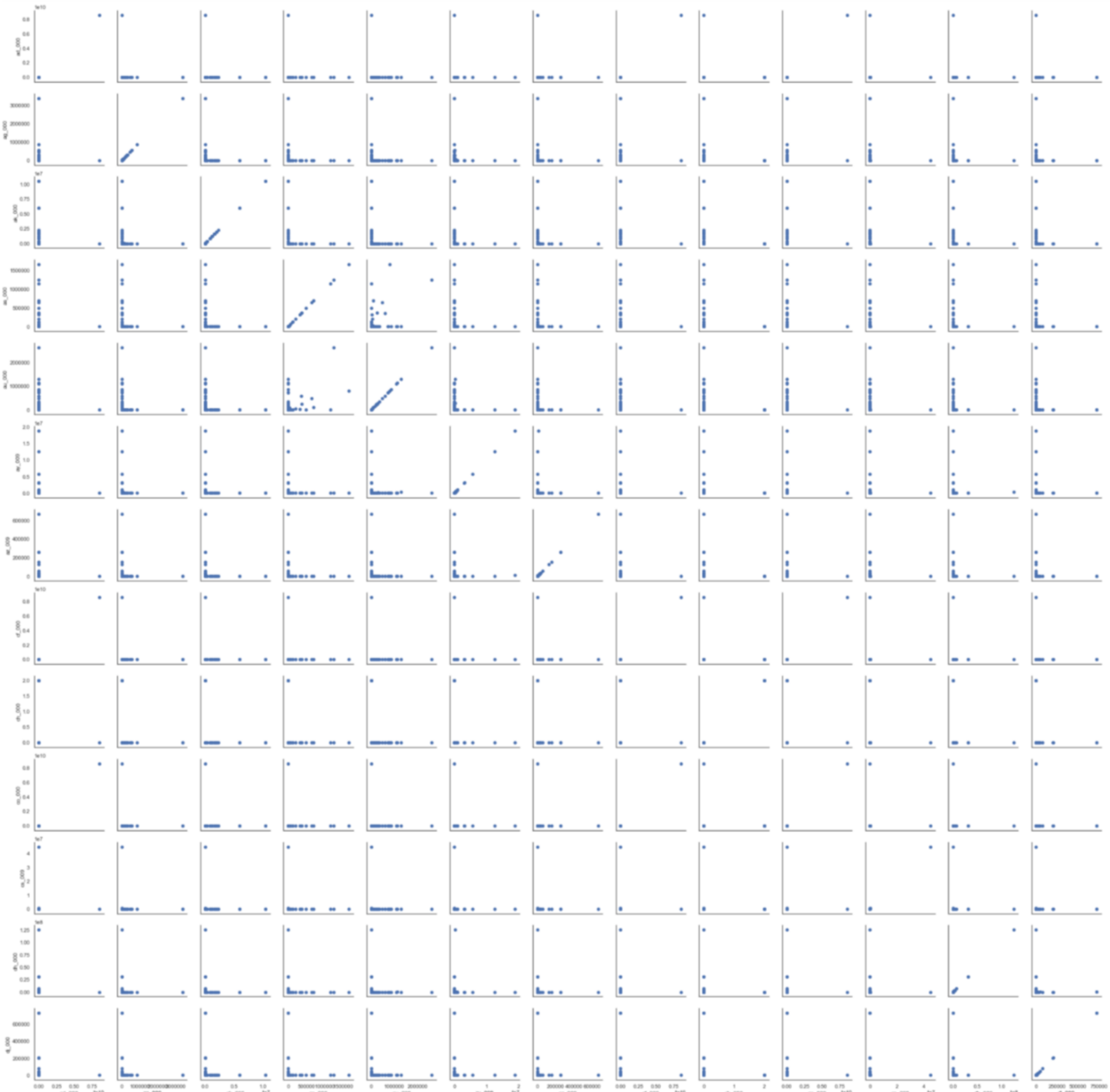
iv) Finding 13 items with highest CV in training set:

```
13 items with highest CV in Training Set:  {'ad_000': 193.92271732031412, 'ag_000': 91.97621400530639, 'ak_000': 74.58496957946791, 'as_000':
85.56270754186676, 'au_000': 67.94574598682685, 'ay_009': 83.78891005078252, 'az_009': 77.05880828113166, 'cf_000': 194.35164174549917, 'ch_0
00': 57.88579458513687, 'co_000': 194.03753604946613, 'cs_009': 234.45143434141406, 'dh_000': 115.647498393571, 'dj_000': 111.14757254382367}
```

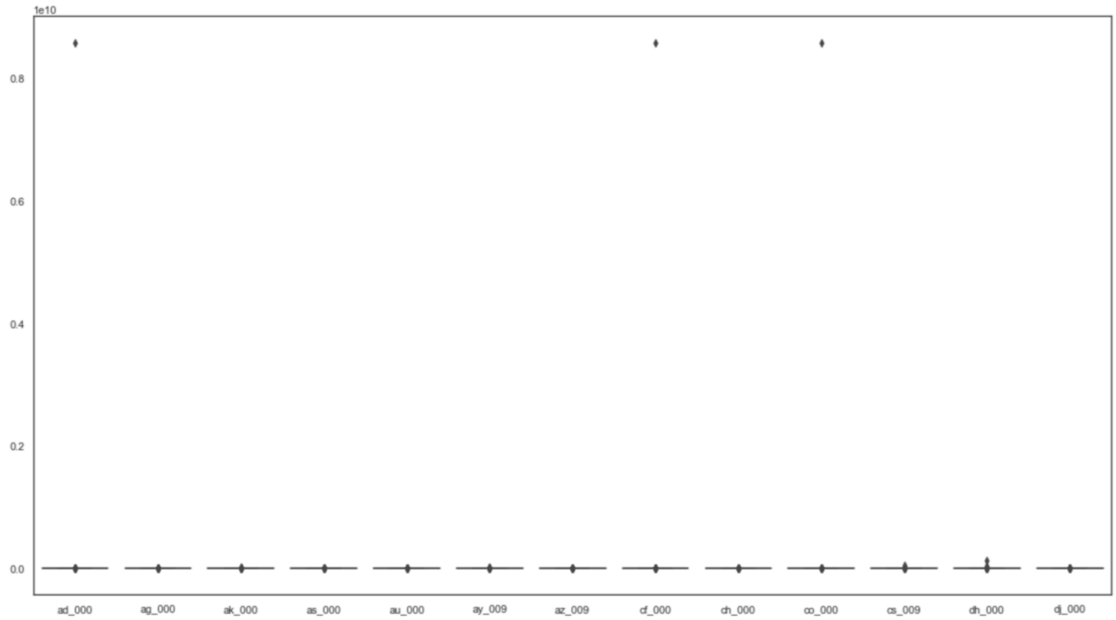
Scatter plots for the above selected features & their CV values:



