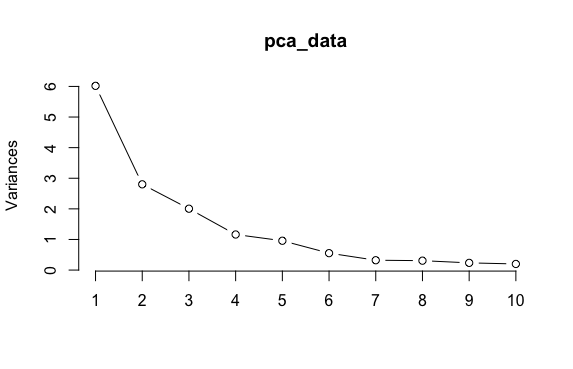
**Homework 4**

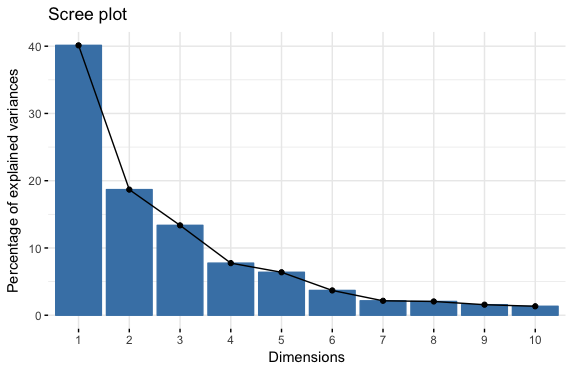
**Question 9.1 Using the same crime data set uscrime.txt as in Question 8.2, apply Principal Component Analysis and then create a regression model using the first few principal components. Specify your new model in terms of the original variables (not the principal components), and compare its quality to that of your solution to Question 8.2. You can use the R function prcomp for PCA. (Note that to first scale the data, you can include scale. = TRUE to scale as part of the PCA function. Don’t forget that, to make a prediction for the new city, you’ll need to unscale the coefficients (i.e., do the scaling calculation in reverse)!)**

As I write this narrative its pretty amazing how 7-9 hours of testing code, posting questions and even some brute force eigen calculations are now about to be reduced to an explanation paragraph and 20 lines of code.

For question 9.1 I read in the crime data and removed the response data to do PCA. I note from the instructor lecture that given the small data set threre is limited benefit in splitting data into a training set and validation set for the purpose of answering the question .

Remembering to scale the data I created the pca\_data set and applied three graphs

Screeplot, biplot and fviz\_eig



Based on the screeplot data and summary I initially chose the first 4 Principal components since 80 percent seemed like a reasonable threshold and subsequent components added less value.

Importance of components:

PC1 PC2 PC3 PC4 PC5 PC6 PC7 PC8 PC9 PC10 PC11

Standard deviation 2.4534 1.6739 1.4160 1.07806 0.97893 0.74377 0.56729 0.55444 0.48493 0.44708 0.41915

Proportion of Variance 0.4013 0.1868 0.1337 0.07748 0.06389 0.03688 0.02145 0.02049 0.01568 0.01333 0.01171

Cumulative Proportion 0.4013 0.5880 0.7217 0.79920 0.86308 0.89996 0.92142 0.94191 0.95759 0.97091 0.98263

PC12 PC13 PC14 PC15

Standard deviation 0.35804 0.26333 0.2418 0.06793

Proportion of Variance 0.00855 0.00462 0.0039 0.00031

Cumulative Proportion 0.99117 0.99579 0.9997 1.00000

When I analyzed the results of this initial model my R squared value was much lower then I expected until I realized the percentage of variance was not going to be equal to percentage of relevance to the predictor.

Residual standard error: 336.4 on 42 degrees of freedom

Multiple R-squared: 0.3091, Adjusted R-squared: 0.2433

F-statistic: 4.698 on 4 and 42 DF, p-value: 0.003178

At this point I looked at 5 predictors and R squared increased dramatically to 60 percent. Going to 6, 7 only increased it to 63 percent. Not wanting to overfit the model I settled on 5 since the benefit of incremental predictors was not apparent.

Now the fun part was deciphering how to translate the PCA coefficients into terms with the original predictors.

A nice tangent resulted in me transforming my data set back to the original data set I done a translation on the data in my model and not the linear coefficients.

Multiplying my coefficient matrix with the transpose of the rotation matrix in my model gave me unscaled coefficients which then had to be divided by the standard deviation to rescale.

