Homework 03

Logistic Regression

Xinyi Wang

September 11, 2018

Data analysis

1992 presidential election

The folder `nes` contains the survey data of presidential preference and income for the 1992 election analyzed in Section 5.1, along with other variables including sex, ethnicity, education, party identification, and political ideology.

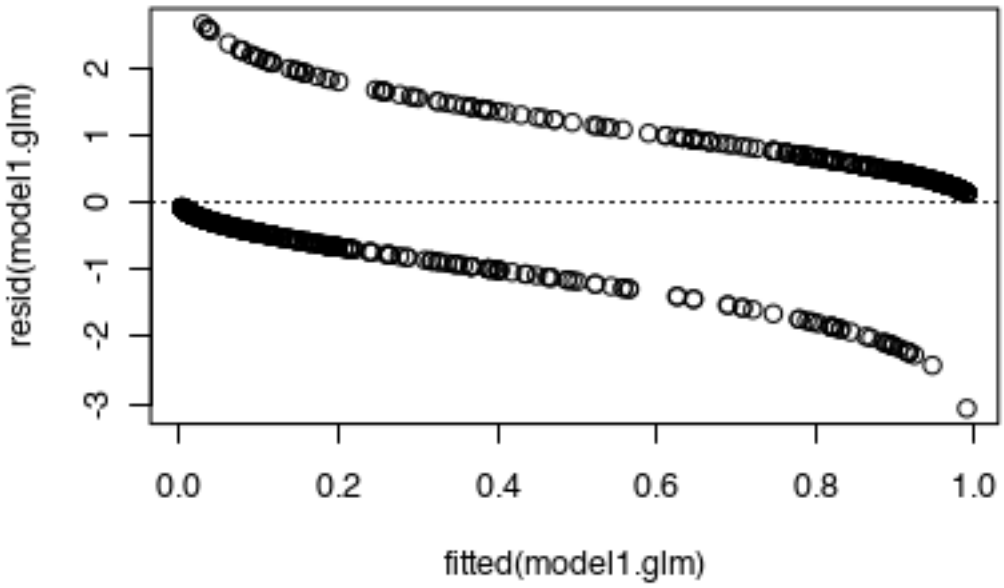
1. Fit a logistic regression predicting support for Bush given all these inputs. Consider how to include these as regression predictors and also consider possible interactions.

```
a = select(nes5200_dt_s,income,female,race,educ1,partyid7,real_ideo,vote_rep)
new_nes = na.omit(a)
new_nes$income = as.integer(new_nes$income)
new_nes$race = as.integer(new_nes$race) - 1
new_nes$educ1 = as.integer(new_nes$educ1)
new_nes$partyid7 = as.integer(new_nes$partyid7)
ideo_feel = new_nes$ideo_feel
#new_nes$ideo_feel = (ideo_feel-mean(ideo_feel))/(2*sd(ideo_feel))
model1.glm = glm(vote_rep~income+female+race+educ1+partyid7+real_ideo,new_nes,family=binomial(link="logi
summary(model1.glm)
```

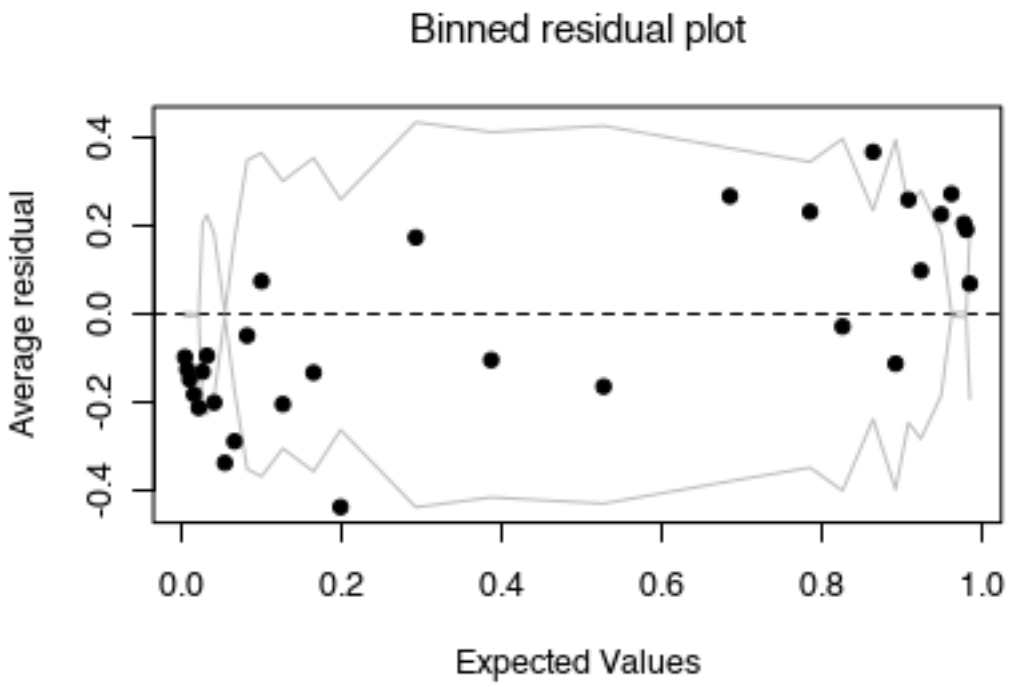
```
##
## Call:
## glm(formula = vote_rep ~ income + female + race + educ1 + partyid7 +
##     real_ideo, family = binomial(link = "logit"), data = new_nes)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0700  -0.3885  -0.1307   0.3940   2.6526
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.7818857  0.8092256 -10.852  < 2e-16 ***
## income      -0.0009922  0.1131949  -0.009    0.993
## female       0.1494255  0.2268369   0.659    0.510
## race         0.0506804  0.1239162   0.409    0.683
## educ1        0.0908412  0.1351263   0.672    0.501
## partyid7     1.0005305  0.0670931  14.913  < 2e-16 ***
## real_ideo    0.7187056  0.0970062   7.409 1.27e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1296.32  on 947  degrees of freedom
```

```
## Residual deviance:  545.14  on 941  degrees of freedom
## AIC: 559.14
##
## Number of Fisher Scoring iterations: 6
```

```
plot(fitted(model1.glm),resid(model1.glm)); abline(h=0,lty=3)
```




```
binnedplot(fitted(model1.glm),resid(model1.glm))
```