Pairwise scatterplots for the above selected features and all their values:



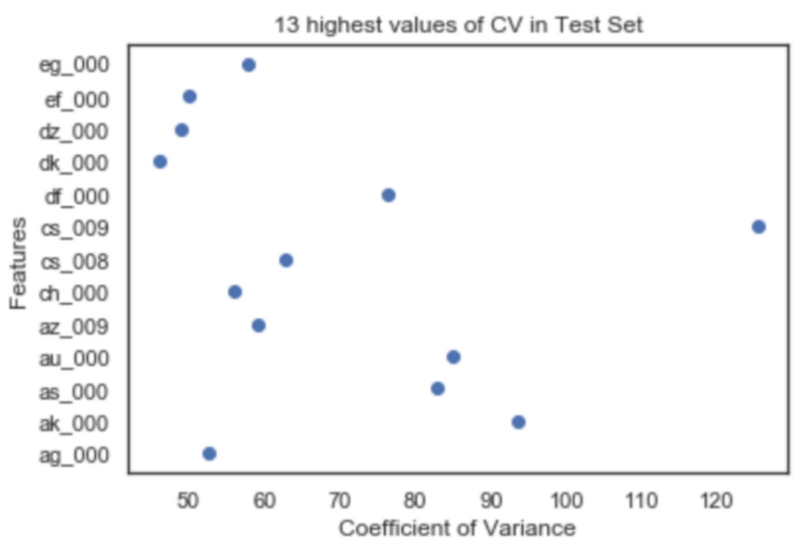
Boxplots for the above selected features & all their values:



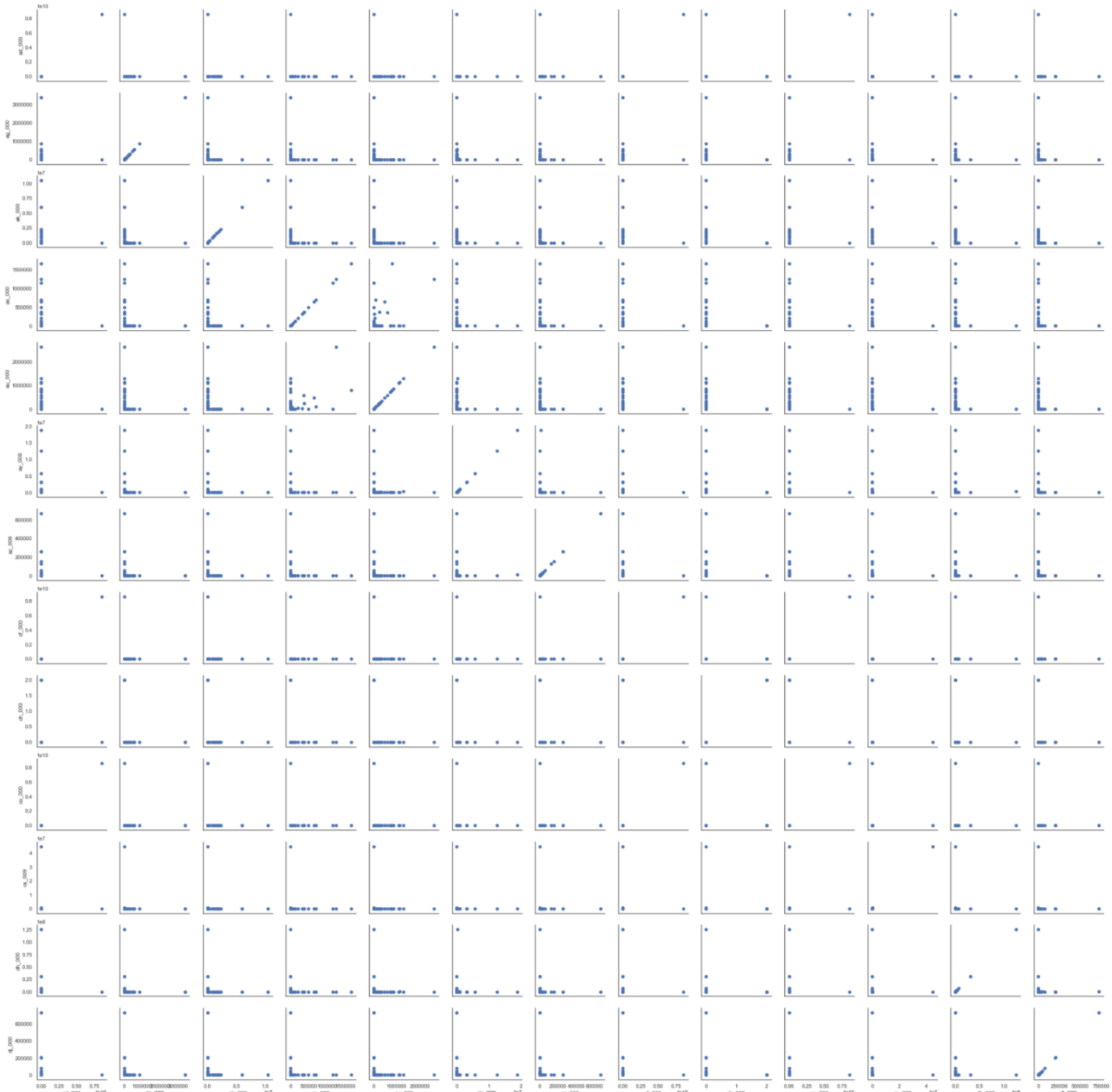
Finding 13 items with highest CV in test set:

```
13 items with highest CV in Test Set:  {'ag_000': 52.679048304298966, 'ak_000': 93.4122359471849, 'as_000': 82.9064319681767, 'au_000': 84.93287
614445322, 'az_009': 59.09817485274263, 'ch_000': 56.091469599127024, 'cs_008': 62.77987050938018, 'cs_009': 125.35516222509425, 'df_000': 76.39
997383607387, 'dk_000': 45.992092651265814, 'dz_000': 48.91520234442738, 'ef_000': 49.94822486440893, 'eg_000': 57.92830589349868}
```

