PDA week 5 and 6

David Carbonello

10/6/2020

Question 1: Linear Regression

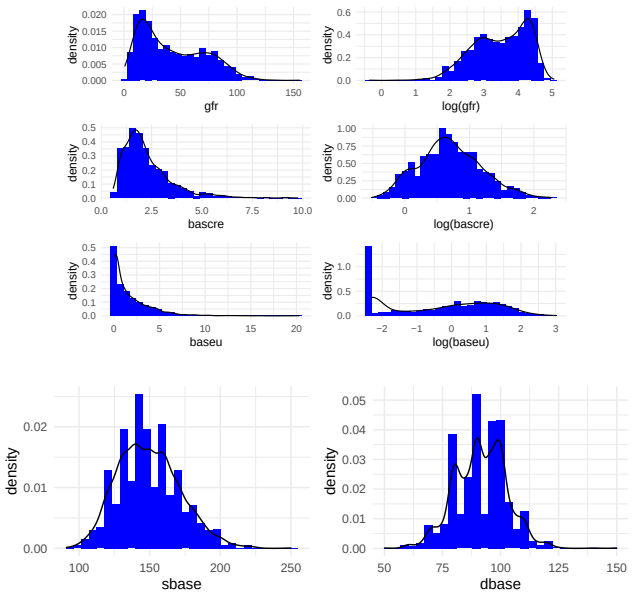
Subsetting dataframe to only include variables we are concerned with and observing structure of data.

```
## 'data.frame':    1860 obs. of  8 variables:
##  $ gfr   : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ bascre: num  4.4 2.5 2.1 3.3 2.8 4.9 2.9 2 1.9 1.7 ...
##  $ sbase : num  190 180 150 150 165 160 190 190 170 170 ...
##  $ dbase : num  100 115 110 110 110 110 120 120 110 110 ...
##  $ baseu : num  2.6 0.1 0.6 0.1 0.1 3 0.875 2.25 0.1 0.3 ...
##  $ AGE   : int  66 53 31 42 64 30 50 61 51 58 ...
##  $ SEX   : int  0 0 1 0 0 0 0 1 0 1 ...
##  $ black : int  0 0 0 0 0 0 0 0 0 0 ...
```

Data Exploration

Variable Transformations

Looking at continuous variable's distributions and their log transformations. It appears, that gfr,basecre, and baseu are more normally distributed under the log transform. Both dbase, and sbase appear to be normally distributed as is, so no log transformation is shown



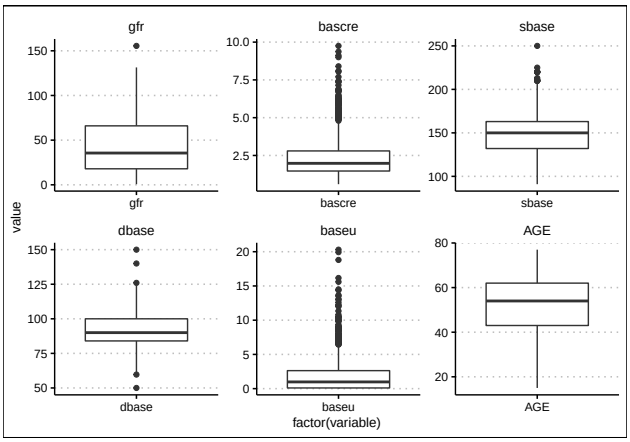
Summary Statistics

Looking at summary statistics of each variable in dataset

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
gfr	1	1249	42.591194	28.3772687	35.560000	40.380155	31.6979879	0.7000000	155.50	154.800000	0.5842806	-0.6721926	0.8029516
bascre	2	1860	2.269101	1.2227405	1.979638	2.090884	0.9197487	0.5995475	9.75	9.150453	1.7731100	4.5692883	0.0283516
sbase	3	1860	149.417598	21.9986043	150.000000	148.546707	22.2390000	91.0000000	250.00	159.000000	0.3982365	0.0746278	0.5100804
dbase	4	1860	91.484788	11.1192895	90.000000	91.478226	13.3434000	50.0000000	150.00	100.000000	0.0356929	0.4654067	0.2578224
baseu	5	1860	1.801519	2.3204381	0.982000	1.352864	1.3076532	0.1000000	20.30	20.200000	2.5596100	10.3056429	0.0538039
AGE	6	1860	51.945699	12.9369621	54.000000	52.928091	13.3434000	15.0000000	77.00	62.000000	-0.5908363	-0.3973371	0.2999686
SEX*	7	1860	1.653226	0.4760711	2.000000	1.691532	0.0000000	1.0000000	2.00	1.000000	-0.6433637	-1.5869355	0.0110386
black*	8	1860	1.061290	0.2399266	1.000000	1.000000	0.0000000	1.0000000	2.00	1.000000	3.6550665	11.3656222	0.0055632

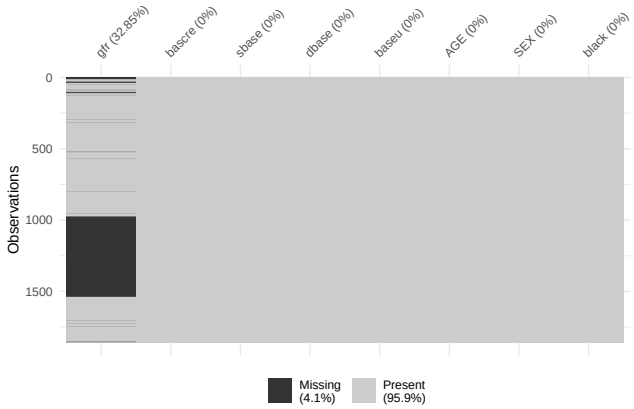
Observing Outliers

Creating boxplots of all continuous predictors. Outcome variable contains 1 outlier. Overall, data looks pretty good, but perhaps some validation of basecre or baseu should be performed. For now, no outliers will be removed for the model building process.