With the intercept formula I was able to get predictions from the PCA model in terms of the original variables

[,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]

[1,] 713.6803 1195.707 506.4008 1744.815 1004.322 901.3083 817.7618 1158.016 862.66 906.1942 1309.847

[,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22]

[1,] 831.7397 668.7175 653.8079 663.3242 933.786 467.7924 1097.833 975.2212 1238.845 805.7895 769.6724

[,23] [,24] [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33]

[1,] 768.1369 928.9523 604.2355 1845.757 480.427 1015.084 1463.794 801.6455 687.8542 969.6941 722.6822

[,34] [,35] [,36] [,37] [,38] [,39] [,40] [,41] [,42] [,43] [,44]

[1,] 841.7013 914.9564 977.8353 1211.689 604.2928 627.6148 1069.894 841.4929 272.2545 1043.452 1126.343

[,45] [,46] [,47]

[1,] 425.4541 927.1627 1139.354

>

And compare this to actual values

[1] 791 1635 578 1969 1234 682 963 1555 856 705 1674 849 511 664 798 946 539 929 750 1225 742

[22] 439 1216 968 523 1993 342 1216 1043 696 373 754 1072 923 653 1272 831 566 826 1151 880 542

[43] 823 1030 455 508 849

I squared the correlation between the two data sets to get an R squared value of 64.5% which matches what the summary of the model provided

Residual standard error: 244 on 41 degrees of freedom

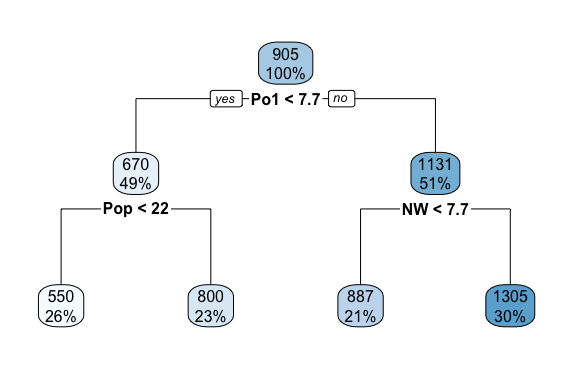
Multiple R-squared: 0.6452, Adjusted R-squared: 0.6019

F-statistic: 14.91 on 5 and 41 DF, p-value: 2.446e-08

The PCA model was not as good as the linear model from question 8.2 .

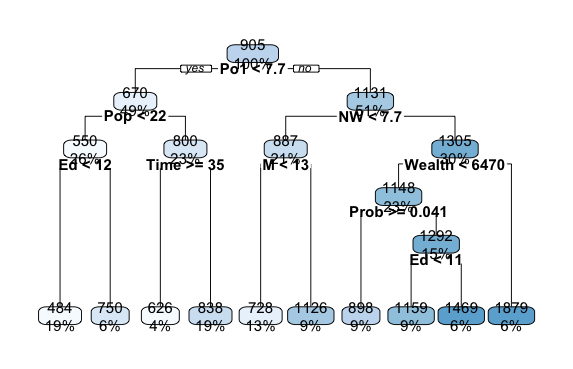
**Questions 10.1 Using the same crime data set uscrime.txt as in Questions 8.2 and 9.1, find the best model you can using (a) a regression tree model, and (b) a random forest model. In R, you can use the tree package or the rpart package, and the randomForest package. For each model, describe one or two qualitative takeaways you get from analyzing the results (i.e., don’t just stop when you have a good model, but interpret it too).**

Given the small size of the data set I ran the rpart on the full data set as the instructor lecture suggested that splitting such a small data set would effect results and not give us an ideal test data sample.



This data set had a low R squared value. (56%)

I forced the rpart function to generate more leaves and got a much higher R squared (91%) value but given the limited number of variables there is certainly over fitting.



Some qualitative observations on the data are the importance of the top variables in explaining variance in the data. Po1 is an important variable followed by POP and NW in both trees.

The random tree model also gave a low R square value

I was able to look at the importance data in the model to make the same type of observation as above for insight into which predictors were more important then others. In this data the larger numbers for Po1 and Po2 indicate they are realtively more important variables then their peers.

M 214855.11

So 31689.83

Ed 242875.99

Po1 1215177.08

Po2 1183441.38

LF 241700.69

M.F 252304.84

Pop 321128.42

NW 520656.19

U1 125870.39

U2 195710.16

Wealth 610317.98

Ineq 199336.36

Prob 812593.57

Time 217823.49

**Question 10.2 Describe a situation or problem from your job, everyday life, current events, etc., for which a logistic regression model would be appropriate. List some (up to 5) predictors that you might use.**

At our company we offer a subscription based service. We can give a free subscription to a use we think is a good prospect for becoming a long time subscriber. Because we pay costs associated with the subscription we want to give it to a profile that is likely to subscribe so we would want to use the predictors to predict either likely to subscribe or not likely. Predictors could include income, geography, source and age.

**1. Using the GermanCredit data set germancredit.txt from http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german / (description at http://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29 ), use logistic regression to find a good predictive model for whether credit applicants are good credit risks or not. Show your model (factors used and their coefficients), the software output, and the quality of fit. You can use the glm function in R. To get a logistic regression (logit) model on data where the response is either zero or one, use family=binomial(link=”logit”) in your glm function call.**

**2. Because the model gives a result between 0 and 1, it requires setting a threshold probability to separate between “good” and “bad” answers. In this data set, they estimate that incorrectly identifying a bad customer as good, is 5 times worse than incorrectly classifying a good customer as bad. Determine a good threshold probability based on your model**

For the final question I followed the instructions read in the german credit file converted the response data from 1 for no and 2 for yes to 0 for no and 1 for yes since we need a binary response for linear regression. A summary of the model and the coefficients are pasted below. The asterisk indicate which variables are important in the model relative to other predictors.