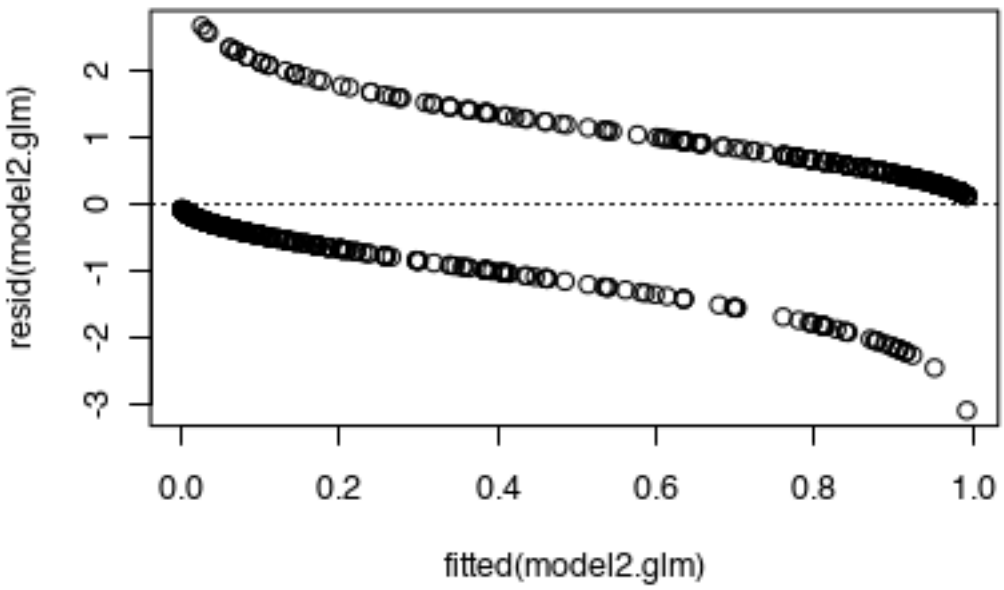
2. Evaluate and compare the different models you have fit. Consider coefficient estimates and standard errors, residual plots, and deviances.

```
model2.glm = glm(vote_rep~income+female+race+educ1+partyid7+real_ideo+race*female,new_nes,family=binomia
summary(model2.glm)
```

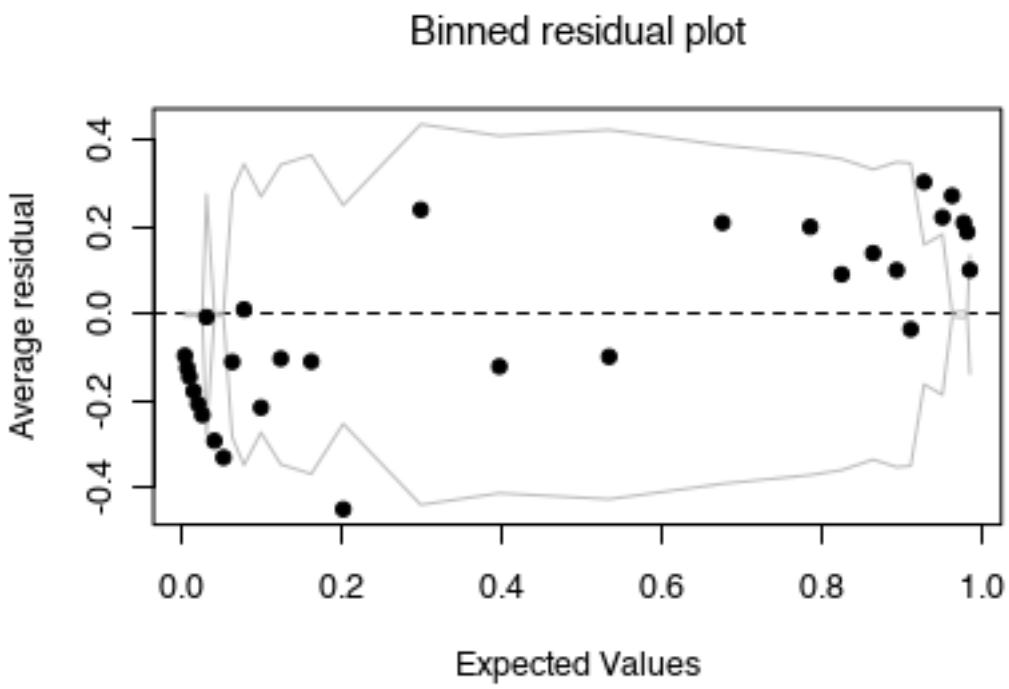
```
##
## Call:
## glm(formula = vote_rep ~ income + female + race + educ1 + partyid7 +
##     real_ideo + race * female, family = binomial(link = "logit"),
##     data = new_nes)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0959  -0.3805  -0.1283   0.3854   2.6840
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.745520   0.812779 -10.760  < 2e-16 ***
## income       0.007837   0.113321   0.069   0.9449
## female       0.008749   0.241558   0.036   0.9711
## race        -0.229645   0.209001  -1.099   0.2719
## educ1        0.095798   0.134927   0.710   0.4777
## partyid7     1.009469   0.067695  14.912  < 2e-16 ***
## real_ideo    0.706203   0.097519   7.242 4.43e-13 ***
## female:race  0.445288   0.258579   1.722   0.0851 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 1296.3  on 947  degrees of freedom
## Residual deviance:  542.1  on 940  degrees of freedom
## AIC: 558.1
##
## Number of Fisher Scoring iterations: 6
```

```
plot(fitted(model2.glm),resid(model2.glm)); abline(h=0,lty=3)
```




```
binnedplot(fitted(model2.glm),resid(model2.glm))
```