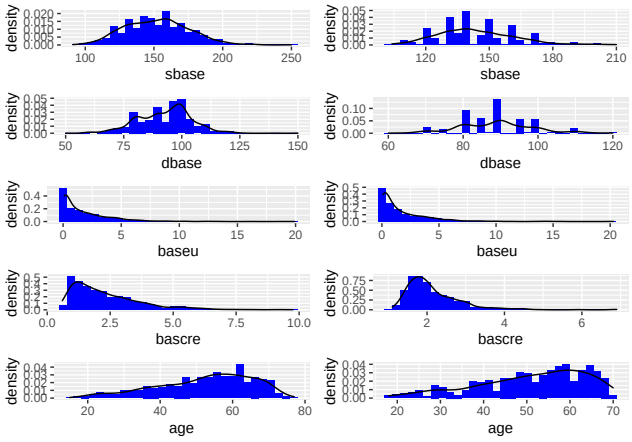


Missing Data

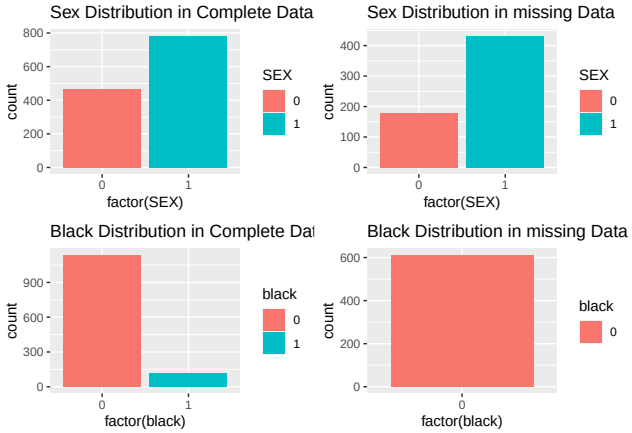
Looking At Missing Data. All missing data occurs in the outcome variable gfr.



Comparing Distributions of variables where GFR data is present (left) and where GFR data is missing (right). The graphs below suggest that there is nothing unusual going on in the continuous variables where there is missing data.

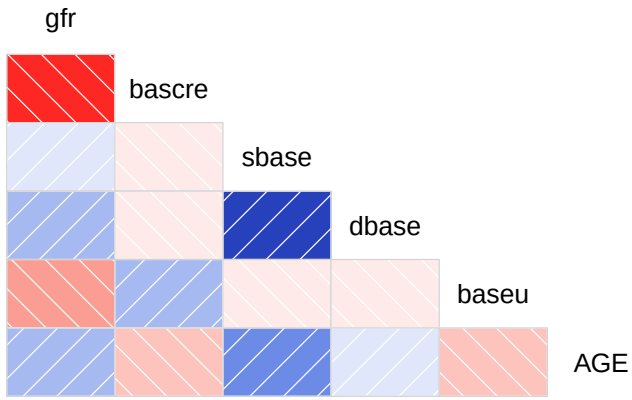


Comparing categorical variables where GFR data is present (left) and not missing (right).



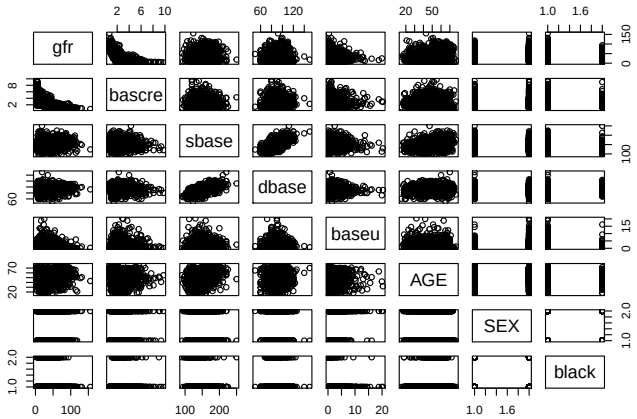
Collinearity

Checking the collinearity of variables. The corrgram suggests that sbase and dbase are highly correlated. When building the regression model likely on one of these variables will be included.

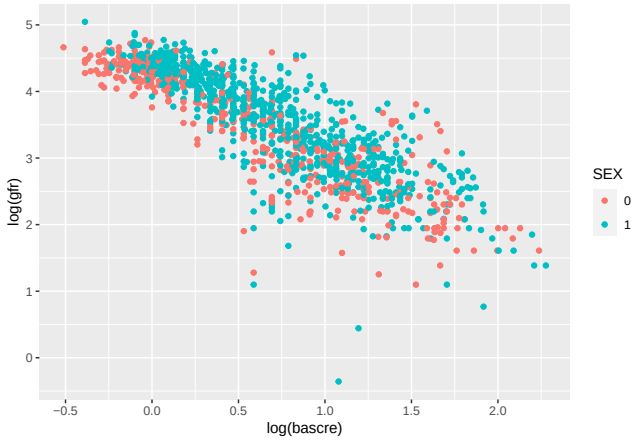


Variable relationships

Observing Pairwise relationships. The plot below suggests that both baseu and basecre may exhbit some linear association with the outcome of gfr. Also, sbase and dbase exhbit strong positive correlation, as suggested in the corrgram as well.



The plot below shows the relation ship beween the log transforms of gfr and basecre, grouped by SEX. It appears there is a strong linear relationship between these variables, and sex may have an effect on this relationship, where males are have slightly larger gfr than females.