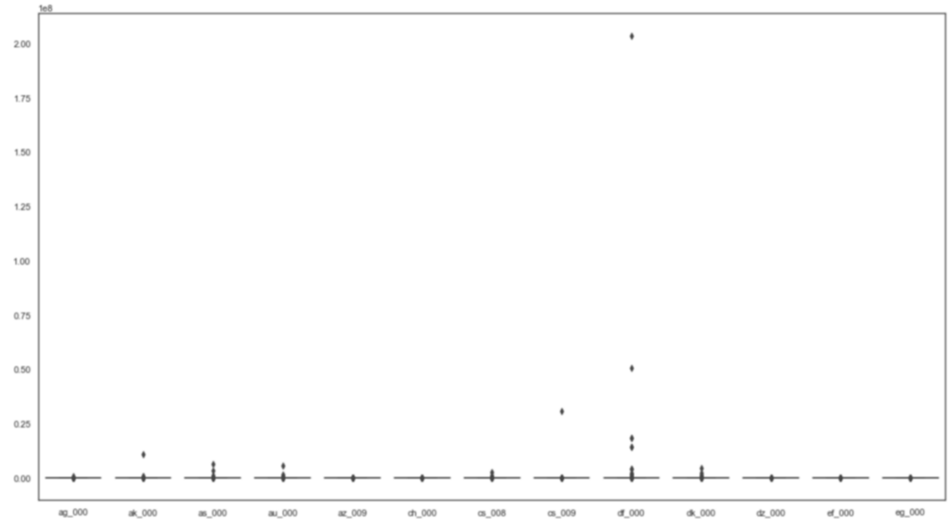
Scatter plots for the above selected features & their CV values:



Pairwise scatterplots for the above selected features and all their values:



Boxplots for the above selected features & all their values:



From the above scatter plots, we can see that all the values are very closely bound to each other and for few values any one of planes (x or y) value remains constant. The values are not spread apart.

v) Total number of positive & negative values in Training set:
Negative: 59000
Positive: 1000

Total number of positive & negative values in Test set:
Negative: 15625
Positive: 375

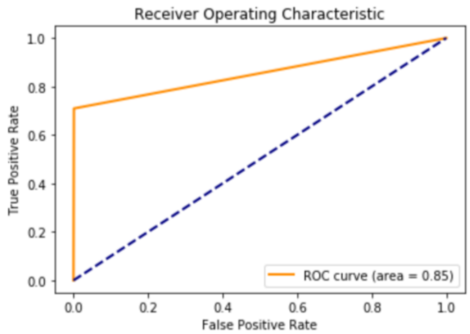
Yes, the dataset is Imbalanced because the number of negative values is lot more than the number of positive values.

c) Random Forest Classifier
Misclassification Rate: 0.7874999999999965

Confusion Matrix:[[15608,   17],
[109,   266]]

AUC: 0.8541226666666667

ROC Curve:



Out of bag error : 0.99405
Test error: 0.007875
Comparing the above Out of bag error to Test error, we can see that Test error is very low compared to Out of Bag error.

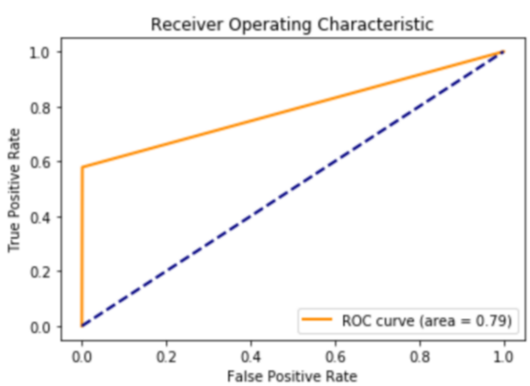
d) Random Forest Classifier by addressing class imbalance.
Dealing with class imbalance in Random Forest - In scikit learn library, the RandomForestClassifier has a parameter class_weight and if the value of this is equal to "balanced", then the classes are automatically weighted inversely proportional to how frequently they appear in the data.

Misclassification Rate: 1.0874999999999968

Confusion matrix: [[15609, 16],
[158, 217]]

AUC: 0.7888213333333334

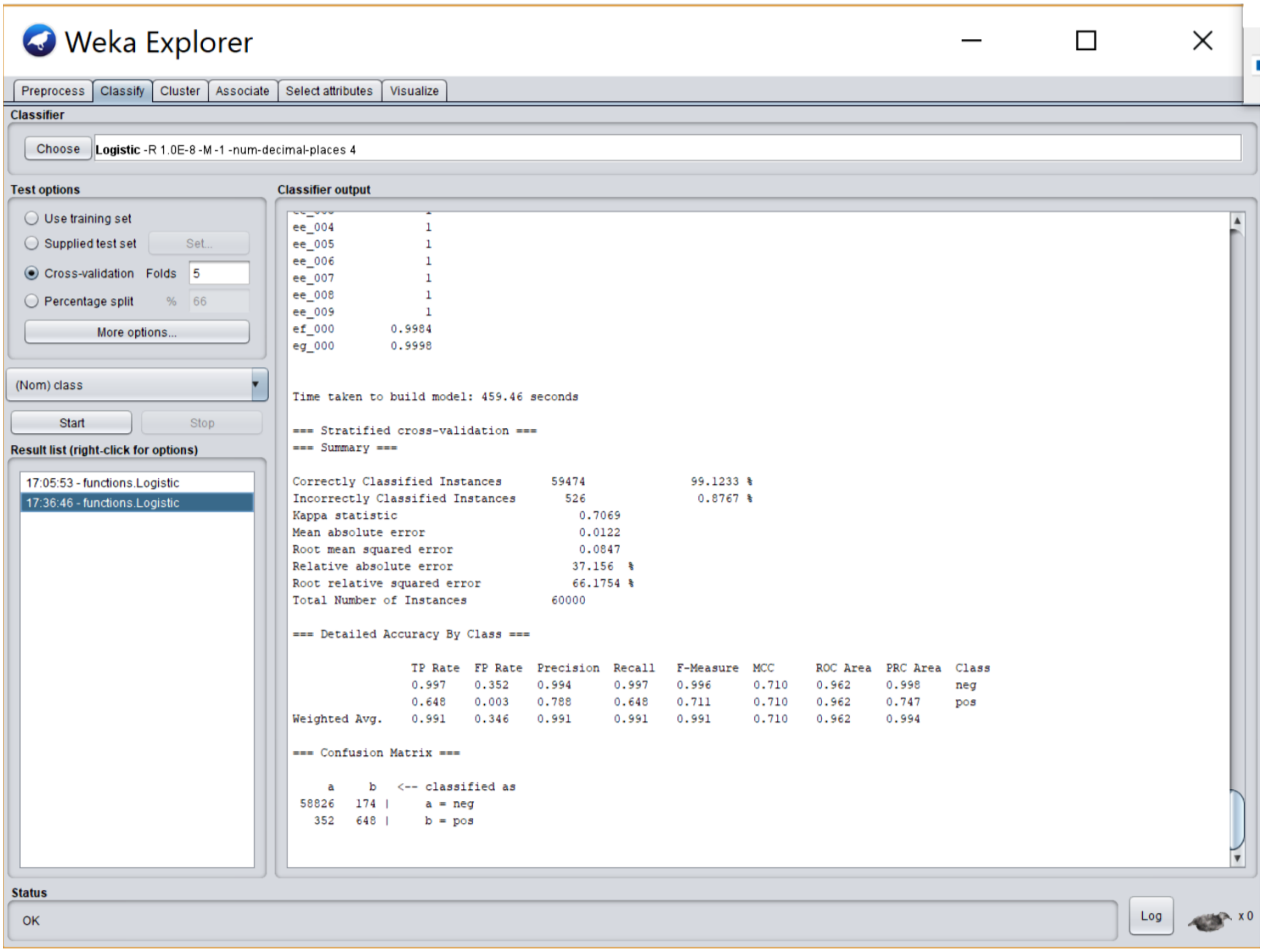
ROC curve:



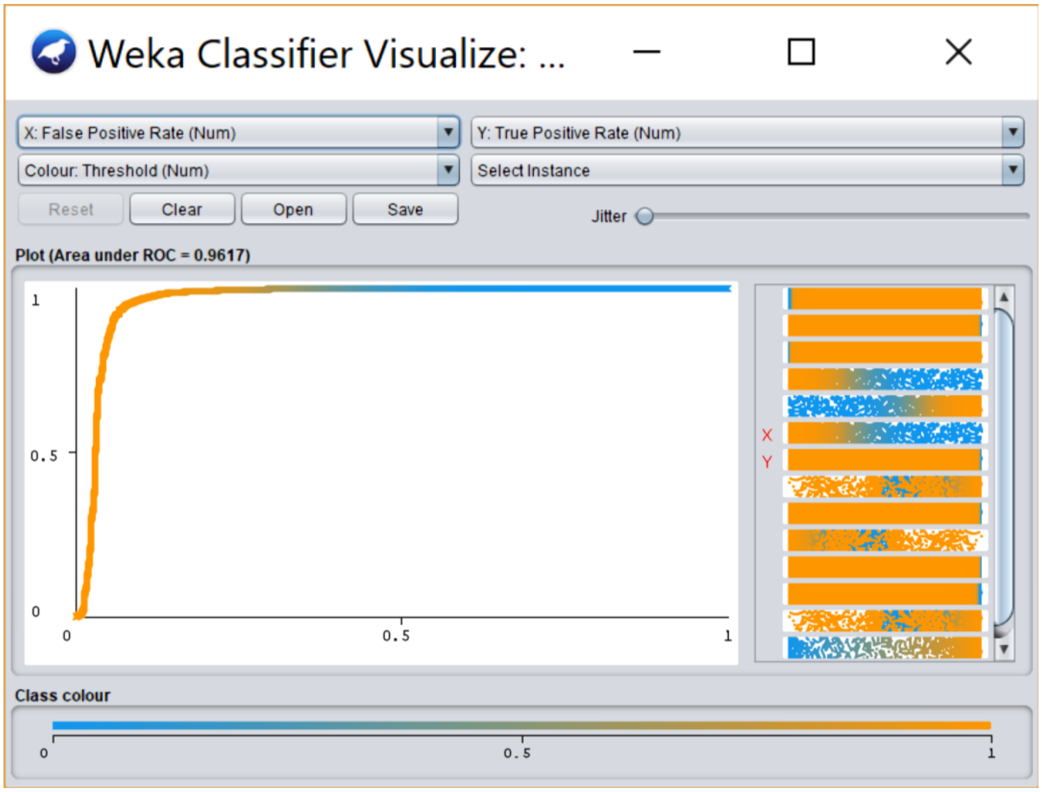
Out of bag error : 0.9920666666666667
Test error: 0.010875

Comparing the above Out of bag error to Test error, we can see that Test error is very low compared to Out of Bag error.
Comparing the balanced Random forest to the imbalanced Randon forest, we can see that the Misclassification rate & Test error has increased for the balanced class when compared to the imbalanced class

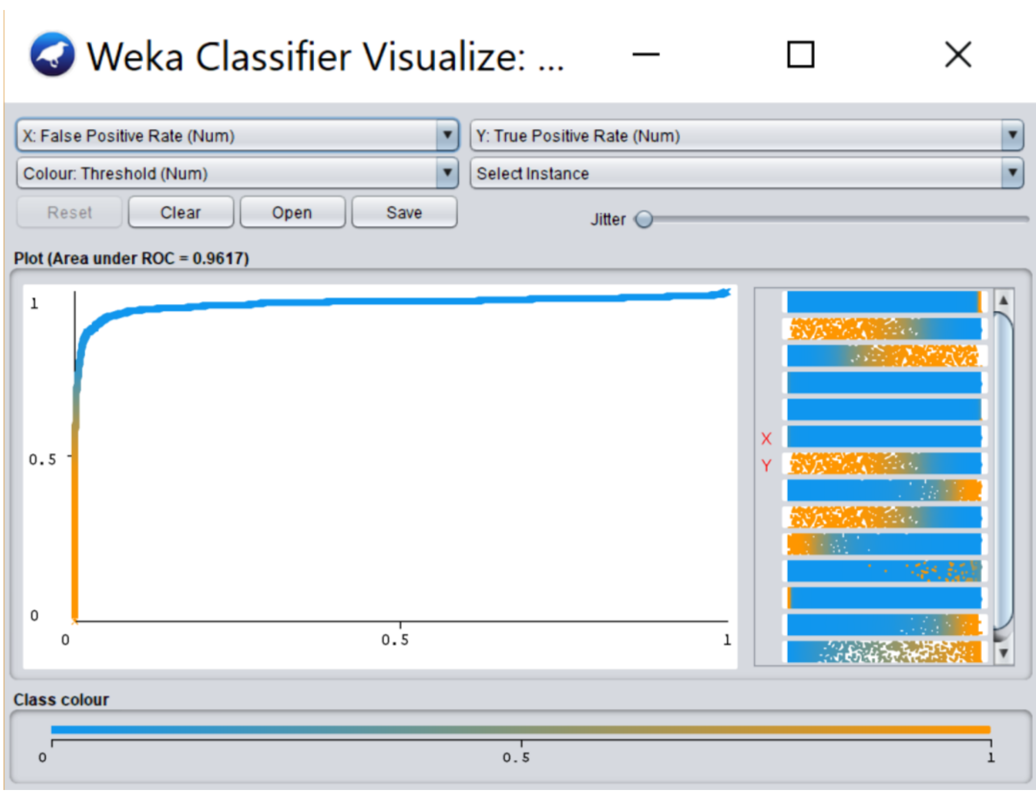
e) Model Trees using Weka & 5-fold cross validation