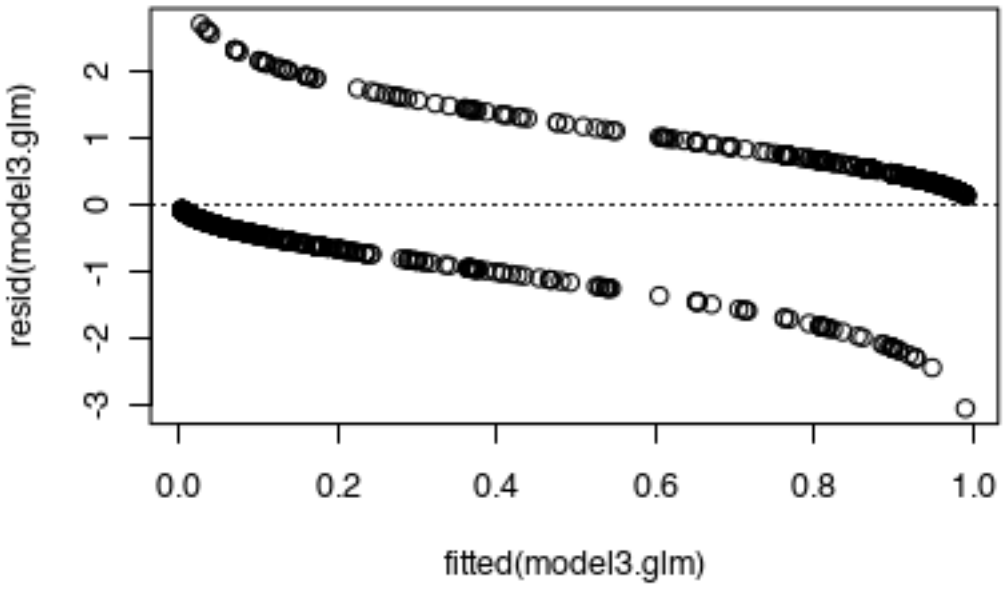



```
model3.glm = glm(vote_rep~income+female+race+educ1+partyid7+real_ideo+female*educ1,new_nes,family=binom
summary(model3.glm)
```

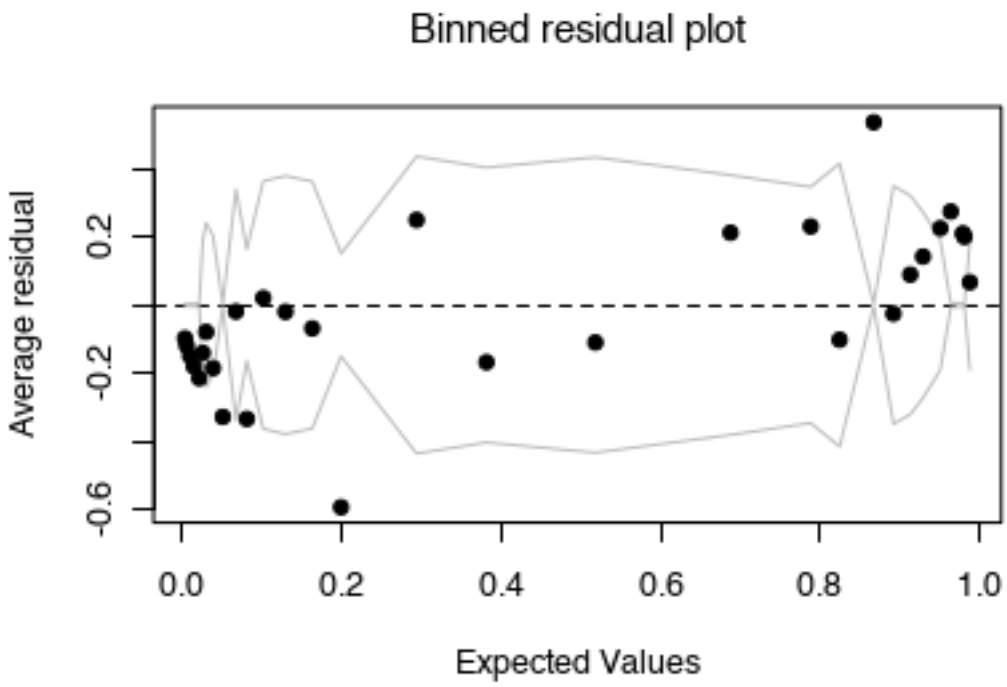
```
##
## Call:
## glm(formula = vote_rep ~ income + female + race + educ1 + partyid7 +
##     real_ideo + female * educ1, family = binomial(link = "logit"),
##     data = new_nes)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0423  -0.3847  -0.1304   0.3881   2.6904
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -8.380658   0.927085  -9.040  < 2e-16 ***
## income       -0.002337   0.113338  -0.021    0.984
## female       -0.696139   1.008001  -0.691    0.490
## race          0.057099   0.124590   0.458    0.647
## educ1        -0.016688   0.183647  -0.091    0.928
## partyid7      1.004050   0.067368  14.904  < 2e-16 ***
## real_ideo     0.722724   0.097257   7.431 1.08e-13 ***
## female:educ1  0.215118   0.250014   0.860    0.390
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1296.3  on 947  degrees of freedom
```

```
## Residual deviance:  544.4  on 940  degrees of freedom
## AIC: 560.4
##
## Number of Fisher Scoring iterations: 6
```

```
plot(fitted(model3.glm),resid(model3.glm)); abline(h=0,lty=3)
```




```
binnedplot(fitted(model3.glm),resid(model3.glm))
```



Compared to three models above, model 2 has smaller residual deviance and AIC which means this model better fits to data. The interaction term gender:race also has smaller p-value than gender:education.