Fitting Models

```
baseseg_data$log_basecre<-log(baseseg_data$bascre)
baseseg_data$log_baseu<-log(baseseg_data$baseu)
# Full model
full_model<-lm(data = baseseg_data, log(gfr)~bascre+sbase+dbase+baseu+AGE+SEX+black)

fit1<-lm(data = baseseg_data, log(gfr)~bascre)
fit2<-lm(data = baseseg_data, log(gfr)~bascre+sbase+baseu)
fit3<-lm(data = baseseg_data, log(gfr)~bascre+dbase+baseu)
fit4<-lm(data = baseseg_data, log(gfr)~bascre+dbase+baseu+SEX)
fit5<-lm(data = baseseg_data, log(gfr)~log(bascre)+dbase+baseu+SEX)
```

```r
fit6<-lm(data = baseseg_data, log(gfr)~log(bascre)+dbase+baseu+SEX+AGE)
fit7<-lm(data = baseseg_data, log(gfr)~log(bascre)+dbase+log(baseu)+SEX+AGE)
fit8<-lm(data = baseseg_data, log(gfr)~log_basecre+log_baseu+SEX+AGE)
```

Results

```
##   Model AIC_Values BIC_Values  Rsquared
## 1  fit1   1767.801   1783.191 0.6266459
## 2  fit2   1740.326   1765.976 0.6359369
## 3  fit3   1739.941   1765.592 0.6360489
## 4  fit4   1702.867   1733.648 0.6472587
## 5  fit5   1356.229   1387.010 0.7327452
## 6  fit6   1355.018   1390.929 0.7334313
## 7  fit7   1342.330   1378.241 0.7361256
## 8  fit8   1340.868   1371.649 0.7361256
```

The table above suggests that fit8 has the best fit based on the smallest values of both AIC and BIC, as well as the highest R squared value of .736, suggesting that 73.6% of the variation in log(gfr) is explained by the model.

The coefficients of fit 7 are shown below:

```
##                 Estimate   Std. Error    t value      Pr(>|t|)
## (Intercept)  4.270391972 0.0574746682  74.300420  0.000000e+00
## log_basecre -1.194620209 0.0241922977 -49.380188 1.942581e-295
## log_baseu   -0.044813174 0.0092724145  -4.832956  1.512833e-06
## SEX1         0.243772901 0.0245491018   9.930013  2.049284e-22
## AGE         -0.002630297 0.0009649349  -2.725880  6.503038e-03
```

The fit8 linear regression model contains all significant predictors based on the small pvalues and cutoff value of .05.

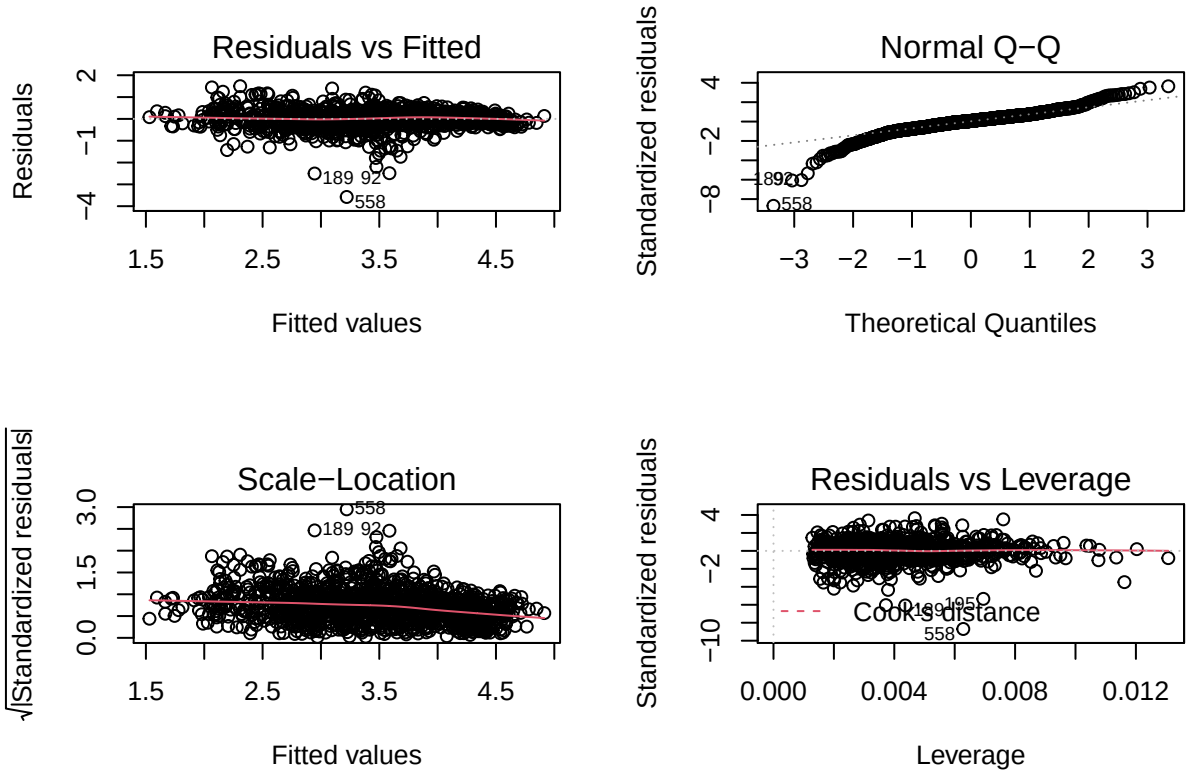
The Age coefficient suggests that a 1 year increase in age is associated with a .00263 decrease in log(gfr) holding other variables constant. Or, a 1 year increase is associated with associated with a .99 times decrease in gfr holding other variables constant.

The Sex coefficient suggests that moving from the female to male group is associated with a .243 increase in log(gfr) holding other variables constant.