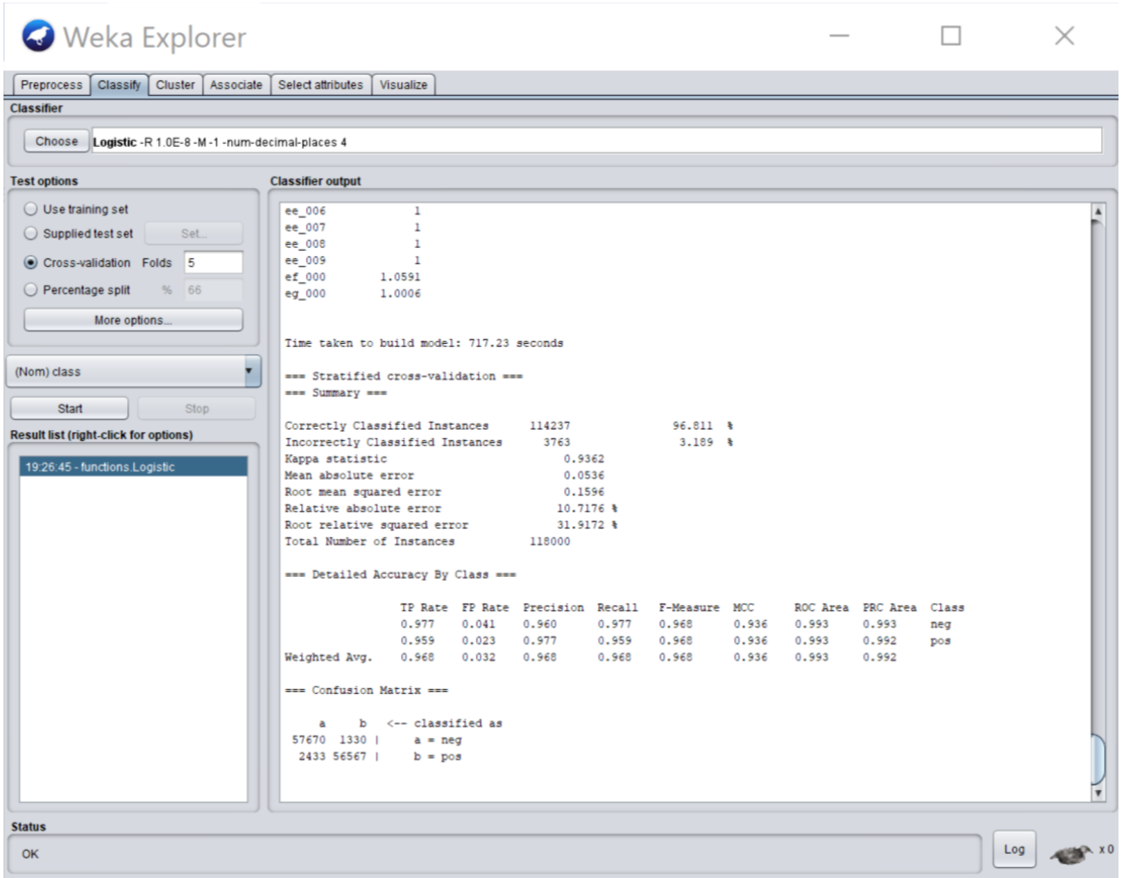
ROC Curve for Negative values: AUC: 0.9617



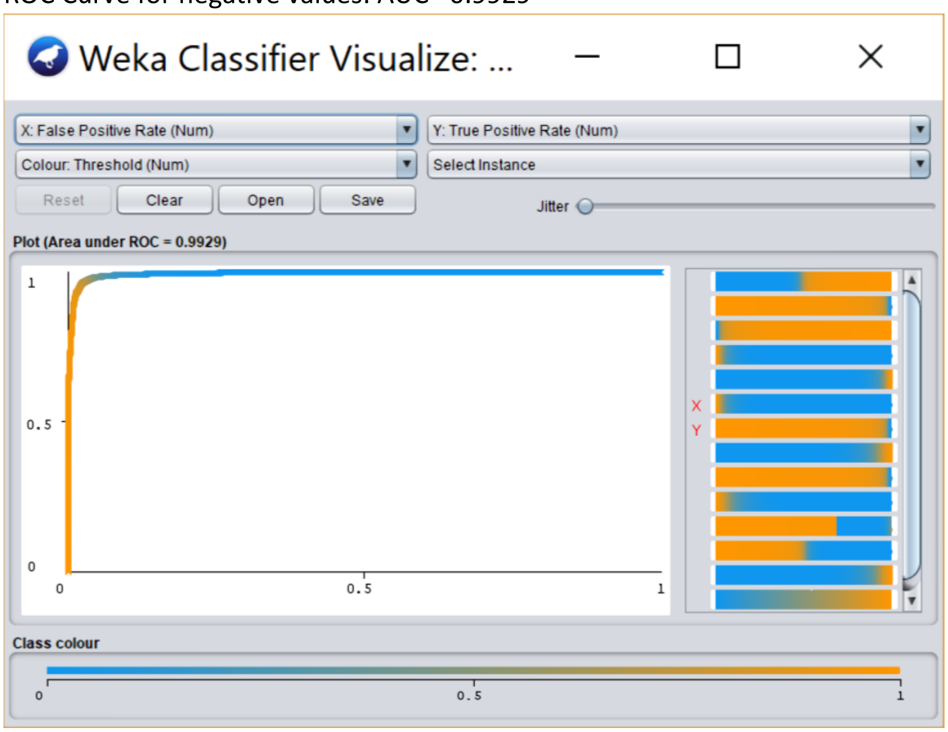
ROC Curve for Positive values: AUC: 0.9617