3. For your chosen model, discuss and compare the importance of each input variable in the prediction.

```
display(model2.glm)
```

```
## glm(formula = vote_rep ~ income + female + race + educ1 + partyid7 +
##     real_ideo + race * female, family = binomial(link = "logit"),
##     data = new_nes)
##             coef.est coef.se
## (Intercept) -8.75     0.81
## income       0.01     0.11
## female       0.01     0.24
## race        -0.23     0.21
## educ1        0.10     0.13
## partyid7     1.01     0.07
## real_ideo    0.71     0.10
## female:race  0.45     0.26
## ---
##   n = 948, k = 8
##   residual deviance = 542.1, null deviance = 1296.3 (difference = 754.2)
```

intercept: A male with catagory of income,race,educ1,partyid7 and real_ideo equal to 0 would have log odds of -8.75 to vote for George W. Bush.

partyid7: With the same level of all the rest variables, when party level increases by 1, then the expected value of the voter's log odds of support for Bush would decrease by 1.01 unit.

real_ideo: With the same level of all the rest variables, when real_ideo level increases by 1, then the expected value of the voter's log odds of support for Bush would decrease by 0.71 unit.

female:race: With the same level of all the rest variables, for each additional level of race, the value 0.45 is added to the coefficient for female.

income,gender,race and educ1 are not significant in choosen model 2.

Graphing logistic regressions:

the well-switching data described in Section 5.4 of the Gelman and Hill are in the folder `arsenic`.

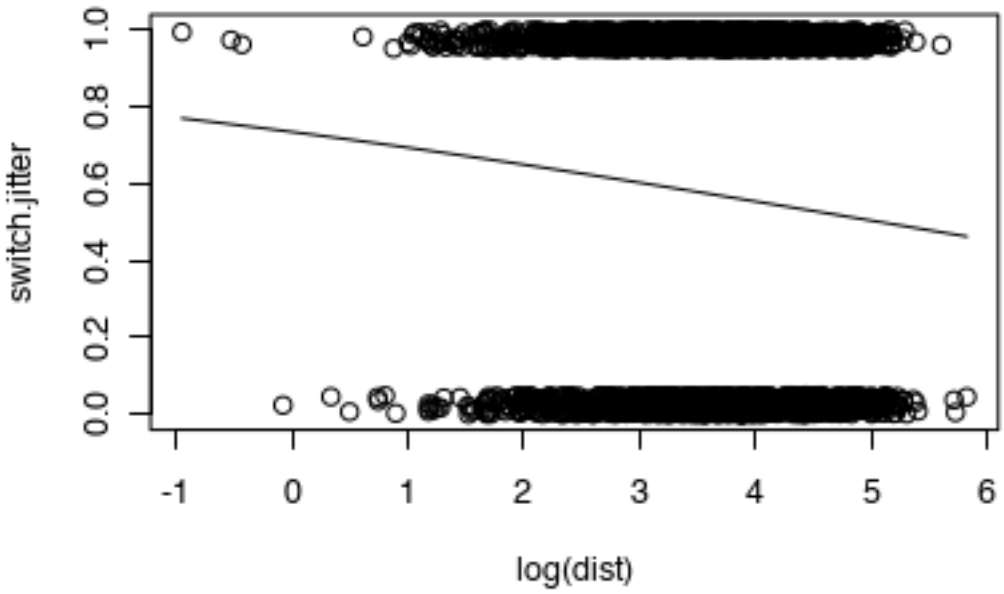
1. Fit a logistic regression for the probability of switching using log (distance to nearest safe well) as a predictor.

```
wells.glm = glm(switch~log(dist),data=wells_dt,family=binomial)
```

2. Make a graph similar to Figure 5.9 of the Gelman and Hill displaying Pr(switch) as a function of distance to nearest safe well, along with the data.

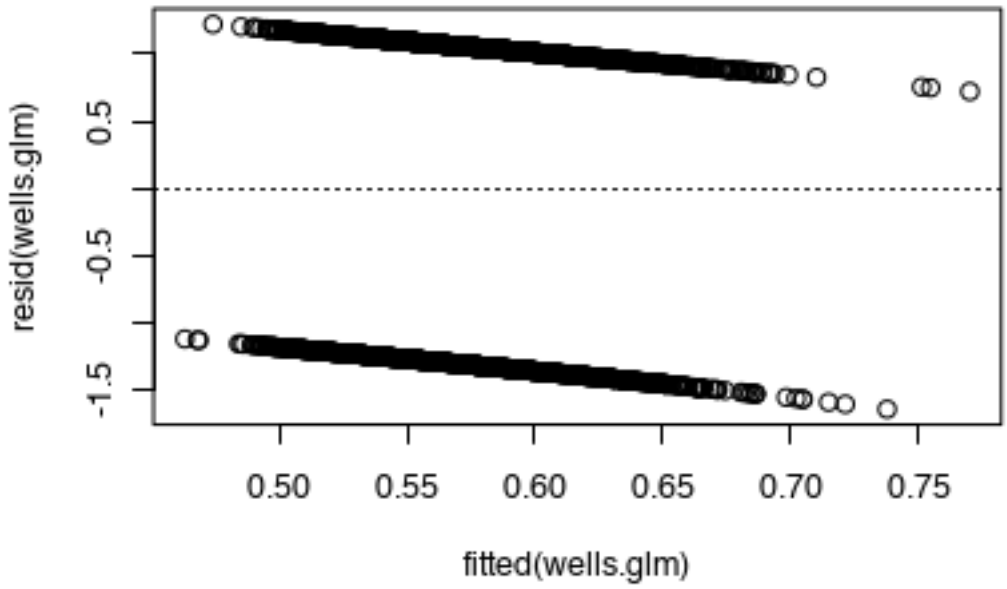
```
dist = wells_dt$dist
switch = wells_dt$switch

jitter.binary = function(a, jitt=.05){
ifelse (a==0, runif (length(a), 0, jitt), runif (length(a), 1-jitt, 1))
}
switch.jitter = jitter.binary (switch)
plot (log(dist), switch.jitter)
curve (invlogit (coef(wells.glm) [1] + coef(wells.glm) [2]*x), add=TRUE)
```