The log(basecre) variable suggests that if basecre doubles, then gfr will decrease by 56% exp(-1.194620209*log(2))=.4369

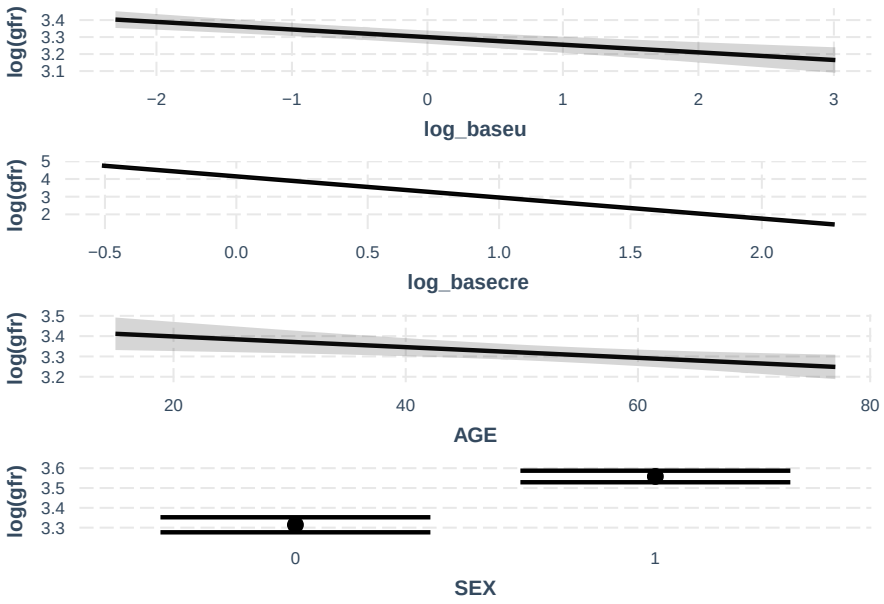
The log(baseu) variable suggests that if baseu doubles, then gfr will decrease by 3.05% exp(-0.044813174*log(2))=.9694153

Diagnostic Plots



The model diagnostics suggest that the assumptions of the linear model are fairly met. The residuals vs Fitted plot does not exhibit any trend and residuals appear to be equally variable throughout suggesting a constant variance. The assumption of normality of the residuals leaves room for improvement based on higher and lower values. The scale-location plot is fairly horizontal suggesting good homoscedasticity, meaning the noise or error is equally distributed for all values of the predictors in the model. Lastly, the residuals vs leverage does not indicate any very influential points affecting the models since no points lie outside of cook's distance. Overall, the diagnostic plots suggest the assumptions of linear regression are met.

Effect Plots



The effects plots show the partial slope for each predictor. The effects of each variable in the model are linear and echo what is described in the summary output for fit8. For example, a 1 unit increase in log(basecre) is associated with a 1.1946202 decrease in the log(gfr) holding other variables constant. Additionally, the plots show the negative linear relationship that both log_basecre and age exhibit with the outcome log(gfr). The effects plot for sex show that switching from the female group to male group is associated with a greater log(gfr).

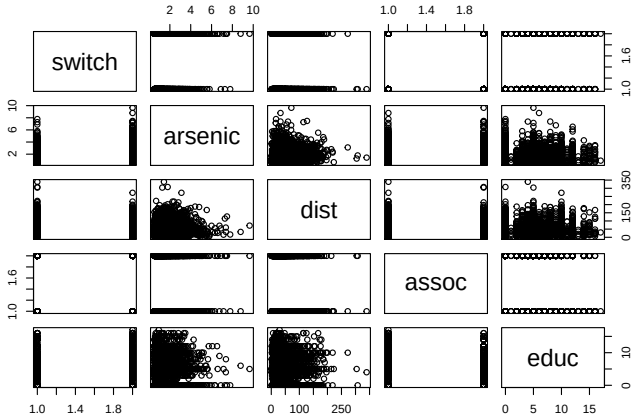
Logistic Regression

Creating Training and Test Set

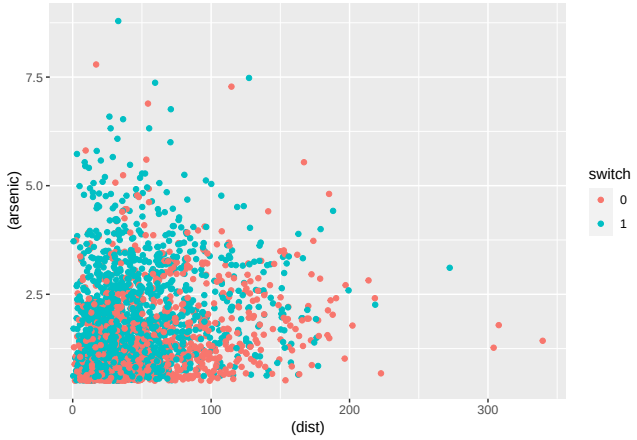
```
train<-wells_data[(1:2520),]
test<-wells_data[-(1:2520),]
```

Obseving relationships of variables

The plot below suggests there is a slight positive linear association between the arsenic and dist variables.



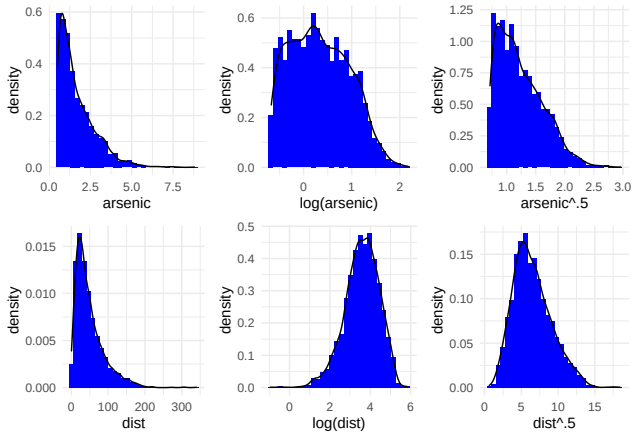
The plot below suggests that there is a higher concentration of switchers when the distance from a clean well is low, and there are high arsenic levels. Additionally, people are less likely to switch when the are far from a clean water source, and there are lower levels of arsenic. This relationship makes sense from a practical point of view.



Construct a good logistic regression model predicting the decision to switch wells as a function of the 4 predictors (arsenic, distance, association and education) on the training data. Consider potential transformations of continuous variables and possible interactions.