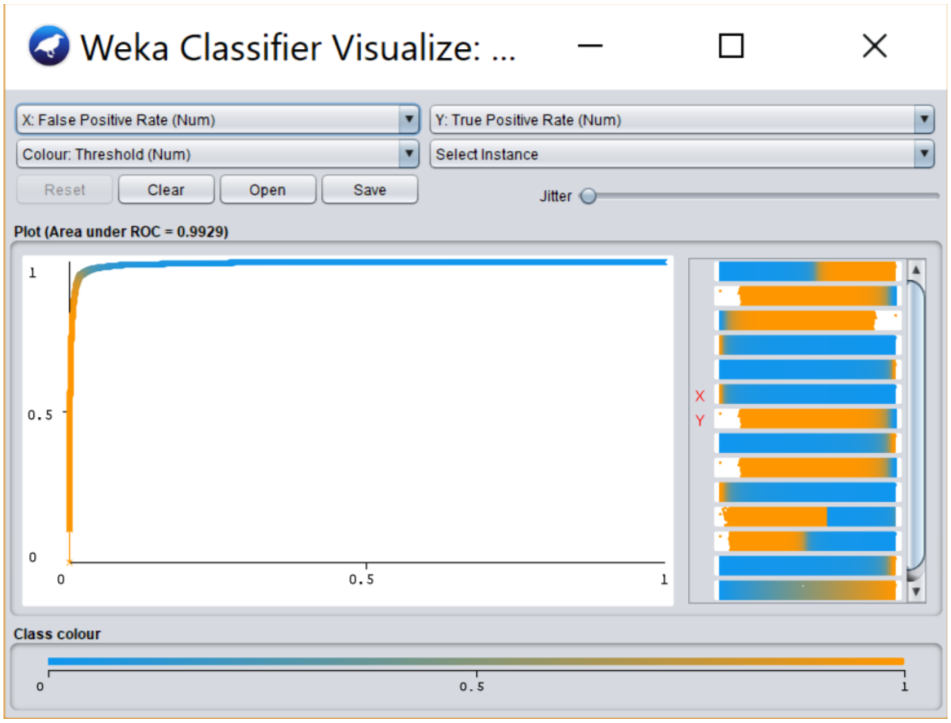
f) SMOTE to deal with class imbalance.
SMOTE oversampling is used to deal with minority class.



ROC Curve for negative values: AUC= 0.9929



ROC Curve for positive values: AUC – 0.9929



3) **ISLR 6.8.3**: Suppose we estimate the regression coefficients in a linear regression model by minimizing

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 \quad \text{subject to} \quad \sum_{j=1}^{p}|\beta_j| \leq s$$

for a particular value of s. For parts (a) through (e), indicate which of i. through v. is correct. Justify your answer.

a) As we increase s from 0, the training RSS will:
iv. <u>Steadily decrease</u> - As we increase s from 0, we are restricting the βj coefficients less and less (the coefficients will increase to their least squares estimates), and so the model is becoming more and more flexible which provokes a steady decrease in the training RSS.

b) Repeat (a) for test RSS:
ii. <u>Decrease initially, and then eventually start increasing in a U shape</u> - As we increase s from 0, we are restricting the βj coefficients less and less (the coefficients will increase to their least squares estimates), and so the model is becoming more and more flexible which provokes at first a decrease in the test RSS before increasing again after that in a typical U shape.

c) Repeat (a) for variance:

iii. <u>Steadily increase</u> - As we increase s from 0, we are restricting the βj coefficients less and less (the coefficients will increase to their least squares estimates), and so the model is becoming more and more flexible which provokes a steady increase in variance.

d) Repeat (a) for (squared) bias:
iv. Steadily decrease - As we increase s from 0, we are restricting the βj coefficients less and less (the coefficients will increase to their least squares estimates), and so the model is becoming more and more flexible which provokes a steady decrease in bias.

e) Repeat (a) for the irreducible error:
v. Remain constant - By definition, the irreducible error is independant of the model, and consequently independent of the value of s.

4) **ISLR 6.8.5:** It is well-known that ridge regression tends to give similar coefficient values to correlated variables, whereas the lasso may give quite different coefficient values to correlated variables. We will now explore this property in a very simple setting.
Suppose that $n = 2$, $p = 2$, $x_{11} = x_{12}$, $x_{21} = x_{22}$. Furthermore, suppose that $y_1+y_2 = 0$ and $x_{11}+x_{21} = 0$ and $x_{12}+x_{22} = 0$, so that the estimate for the intercept in a least square, ridge regression, or lasso model is zero: $\hat{\beta}_0 = 0$.

a) Write out the ridge regression optimization problem in this setting.

a) According to this setting ($x_{11} = x_{12} = x_1$ & $x_{21} = x_{22} = x_2$), the ridge regression problem seeks to m~~a~~inimize.

$$(y_1 - \hat{\beta}_1 x_1 - \hat{\beta}_2 x_1)^2 + (y_2 + \hat{\beta}_1 x_2 - \hat{\beta}_2 x_2)^2 + \lambda(\hat{\beta}_1^2 + \hat{\beta}_2^2).$$

b) Argue that in this setting, the ridge coefficient estimates satisfy $\hat{\beta}_1 = \hat{\beta}_2$.

b) By taking the derivatives of the above expression w.r.t. $\hat{\beta}_1$ & $\hat{\beta}_2$ and setting them equal to 0, we obtain

$$\hat{\beta}_1(x_1^2 + x_2^2 + \lambda) + \hat{\beta}_2(x_1^2 + x_2^2) = y_1x_1 + y_2x_2$$

and

$$\hat{\beta}_1(x_1^2 + x_2^2) + \hat{\beta}_2(x_1^2 + x_2^2 + \lambda) = y_1x_1 + y_2x_2$$

By subtracting the 2 expressions above, we get $\hat{\beta}_1 = \hat{\beta}_2$.

c) Write out the lasso optimization problem in this setting.

c) According to this setting ($x_{11} = x_{12} = x_1$ and $x_{21} = x_{22} = x_2$), the lasso optimization problem seeks to minimize.

$$(y_1 - \hat{\beta}_1x_1 - \hat{\beta}_2x_1)^2 + (y_2 - \hat{\beta}_1x_2 - \hat{\beta}_2x_2)^2 + \lambda(|\hat{\beta}_1| + |\hat{\beta}_2|).$$

d) Argue that in this setting, the lasso coefficients ˆ β1 and ˆ β2 are not unique—in other words, there are many possible solutions to the optimization problem in (c). Describe these solutions.

d) We will use the alternate form of the lasso optimization problem

$(y_1 - \hat{\beta}_1 x_1 - \hat{\beta}_2 x_1)^2 + (y_2 - \hat{\beta}_1 x_2 - \hat{\beta}_2 x_2)^2$ subject to $|\hat{\beta}_1| + |\hat{\beta}_2| \leq s$

Geometrically the lasso constraint take the form of a diamond centered at the origin of the plane $(\hat{\beta}_1, \hat{\beta}_2)$ which intersects the axes at a distance s from the origin. By using the setting of this problem ($x_{11} = x_{12} = x_1$, $x_{21} = x_{22} = x_2$, $x_1 + x_2 = 0$ and $y_1 + y_2 = 0$), we have to minimize the expression $2[y_1 - (\hat{\beta}_1 + \hat{\beta}_2)x_1]^2 \geq 0$