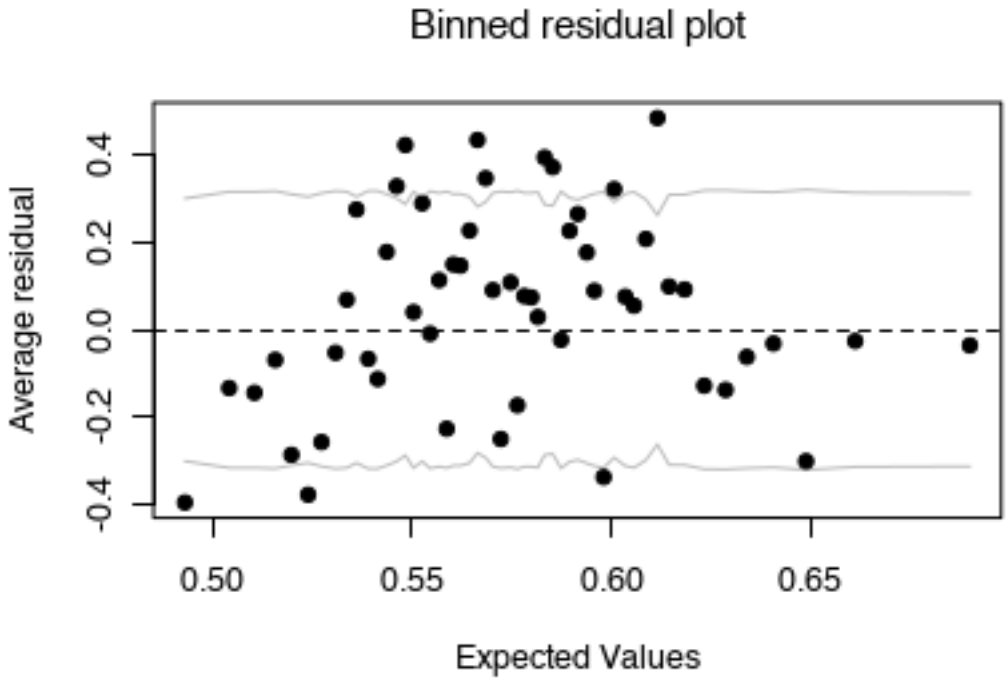


3. Make a residual plot and binned residual plot as in Figure 5.13.

```
plot(fitted(wells.glm),resid(wells.glm)); abline(h=0,lty=3)
```




```
binnedplot(fitted(wells.glm),resid(wells.glm))
```



4. Compute the error rate of the fitted model and compare to the error rate of the null model.

```
n = nrow(wells_dt)
fitted = fitted(wells.glm)
error.rate = mean ((fitted>0.5 & switch==0) | (fitted<0.5 & switch==1))
error.rate
```

```
## [1] 0.4192053
```

```
wells2.glm = glm(switch ~ 1,wells_dt,family = binomial)
fitted2 = fitted(wells2.glm)
error.rate2 = mean ((fitted2>0.5 & switch==0) | (fitted2<0.5 & switch==1))
error.rate2
```

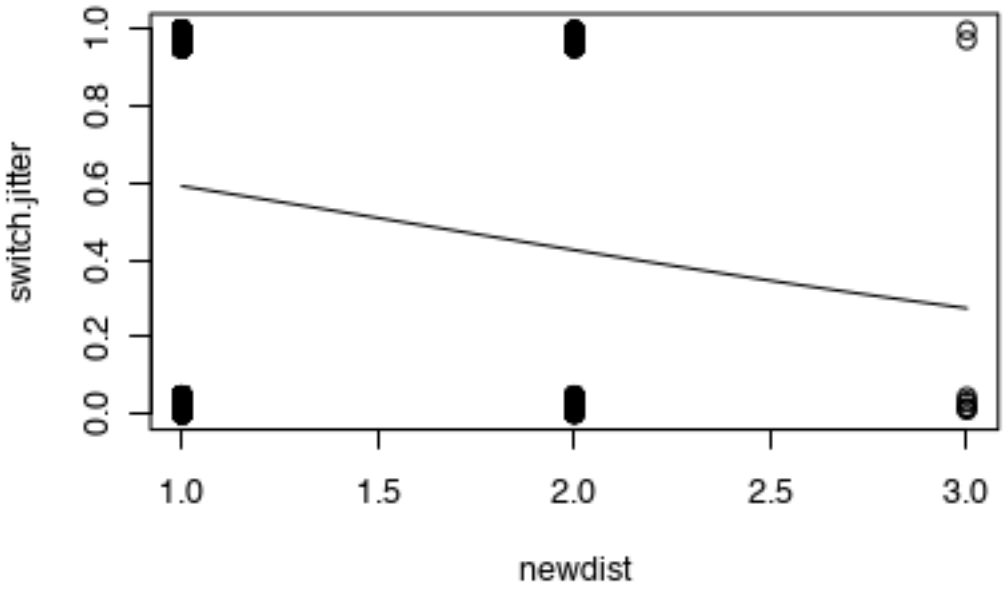
```
## [1] 0.4248344
```

5. Create indicator variables corresponding to `dist < 100`, `100 =< dist < 200`, and `dist > 200`. Fit a logistic regression for Pr(switch) using these indicators. With this new model, repeat the computations and graphs for part (1) of this exercise.