Variable Transformations

Looking at potential transformations of continuous variables



Fitting models

```
fit1<-glm(data = train,switch~arsenic,family="binomial")

fit2<-glm(data = train,switch~arsenic+dist+assoc+educ,family="binomial")

fit3<-glm(data = train,switch~dist+arsenic+dist*arsenic+educ+educ*dist,family="binomial")

fit4<-glm(data = train,switch~log(dist)+log(arsenic)+log(dist)*log(arsenic)+educ+educ*log(dist),family=

fit5<-glm(data = train,switch~(dist)+(arsenic)+educ+educ*(dist),family="binomial")
```

The table below shows evaluation of the different models fit. Prediction accuracy and missclassification rate were calculated for the models performance on the test set, and also AIC and BIC were caluclated.

```
##   Model Accuracy MissClass AIC_Values BIC_Values
## 1  fit1    0.486     0.514   3322.495   3334.159
## 2  fit2    0.522     0.478   3236.967   3266.127
## 3  fit3    0.526     0.474   3231.000   3265.992
## 4  fit4    0.576     0.424   3234.524   3269.517
## 5  fit5    0.526     0.474   3229.734   3258.894
```

Based on prediction the highest Accuracy on the test set, fit 4 appears to perform the best. This model also has fairly low AIC compared to the other models, although it does not have the lowsest AIC or BIC. Below is the summary for the model.

```
##                            Estimate Std. Error   z value     Pr(>|z|)
## (Intercept)              1.82381414 0.30414406  5.996547 2.015573e-09
## log(dist)               -0.51479584 0.08426354 -6.109354 1.000350e-09
## log(arsenic)             1.42416093 0.33334182  4.272374 1.934028e-05
## educ                    -0.10844694 0.04690593 -2.312009 2.077718e-02
## log(dist):log(arsenic) -0.15987694 0.08889175 -1.798558 7.208868e-02
## log(dist):educ           0.04166395 0.01279844  3.255393 1.132354e-03
```

Using the pvalue cutoff of.05 for significance, all of the predictors other than the interaction between log(dist) and log(arsenic) are significant. The interaction between log(dist) and log(arsenic) is very close to this cutoff point with a p-value of.07 suggesting mild evidence.

The specifics of the significant variables are interpreted below:

The intercept suggests that the odds of switching are 6.195418 when all other predictors are 0. exp(1.82381)=6.195418

The coefficient of the log(dist) variable suggests that a 1 unit increase in log(dist) is associated with an .5976226 times decrease in the odds of switching holding other variables constant. (exp(-0.51479584)=.5976226)

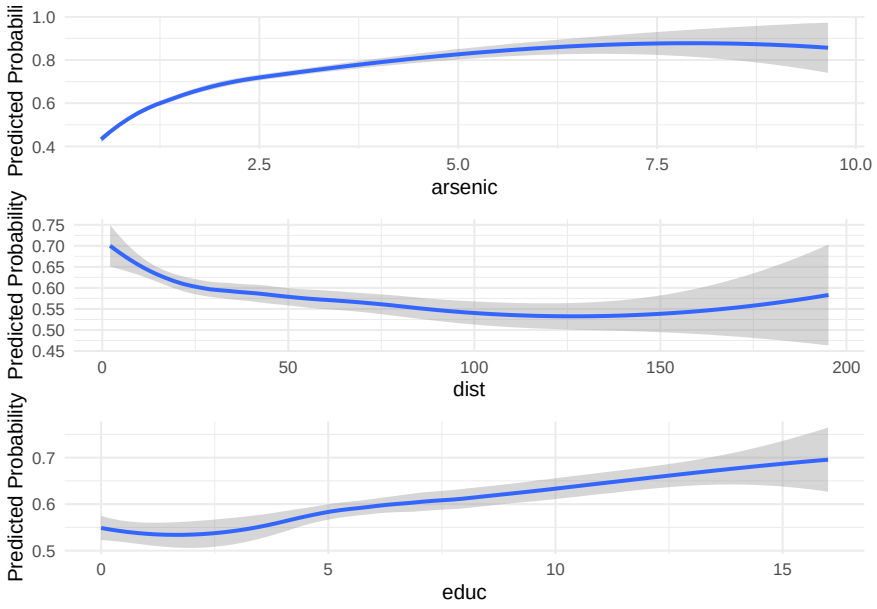
The coefficient of the log(arsenic) variable suggests that a 1 unit increase in log(arsenic) is associated with a 4.154367 times increase in the odds of switching holding other variables constant.(exp(1.42416)=4.154367)

The coefficient of the log(arsenic) variable suggests that a 1 unit increase in log(arsenic) is associated with a 4.154367 times increase in the odds of switching holding other variables constant. (exp(1.42416)=4.154367)

The coefficient of the educ variable suggests that a 1 unit increase in educ is associated with a 0.8972238 times decrease in the odds of switching holding other variables constant. (exp(-0.10845)=0.8972238)

The interaction term log(dist):educ suggests that the difference in log odds corresponding to a 1 unit increase in log(dist) is 0.04166, where the two groups being compared differ by 1 year of education. (holding other variables constant.)

Compute and graph the predicted probabilities stratifying by the predictors. You could do this using graphs such as in the papers we discussed in class or by using contour plots which would allow you to graph two continuous predictors on the same plot. You can array different lines and plots to try to put this all on one sheet or you can spread across different plots. See what works best.



The plots about show the predicted probabilities using the final model when evaluating on the test set, stratified for each predictor. The first plot suggests that an arsenic level of 2.5 corresponds to about a 65% chance of switching, and an arsenic level of 7.5 corresponds to a 90% chance of switching.