This optimization problem has a simple solution: $\hat{\beta}_1 + \hat{\beta}_2 = y_1 / x_1$. Geometrically, this is a line parallel to the edge of the diamond of the constraints. Now, solutions to the lasso optimization problem are contours of the function $[y_1 - (\hat{\beta}_1 + \hat{\beta}_2)x_1]^2$ that intersects the diamond of the constraints. So, the entire edge $\hat{\beta}_1 + \hat{\beta}_2 = s$ (as is the edge $\hat{\beta}_1 + \hat{\beta}_2 = -s$) is a potential solution to the lasso optimization problem. Thus, the lasso optimization problem has a whole set of solutions instead of a unique one:

$\{(\hat{\beta}_1; \hat{\beta}_2) : \hat{\beta}_1 + \hat{\beta}_2 = s$ with $\hat{\beta}_1, \hat{\beta}_2 \geq 0$ and $\hat{\beta}_1 + \hat{\beta}_2 = -s$ with $\hat{\beta}_1, \hat{\beta}_2 \leq 0\}$.

5) **ISLR 8.4.5:** Suppose we produce ten bootstrapped samples from a data set containing red and green classes. We then apply a classification tree to each bootstrapped sample and, for a specific value of X, produce 10 estimates of P(Class is Red|X):

0.1, 0.15, 0.2, 0.2, 0.55, 0.6, 0.6, 0.65, 0.7, and 0.75.

There are two common ways to combine these results together into a single class prediction. One is the majority vote approach discussed in this chapter. The second approach is to classify based on the average probability. In this example, what is the final classification under each of these two approaches?

<u>Ans</u>: With the majority vote approach, we classify X as Red as it is the most commonly occurring class among the 10 predictions (6 for Red vs 4 for Green). With the average probability approach, we classify X as Green as the average of the 10 probabilities is 0.45.

6) **ISLR 9.7.3:** Here we explore the maximal margin classifier on a toy data set.

a) We are given n = 7 observations in p = 2 dimensions. For each observation, there is an associated class label.

Obs.	X_1	X_2	Y
1	3	4	Red
2	2	2	Red
3	4	4	Red
4	1	4	Red
5	2	1	Blue
6	4	3	Blue
7	4	1	Blue

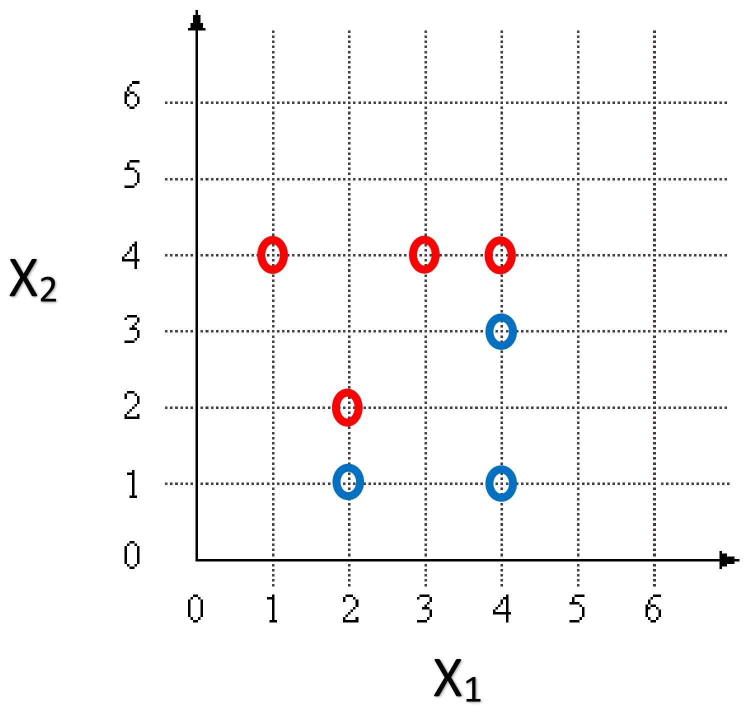
Sketch the observations.

<u>Ans</u>: x_1 = c (3, 2, 4, 1, 2, 4, 4)

X_2 = c (4, 2, 4, 4, 1, 3, 1)

colors = c ("red", "red", "red", "red", "blue", "blue", "blue")

plot(x_1, x_2, col = colors, xlim = c(0, 5), ylim = c(0, 5))



b) Sketch the optimal separating hyperplane, and provide the equation for this hyperplane (of the form (9.1)).

Ans: The maximal margin classifier has to be in between observations #2, #3 and #5, #6.

(2,2), (4,4)

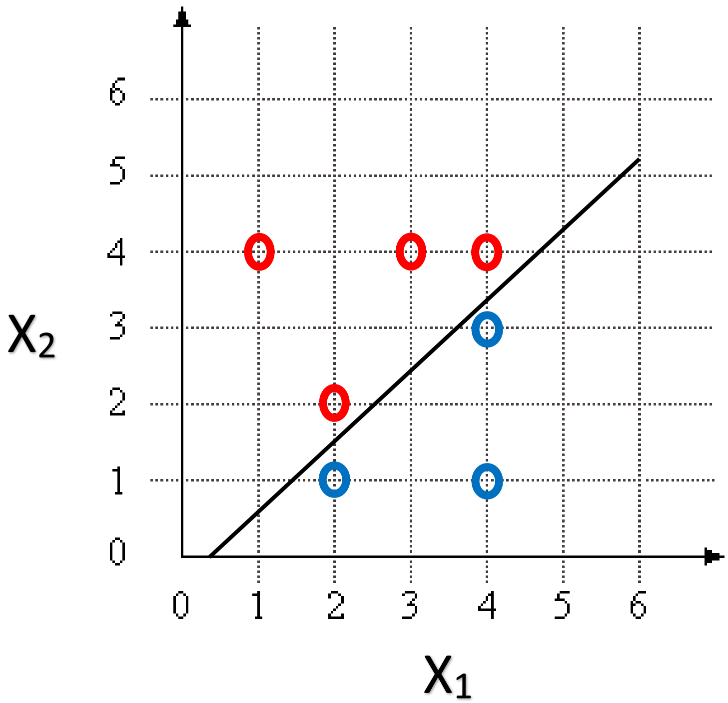
(2,1), (4,3)

=> (2,1.5),(4,3.5)

b= (3.5−1.5)/(4−2)=1

$a=x_2-x_1=1.5-2=-0.5$

```
plot(x1, x2, col = colors, xlim = c(0, 5), ylim = c(0, 5))
abline(-0.5, 1)
```