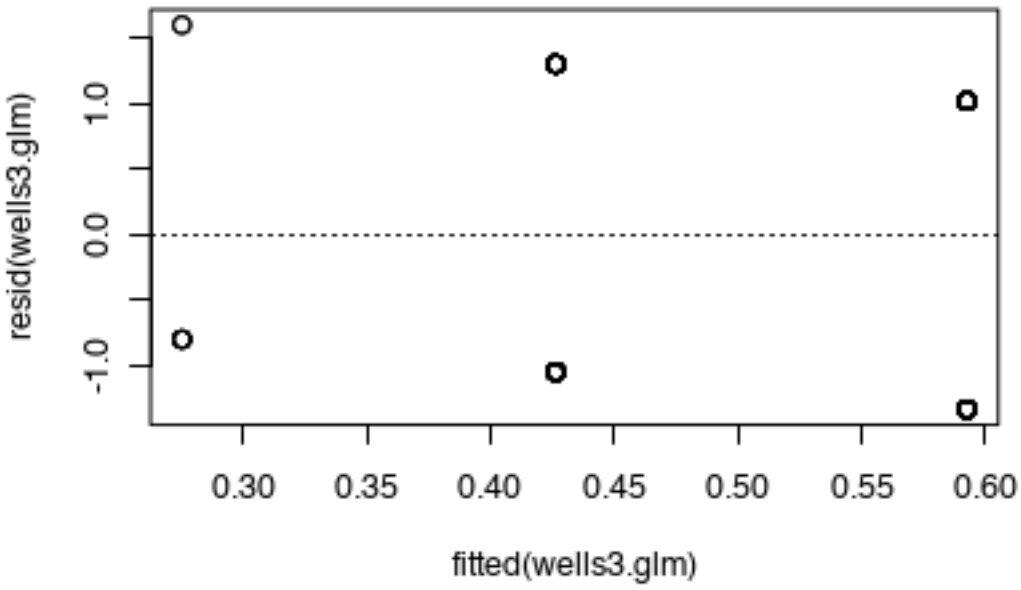
```
newdist = wells_dt$dist
newdist[newdist<100] = 1
newdist[newdist>=100 & newdist<200] = 2
newdist[newdist>=200] = 3
wells3.glm = glm(switch~newdist,family=binomial(link="logit"),data = wells_dt)
summary(wells3.glm)
```

```
##
## Call:
## glm(formula = switch ~ newdist, family = binomial(link = "logit"),
##     data = wells_dt)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.340  -1.340   1.023   1.023   1.606
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.0456     0.1353   7.727 1.10e-14 ***
## newdist      -0.6712     0.1178  -5.697 1.22e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4118.1  on 3019  degrees of freedom
## Residual deviance: 4084.8  on 3018  degrees of freedom
## AIC: 4088.8
##
## Number of Fisher Scoring iterations: 4
```

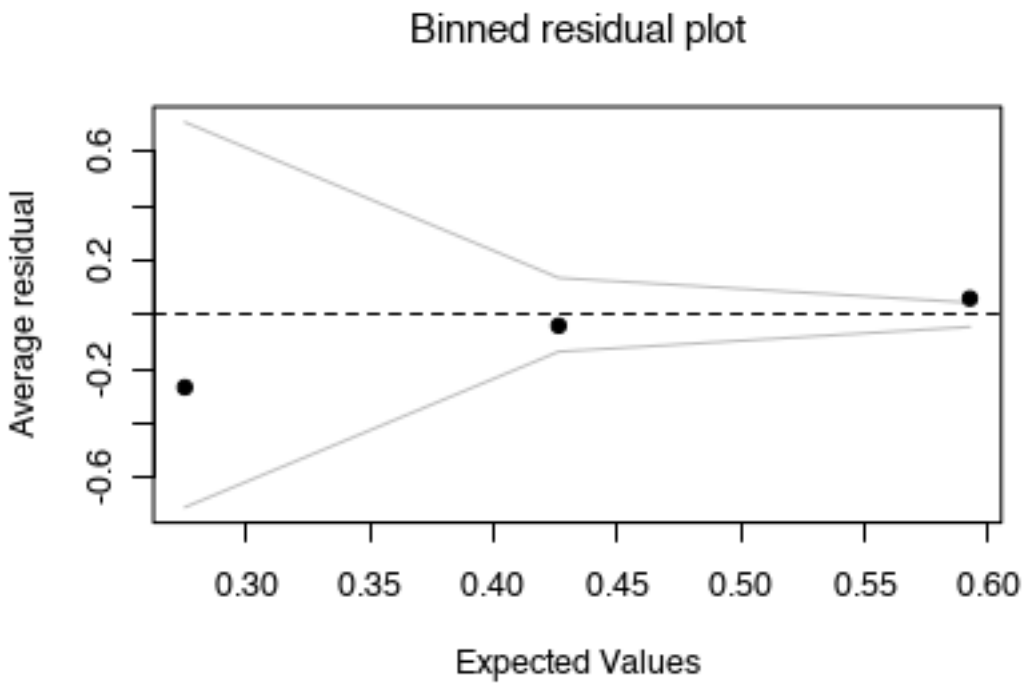
```
jitter.binary <- function(a, jitt=.05){
ifelse (a==0, runif (length(a), 0, jitt), runif (length(a), 1-jitt, 1))
}
switch.jitter = jitter.binary (wells_dt$switch)
plot (newdist, switch.jitter)
curve (invlogit (coef(wells3.glm)[1] + coef(wells3.glm)[2]*x), add=TRUE)
```




```
plot(fitted(wells3.glm),resid(wells3.glm)); abline(h=0,lty=3)
```




```
binnedplot(fitted(wells3.glm),resid(wells3.glm))
```




```
n2 = nrow(wells_dt)
fitted3 = fitted(wells3.glm)
error.rate3 = mean ((fitted3>0.5 & switch==0) | (fitted3<0.5 & switch==1))
error.rate3
```

```
## [1] 0.4092715
```

```
wells4.glm = glm(switch ~ 1,wells_dt,family = binomial)
fitted4 = fitted(wells4.glm)
error.rate4 = mean ((fitted4>0.5 & switch==0) | (fitted4<0.5 & switch==1))
error.rate4
```

```
## [1] 0.4248344
```