Confusion Matrix

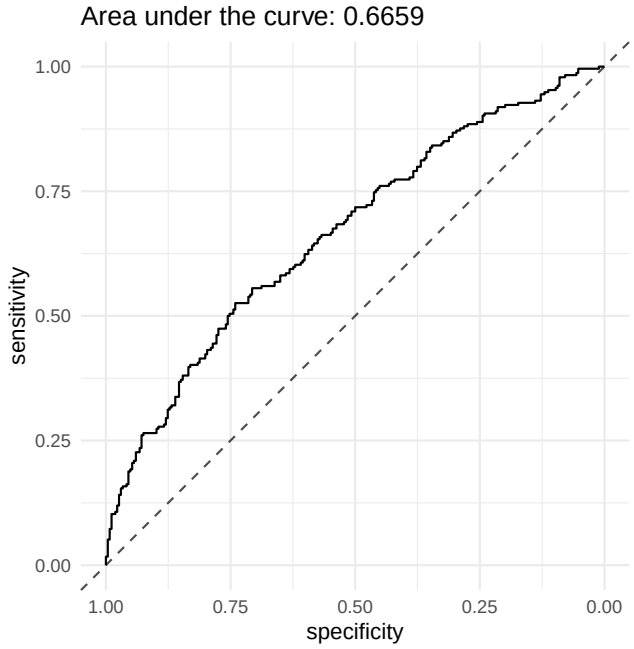
Compute the confusion matrix on the test data using p = 0.5 as a cutoff and discuss what this tells you about the predictive model you have constructed (e.g. sensitivity, specificity, error rate, etc.)

```
##   Accuracy MissClass Sensitivity Specificity
## 1    0.576     0.424   0.8418803   0.3421053
```

The model performs only slightly better than 50% at identifying the correct class based on its accuracy of .576. The sensitivty of .841 indicates the model has a high true positive rate, meaning the model correctly identifies individuals that switch 84.1% of the time. The Specificity of .3421 suggests the model has a low True negative rate, where the model identifies individuals that did not switch correctly on 34.21% of the time. Overall, the model leaves much room for improvement.

ROC and AUC

Construct an ROC plot and compute the area under the ROC curve.




```
##   cutpoints sensitivties specificities
## 1 0.5007360    0.8418803     0.3458647
## 2 0.5014705    0.8376068     0.3458647
## 3 0.5018143    0.8376068     0.3496241
## 4 0.5022585    0.8333333     0.3496241
## 5 0.5025181    0.8290598     0.3496241
## 6 0.5029255    0.8290598     0.3533835
```

What does this curve tell you about choice of threshold that balances sensitivity with specificity (i.e., how would you balance risk of switching and not switching?)

The threshold of .5 corresponds to a sensitivity of approximately .84 and sensitivity of .35. To have a more balanced sensitivity and specificity, a threshold of .58 would yield a sensitivity of .60 and specificity of .60. Determining whether or not it is appropriate to switch the threshold is dependent on which whether the true positive rate or true negative rate is more important in the context of the question.

```
## [1] "Percentage of Correct Predictions: 0.608"
```

```
##
## glm.pred   0   1
##        0 162  92
##        1 104 142
```

By using a threshold that balances sensitivity and specificity, the accuracy of the model improves to .608 from the previous .576.

```
## [1] "Percentage of Correct Predictions: 0.626"
```

```
##
## glm.pred   0   1
##        0 226 147
##        1  40  87
```

Using a threshold of .68 improves the prediction accuracy to .626.