Call:

glm(formula = V21 ~ ., family = binomial(link = "logit"), data = german\_data)

Deviance Residuals:

Min 1Q Median 3Q Max

-2.3410 -0.6994 -0.3752 0.7095 2.6116

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 4.005e-01 1.084e+00 0.369 0.711869

V1A12 -3.749e-01 2.179e-01 -1.720 0.085400 .

V1A13 -9.657e-01 3.692e-01 -2.616 0.008905 \*\*

V1A14 -1.712e+00 2.322e-01 -7.373 1.66e-13 \*\*\*

V2 2.786e-02 9.296e-03 2.997 0.002724 \*\*

V3A31 1.434e-01 5.489e-01 0.261 0.793921

V3A32 -5.861e-01 4.305e-01 -1.362 0.173348

V3A33 -8.532e-01 4.717e-01 -1.809 0.070470 .

V3A34 -1.436e+00 4.399e-01 -3.264 0.001099 \*\*

V4A41 -1.666e+00 3.743e-01 -4.452 8.51e-06 \*\*\*

V4A410 -1.489e+00 7.764e-01 -1.918 0.055163 .

V4A42 -7.916e-01 2.610e-01 -3.033 0.002421 \*\*

V4A43 -8.916e-01 2.471e-01 -3.609 0.000308 \*\*\*

V4A44 -5.228e-01 7.623e-01 -0.686 0.492831

V4A45 -2.164e-01 5.500e-01 -0.393 0.694000

V4A46 3.628e-02 3.965e-01 0.092 0.927082

V4A48 -2.059e+00 1.212e+00 -1.699 0.089297 .

V4A49 -7.401e-01 3.339e-01 -2.216 0.026668 \*

V5 1.283e-04 4.444e-05 2.887 0.003894 \*\*

V6A62 -3.577e-01 2.861e-01 -1.250 0.211130

V6A63 -3.761e-01 4.011e-01 -0.938 0.348476

V6A64 -1.339e+00 5.249e-01 -2.551 0.010729 \*

V6A65 -9.467e-01 2.625e-01 -3.607 0.000310 \*\*\*

V7A72 -6.691e-02 4.270e-01 -0.157 0.875475

V7A73 -1.828e-01 4.105e-01 -0.445 0.656049

V7A74 -8.310e-01 4.455e-01 -1.866 0.062110 .

V7A75 -2.766e-01 4.134e-01 -0.669 0.503410

V8 3.301e-01 8.828e-02 3.739 0.000185 \*\*\*

V9A92 -2.755e-01 3.865e-01 -0.713 0.476040

V9A93 -8.161e-01 3.799e-01 -2.148 0.031718 \*

V9A94 -3.671e-01 4.537e-01 -0.809 0.418448

V10A102 4.360e-01 4.101e-01 1.063 0.287700

V10A103 -9.786e-01 4.243e-01 -2.307 0.021072 \*

V11 4.776e-03 8.641e-02 0.055 0.955920

V12A122 2.814e-01 2.534e-01 1.111 0.266630

V12A123 1.945e-01 2.360e-01 0.824 0.409743

V12A124 7.304e-01 4.245e-01 1.721 0.085308 .

V13 -1.454e-02 9.222e-03 -1.576 0.114982

V14A142 -1.232e-01 4.119e-01 -0.299 0.764878

V14A143 -6.463e-01 2.391e-01 -2.703 0.006871 \*\*

V15A152 -4.436e-01 2.347e-01 -1.890 0.058715 .

V15A153 -6.839e-01 4.770e-01 -1.434 0.151657

V16 2.721e-01 1.895e-01 1.436 0.151109

V17A172 5.361e-01 6.796e-01 0.789 0.430160

V17A173 5.547e-01 6.549e-01 0.847 0.397015

V17A174 4.795e-01 6.623e-01 0.724 0.469086

V18 2.647e-01 2.492e-01 1.062 0.288249

V19A192 -3.000e-01 2.013e-01 -1.491 0.136060

V20A202 -1.392e+00 6.258e-01 -2.225 0.026095 \*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1221.73 on 999 degrees of freedom

Residual deviance: 895.82 on 951 degrees of freedom

AIC: 993.82

Number of Fisher Scoring iterations: 5

Since the question asks us to find a good threshold (which was helpful since it did not require the best) and cost of identifying a bad customer as good as 5 times worse then classifying a good customer as bad.

The initial threshold yield a cost value of 5 \* FP + FN of

> c\_mat

0 1

0 626 74

1 140 160

> cost

[1] 774

By increasing the threshold in increments of I was able to find a good threshold value at 80 percent with a reduced overall cost

> c\_mat

0 1

0 432 268

1 42 258

> cost

[1] 478