Model building and comparison:

continue with the well-switching data described in the previous exercise.

1. Fit a logistic regression for the probability of switching using, as predictors, distance, `log(arsenic)`, and their interaction. Interpret the estimated coefficients and their standard errors.

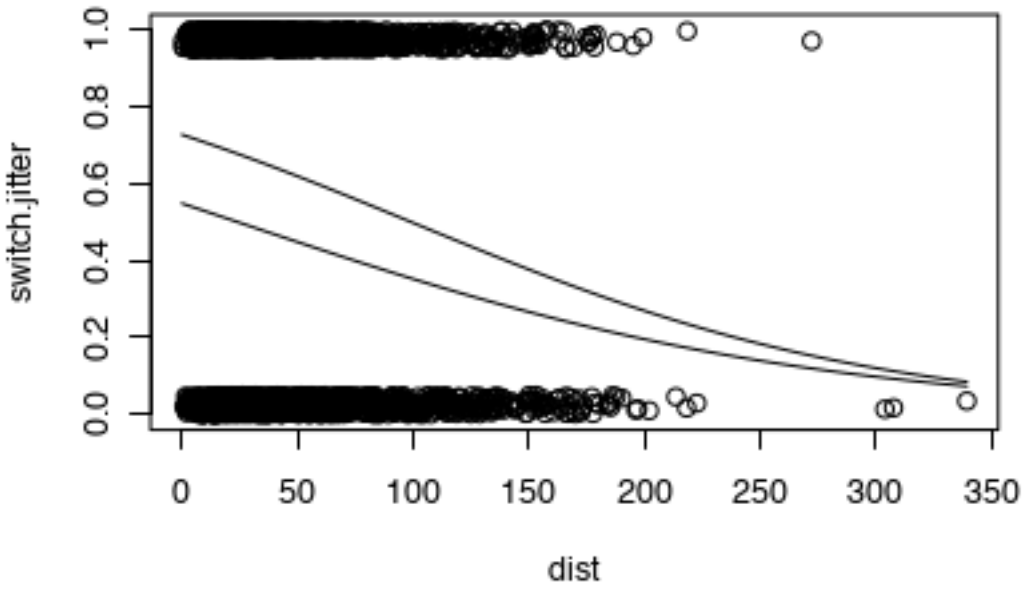
```
arsenic = wells_dt$arsenic
wells_dt$log.arsenic = log(wells_dt$arsenic)
model1 = glm(switch ~ dist * log.arsenic, family=binomial(link="logit"), data=wells_dt)
display(model1)
```

```
## glm(formula = switch ~ dist * log.arsenic, family = binomial(link = "logit"),
##     data = wells_dt)
##                 coef.est coef.se
## (Intercept)        0.49     0.07
## dist              -0.01     0.00
```

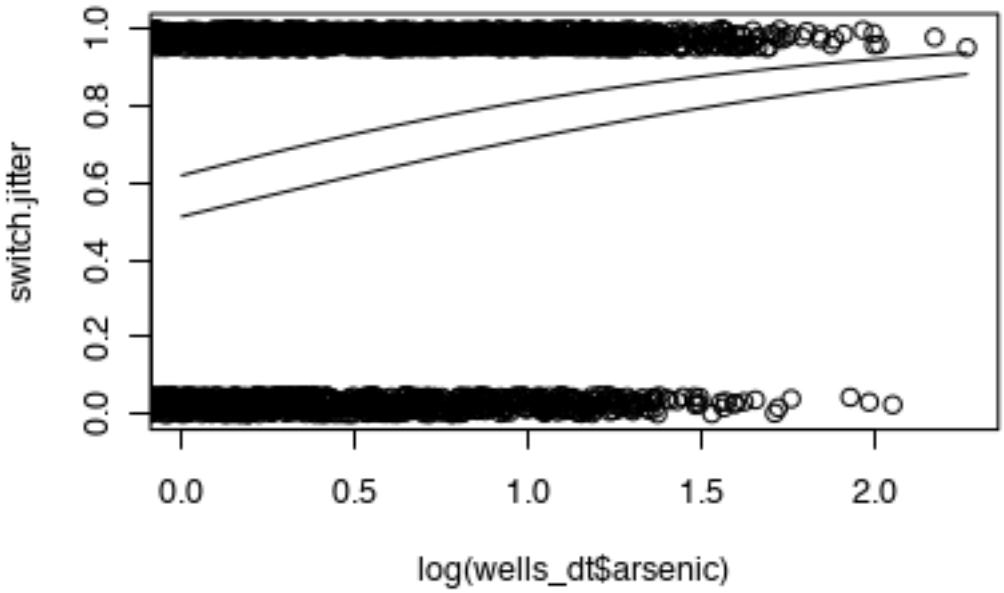
```
## log.arsenic       0.98     0.11
## dist:log.arsenic  0.00     0.00
## ---
##   n = 3020, k = 4
##   residual deviance = 3896.8, null deviance = 4118.1 (difference = 221.3)
```