Code Appendix

```
knitr::opts_chunk$set(echo = FALSE,warning = FALSE, message = FALSE)
setwd('/Users/davidcarbonello/R/PDA/')
baseseg_data<-read.csv('baseseg.csv')
library(dplyr)
library(ggplot2)
library(ggthemes)
library(psych)
library(kableExtra)
library(reshape)
library(naniar)
library(corrgram)
library(gridExtra)
library(InformationValue)
library(ROSE)
library(pROC)
library(jtools)
# Subsetting data to keep variables we are interested in
baseseg_data<-baseseg_data%>%select(gfr,bascre,sbase,dbase,baseu,AGE,SEX,black)
# Observing data
#head(baseseg_data)
str(baseseg_data)
#Converting Categorical Variables
baseseg_data$SEX<-as.factor(baseseg_data$SEX)
baseseg_data$black<-as.factor(baseseg_data$black)
# Consider Log transformations of variables

# Distribution of Outcome
p1<-ggplot(data=baseseg_data, aes(baseseg_data$gfr)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("gfr")+theme_minimal()

p2<-ggplot(data=baseseg_data, aes(log(baseseg_data$gfr))) +
```

```
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("log(gfr)")+theme_minimal()

p3<-ggplot(data=baseseg_data, aes(baseseg_data$bascre)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("bascre")+theme_minimal()

p4<-ggplot(data=baseseg_data, aes(log(baseseg_data$bascre))) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("log(bascre)")+theme_minimal()

p5<-ggplot(data=baseseg_data, aes(baseseg_data$baseu)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("baseu")+theme_minimal()

p6<-ggplot(data=baseseg_data, aes(log(baseseg_data$baseu))) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("log(baseu)")+theme_minimal()

x<-grid.arrange(p1,p2,p3,p4,p5,p6)

plot1<-ggplot(data=baseseg_data, aes(baseseg_data$sbase)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("sbase")+theme_minimal()

plot2<-ggplot(data=baseseg_data, aes(baseseg_data$dbase)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("dbase")+theme_minimal()

y<-grid.arrange(plot1,plot2,ncol=2,widths=c(2.3, 2.3),heights=c(1.6,1.6))

SummaryStatistics<-describe(baseseg_data)  # take out log,sqrt,cube and make fit on page
kable(SummaryStatistics)%>%
kable_classic(full_width = F, html_font = "Cambria")%>%
kable_styling(latex_options="scale_down")
## Observing Distributions/outliers of predictors   # potentially remove outlier in gfr?
Continuous_variables<-baseseg_data%>%select(gfr,bascre,sbase,dbase,baseu,AGE)
meltData <- melt(Continuous_variables)
p <- ggplot(meltData, aes(factor(variable), value))
p + geom_boxplot() + facet_wrap(~variable, scale="free")+ theme_clean()

vis_miss(baseseg_data)

MissingGFRData<-baseseg_data%>%filter(is.na(gfr)==TRUE)
MissingGFRData_Continuous<-MissingGFRData%>%select(bascre,sbase,dbase,baseu,AGE)

# Non missing
Complete_baseseg_data<-baseseg_data%>%filter(!is.na(gfr))

p1<-ggplot(data=Complete_baseseg_data, aes(sbase)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("sbase")
```

```
p2<-ggplot(data=MissingGFRData_Continuous, aes(sbase)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("sbase")

p3<-ggplot(data=Complete_baseseg_data, aes(dbase)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("dbase")

p4<-ggplot(data=MissingGFRData_Continuous, aes(dbase)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("dbase")

p5<-ggplot(data=Complete_baseseg_data, aes(baseu)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("baseu")

p6<-ggplot(data=MissingGFRData_Continuous, aes(baseu)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("baseu")

p7<-ggplot(data=Complete_baseseg_data, aes(bascre)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("bascre")

p8<-ggplot(data=MissingGFRData_Continuous, aes(bascre)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("bascre")

p9<-ggplot(data=Complete_baseseg_data, aes(AGE)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("age")

p10<-ggplot(data=MissingGFRData_Continuous, aes(AGE)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("age")

grid.arrange(p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,ncol=2)

p1<-ggplot(Complete_baseseg_data, aes(factor(SEX),fill=SEX)) + geom_bar()+ggtitle("Sex Distribution in C
p2<-ggplot(MissingGFRData, aes(factor(SEX),fill=SEX)) + geom_bar() +ggtitle("Sex Distribution in missing

p3<-ggplot(Complete_baseseg_data, aes(factor(black),fill=black)) + geom_bar() +ggtitle("Black Distributi
p4<-ggplot(MissingGFRData, aes(factor(black),fill=black)) + geom_bar()+ ggtitle("Black Distribution in m
grid.arrange(p1,p2,p3,p4,ncol=2)
# removing transformed variables from DF for plot
baseseg_data_correlations<-baseseg_data%>%select(gfr,bascre,sbase,dbase,baseu,AGE)
corrgram(baseseg_data_correlations, lower.panel=panel.shade,upper.panel=NULL,order=FALSE)
plot(baseseg_data)
ggplot(data=baseseg_data,aes(log(bascre),log(gfr),color=SEX))+geom_point()
baseseg_data$log_basecre<-log(baseseg_data$bascre)
baseseg_data$log_baseu<-log(baseseg_data$baseu)
# Full model
full_model<-lm(data = baseseg_data, log(gfr)~bascre+sbase+dbase+baseu+AGE+SEX+black)
```

```r
fit1<-lm(data = baseseg_data, log(gfr)~bascre)
fit2<-lm(data = baseseg_data, log(gfr)~bascre+sbase+baseu)
fit3<-lm(data = baseseg_data, log(gfr)~bascre+dbase+baseu)
fit4<-lm(data = baseseg_data, log(gfr)~bascre+dbase+baseu+SEX)
fit5<-lm(data = baseseg_data, log(gfr)~log(bascre)+dbase+baseu+SEX)
fit6<-lm(data = baseseg_data, log(gfr)~log(bascre)+dbase+baseu+SEX+AGE)
fit7<-lm(data = baseseg_data, log(gfr)~log(bascre)+dbase+log(baseu)+SEX+AGE)
fit8<-lm(data = baseseg_data, log(gfr)~log_basecre+log_baseu+SEX+AGE)

# Writing Function To get values from LM
lm_values <- function (modelobject) {
    if (class(modelobject) != "lm") stop("Not an object of class 'lm' ")
    R_Squared<-summary(modelobject)$r.squared
    AIC<-AIC(modelobject)
    BIC<-BIC(modelobject)
    return(list("R_squared"=R_Squared,"AIC"=AIC,"BIC"=BIC))
}

Results1<-lm_values(fit1)
Results2<-lm_values(fit2)
Results3<-lm_values(fit3)
Results4<-lm_values(fit4)
Results5<-lm_values(fit5)
Results6<-lm_values(fit6)
Results7<-lm_values(fit7)
Results8<-lm_values(fit8)

Model<-c('fit1','fit2','fit3','fit4','fit5','fit6','fit7','fit8')
AIC_Values<-c(Results1$AIC,Results2$AIC,Results3$AIC,Results4$AIC,Results5$AIC,Results6$AIC,Results7$AI
BIC_Values<-c(Results1$BIC,Results2$BIC,Results3$BIC,Results4$BIC,Results5$BIC,Results6$BIC,Results7$BI