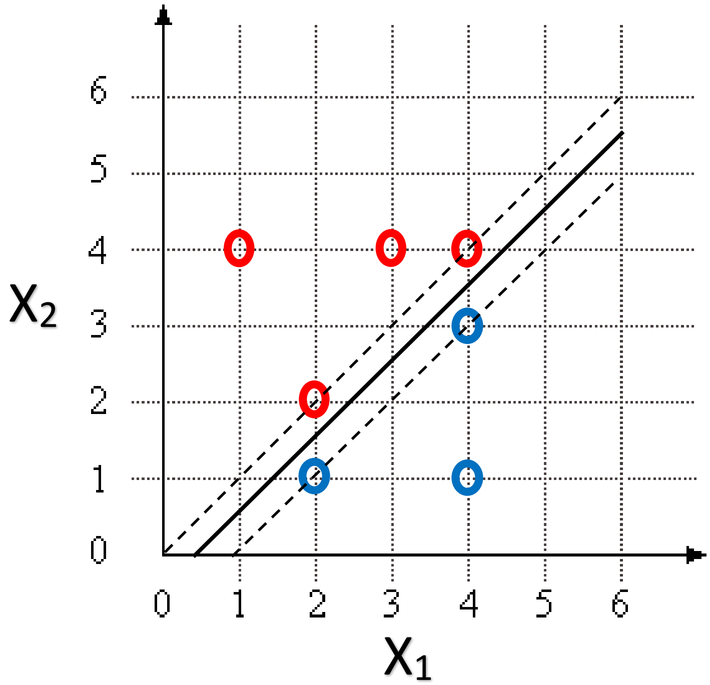
c) Describe the classification rule for the maximal margin classifier. It should be something along the lines of "Classify to Red if $\beta_0 + \beta_1 X_1 + \beta_2 X_2 > 0$ and classify to Blue otherwise." Provide the values for β_0, β_1, and β_2.

Ans: $0.5 - x_1 + x_2 > 0$

d) On your sketch, indicate the margin for the maximal margin hyperplane.

<u>Ans:</u>

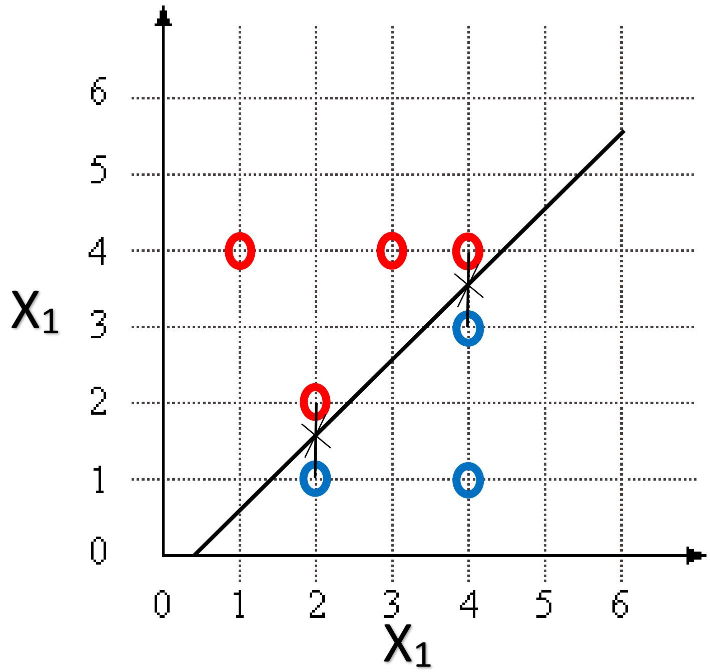
```
plot(x1, x2, col = colors, xlim = c(0, 5), ylim = c(0, 5))
abline(-0.5, 1)
abline(-1, 1, lty = 2)
abline(0, 1, lty = 2)
```



e) Indicate the support vectors for the maximal margin classifier.

Ans:

```
plot(x1, x2, col = colors, xlim = c(0, 5), ylim = c(0, 5))
abline(-0.5, 1)
arrows(2, 1, 2, 1.5)
arrows(2, 2, 2, 1.5)
arrows(4, 4, 4, 3.5)
arrows(4, 3, 4, 3.5)
```



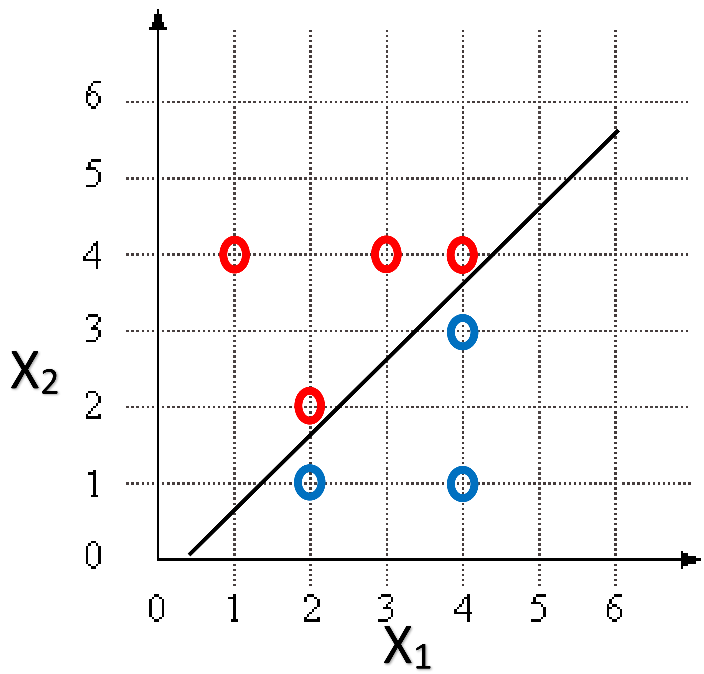
f) Argue that a slight movement of the seventh observation would not affect the maximal margin hyperplane.

Ans: A slight movement of observation #7 (4,1) blue would not have an effect on the maximal margin hyperplane since its movement would be outside of the margin.

g) Sketch a hyperplane that is not the optimal separating hyperplane and provide the equation for this hyperplane.

Ans:

```
plot(x₁, x₂, col = colors, xlim = c(0, 5), ylim = c(0, 5))
abline(-0.8, 1)
```



$-0.8 - X_1 + X_2 > 0$

h) Draw an additional observation on the plot so that the two classes are no longer separable by a hyperplane.

Ans:

```
plot(x1, x2, col = colors, xlim = c(0, 5), ylim = c(0, 5))
points(c(4), c(2), col = c("red"))
```