Rsquared<-c(Results1$R_squared,Results2$R_squared,Results3$R_squared,Results4$R_squared,Results5$R_squa

df<-data.frame(Model,AIC_Values,BIC_Values,Rsquared)
df
# kable(df,col.names = c("Model","AIC","BIC","R Squared"))%>%
# kable_classic(full_width = F, html_font = "Cambria")%>%
# kable_styling(latex_options="scale_down")
summary(fit8)$coef


par(mfrow=c(2,2))
plot(fit8)

p1<-effect_plot(fit8, pred = log_baseu, interval = TRUE)
p2<-effect_plot(fit8, pred = log_basecre, interval = TRUE)
p3<-effect_plot(fit8, pred = AGE, interval = TRUE)
p4<-effect_plot(fit8, pred = SEX, interval = TRUE)
grid.arrange(p1,p2,p3,p4,ncol=1)
setwd('/Users/davidcarbonello/R/PDA/')
wells_data<-read.delim("wells.txt", header = TRUE,sep = " ")

# Changing to factor variables
#wells_data$educ<-as.factor(wells_data$educ)
```

```r
wells_data$assoc<-as.factor(wells_data$assoc)
wells_data$switch<-as.factor(wells_data$switch)
train<-wells_data[(1:2520),]
test<-wells_data[-(1:2520),]
plot(wells_data)
ggplot(data = train,aes(x=(dist),y=(arsenic),color=switch))+geom_point()
# Distribution of Outcome
p1<-ggplot(data=train, aes(arsenic)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("arsenic")+theme_minimal()

p2<-ggplot(data=train, aes(log(arsenic))) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("log(arsenic)")+theme_minimal()

p3<-ggplot(data=train, aes((arsenic^.5))) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("arsenic^.5")+theme_minimal()


p4<-ggplot(data=train, aes(dist)) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("dist")+theme_minimal()

p5<-ggplot(data=train, aes(log(dist))) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("log(dist)")+theme_minimal()

p6<-ggplot(data=train, aes((dist^.5))) +
  geom_histogram(aes(y =..density..), fill = "blue") +
  geom_density() +xlab("dist^.5")+theme_minimal()

grid.arrange(p1,p2,p3,p4,p5,p6,nrow=2)


fit1<-glm(data = train,switch~arsenic,family="binomial")

fit2<-glm(data = train,switch~arsenic+dist+assoc+educ,family="binomial")

fit3<-glm(data = train,switch~dist+arsenic+dist*arsenic+educ+educ*dist,family="binomial")

fit4<-glm(data = train,switch~log(dist)+log(arsenic)+log(dist)*log(arsenic)+educ+educ*log(dist),family=

fit5<-glm(data = train,switch~(dist)+(arsenic)+educ+educ*(dist),family="binomial")

# full Logistic Regression model
# fit1<-glm(data = train,switch~arsenic,family="binomial")
# fit2<-glm(data = train,switch~arsenic+dist,family="binomial")
# fit3<-glm(data = train,switch~arsenic+dist+educ,family="binomial")
# fit4<-glm(data = train,switch~arsenic+dist+assoc+educ,family="binomial")
# fit5<-glm(data = train,switch~log(arsenic)+dist+assoc+educ,family="binomial")
# fit6<-glm(data = train,switch~log(arsenic)+log(dist)+assoc+educ,family="binomial")
# fit7<-glm(data = train,switch~log(arsenic)+log(dist)+educ,family="binomial")
```

```r
prediction_accuracy<-function(modelobject){
  if (class(modelobject) != "glm") stop("Not an object of class 'glm' ")
  probs=predict(modelobject,test,type="response")
  Accuracy<-1-misClassError(test$switch,probs,threshold = .5)
  MissClass<-misClassError(test$switch,probs,threshold = .5)
  AIC<-AIC(modelobject)
  BIC<-BIC(modelobject)
  return(list("Accuracy"=Accuracy,"MissClass"=MissClass,"AIC"=AIC,"BIC"=BIC))
}

Results1<-prediction_accuracy(fit1)
Results2<-prediction_accuracy(fit2)
Results3<-prediction_accuracy(fit3)
Results4<-prediction_accuracy(fit4)
Results5<-prediction_accuracy(fit5)


Model<-c('fit1','fit2','fit3','fit4','fit5')

Accuracy<-c(Results1$Accuracy,Results2$Accuracy,Results3$Accuracy,Results4$Accuracy,Results5$Accuracy)
MissClass<-c(Results1$MissClass,Results2$MissClass,Results3$MissClass,Results4$MissClass,Results5$MissCl
AIC_Values<-c(Results1$AIC,Results2$AIC,Results3$AIC,Results4$AIC,Results5$AIC)
BIC_Values<-c(Results1$BIC,Results2$BIC,Results3$BIC,Results4$BIC,Results5$BIC)


df<-data.frame(Model,Accuracy,MissClass,AIC_Values,BIC_Values)
df
# kable(df,col.names = c("Model","Accuracy","MissClass","AIC","BIC"))%>%
# kable_classic(full_width = F, html_font = "Cambria")%>%
# kable_styling(latex_options="scale_down")

summary(fit4)$coef

probs<-predict(fit4,test,type="response")
data=data.frame(prob=probs, arsenic=test$arsenic,dist=test$dist,educ=test$educ)
p1<-ggplot(data=data,aes(arsenic,prob))+theme_minimal()+geom_smooth()+ylab("Predicted Probability")
p2<-ggplot(data=data,aes(dist,prob))+theme_minimal()+geom_smooth()+ylab("Predicted Probability")
p3<-ggplot(data=data,aes(educ,prob))+theme_minimal()+geom_smooth()+ylab("Predicted Probability")
grid.arrange(p1,p2,p3)
# Confusion Matrix
glm.probs=predict(fit4,test,type="response")
glm.pred=rep("0" ,length(glm.probs))
glm.pred[glm.probs >.5]="1"
table<-table(glm.pred ,test$switch)


Acc<- (table[1]+table[4])/length(glm.pred)
sensitivity<- (table[4]/(table[4]+table[3]))
specificity<- (table[1]/(table[1]+table[2]))
error_rate<-1-Acc

df<-data.frame("Accuracy"=Acc,"MissClass"=error_rate,"Sensitivity"=sensitivity,"Specificity"=specificity
df
```

```r
# kable(df)%>%
# kable_classic(full_width = F, html_font = "Cambria")%>%
# kable_styling(latex_options="scale_down")

fit_ROC<-roc(response=test$switch,predictor = glm.probs)
roc.plot<-ggroc(fit_ROC)+geom_abline(slope = 1,intercept = 1,linetype="dashed",alpha=.7)+coord_equal()+
roc.plot

roc.data=data.frame(cutpoints=fit_ROC$thresholds,
                    sensitivties=fit_ROC$sensitivities,
                    specificities=fit_ROC$specificities)

head(roc.data%>%filter(cutpoints>=.5))
# kable(head(roc.data%>%filter(cutpoints>=.5)),caption = "Sample Thresholds")%>%
# kable_classic(full_width = F, html_font = "Cambria")%>%
# kable_styling(latex_options="scale_down")

# Confusion Matrix
glm.probs=predict(fit4,test,type="response")
glm.pred=rep("0" ,length(glm.probs))
glm.pred[glm.probs >.58]="1"
table<-table(glm.pred ,test$switch)

# Percentage
Perc<- (table[1]+table[4])/length(glm.pred)
print(paste("Percentage of Correct Predictions:", Perc))
table

# Confusion Matrix
glm.probs=predict(fit4,test,type="response")
glm.pred=rep("0" ,length(glm.probs))
glm.pred[glm.probs >.68]="1"
table<-table(glm.pred ,test$switch)

# Percentage
Perc<- (table[1]+table[4])/length(glm.pred)
print(paste("Percentage of Correct Predictions:", Perc))
table
```