2. Make graphs as in Figure 5.12 to show the relation between probability of switching, distance, and arsenic level.

```
plot(dist,switch.jitter,xlim=c(0,max(dist)))
curve(invlogit(cbind(1,x,0.5,0.5*x) %*% coef(model1)),add=TRUE)
curve(invlogit(cbind(1,x,-0.3,-0.3*x) %*% coef(model1)),add=TRUE)
```




```
plot(log(wells_dt$arsenic),switch.jitter,xlim=c(0,max(log(wells_dt$arsenic))))
curve(invlogit(cbind(1,0,x,0*x) %*% coef(model1)),add=TRUE)
curve(invlogit(cbind(1,50,x,50*x) %*% coef(model1)),add=TRUE)
```



3. Following the procedure described in Section 5.7, compute the average predictive differences corresponding to:

i. A comparison of dist = 0 to dist = 100, with arsenic held constant.
ii. A comparison of dist = 100 to dist = 200, with arsenic held constant.
iii. A comparison of arsenic = 0.5 to arsenic = 1.0, with dist held constant.
iv. A comparison of arsenic = 1.0 to arsenic = 2.0, with dist held constant. Discuss these results.

```
#i.
b <- coef(model1)
hi <- 100
lo <- 0
delta <- invlogit(b[1] + b[2]*hi + b[3]*wells_dt$log.arsenic + b[4]*wells_dt$log.arsenic*hi) -
         invlogit(b[1] + b[2]*lo + b[3]*wells_dt$log.arsenic + b[4]*wells_dt$log.arsenic*lo)
print(mean(delta))
```

```
## [1] -0.2113356
```

```
#ii.
b <- coef(model1)
hi <- 200
lo <- 100
delta <- invlogit(b[1] + b[2]*hi + b[3]*wells_dt$log.arsenic + b[4]*wells_dt$log.arsenic*hi) -
         invlogit(b[1] + b[2]*lo + b[3]*wells_dt$log.arsenic + b[4]*wells_dt$log.arsenic*lo)
print(mean(delta))
```

```
## [1] -0.2090207
```

```
#iii.
b <- coef(model1)
hi <- 1.0
lo <- 0.5
```

```
delta <- invlogit(b[1] + b[2]*wells_dt$dist + b[3]*hi + b[4]*wells_dt$dist*hi) -
         invlogit(b[1] + b[2]*wells_dt$dist + b[3]*lo + b[4]*wells_dt$dist*lo)
print(mean(delta))
```

```
## [1] 0.09195206
```

```
#iv.
b <- coef(model1)
hi <- 2.0
lo <- 1.0
delta <- invlogit(b[1] + b[2]*wells_dt$dist + b[3]*hi + b[4]*wells_dt$dist*hi) -
         invlogit(b[1] + b[2]*wells_dt$dist + b[3]*lo + b[4]*wells_dt$dist*lo)
print(mean(delta))
```

```
## [1] 0.1353431
```

Building a logistic regression model:

the folder rodents contains data on rodents in a sample of New York City apartments.

Please read for the data details. http://www.stat.columbia.edu/~gelman/arm/examples/rodents/rodents.doc

1. Build a logistic regression model to predict the presence of rodents (the variable y in the dataset) given indicators for the ethnic groups (race). Combine categories as appropriate. Discuss the estimated coefficients in the model.

```
apt_dt = na.omit(apt_dt)
apt_dt$race_comb = "other"
apt_dt$race_comb[apt_dt$asian]<-"asian"
apt_dt$race_comb[apt_dt$black]<-"black"
apt_dt$race_comb[apt_dt$hisp]<-"hisp"
apt_dt$race_comb<-factor(apt_dt$race_comb,levels=c("other","asian","black","hisp"))

model1.glm = glm(y~asian+black+hisp,family=binomial,data=apt_dt)
summary(model1.glm)
```

```
##
## Call:
## glm(formula = y ~ asian + black + hisp, family = binomial, data = apt_dt)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9922  -0.9293  -0.4690  -0.4690   2.1270
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.1521     0.1281 -16.798   <2e-16 ***
## asianTRUE     0.5518     0.2665   2.070   0.0384 *
## blackTRUE     1.5361     0.1687   9.108   <2e-16 ***
## hispTRUE      1.6995     0.1664  10.212   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1672.2  on 1521  degrees of freedom
```

```
## Residual deviance: 1526.3  on 1518  degrees of freedom
## AIC: 1534.3
##
## Number of Fisher Scoring iterations: 4
```

2. Add to your model some other potentially relevant predictors describing the apartment, building, and community district. Build your model using the general principles explained in Section 4.6 of the Gelman and Hill. Discuss the coefficients for the ethnicity indicators in your model.

```
model2.glm = glm(y~defects+poor+floor+asian+black+hisp,family=binomial,data=apt_dt)
summary(model2.glm)
```

```
##
## Call:
## glm(formula = y ~ defects + poor + floor + asian + black + hisp,
##     family = binomial, data = apt_dt)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0276  -0.7066  -0.4085  -0.3256   2.4255
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.018975   0.224223 -13.464  < 2e-16 ***
## defects      0.469617   0.043434  10.812  < 2e-16 ***
## poor         0.170834   0.048006   3.559 0.000373 ***
## floor       -0.009788   0.036578  -0.268 0.789010
## asianTRUE    0.403938   0.284475   1.420 0.155625
## blackTRUE    1.143844   0.183432   6.236 4.50e-10 ***
## hispTRUE     1.286270   0.184931   6.955 3.52e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1672.2  on 1521  degrees of freedom
## Residual deviance: 1349.5  on 1515  degrees of freedom
## AIC: 1363.5
##
## Number of Fisher Scoring iterations: 5
```

Conceptual exercises.

Shape of the inverse logit curve

Without using a computer, sketch the following logistic regression lines:

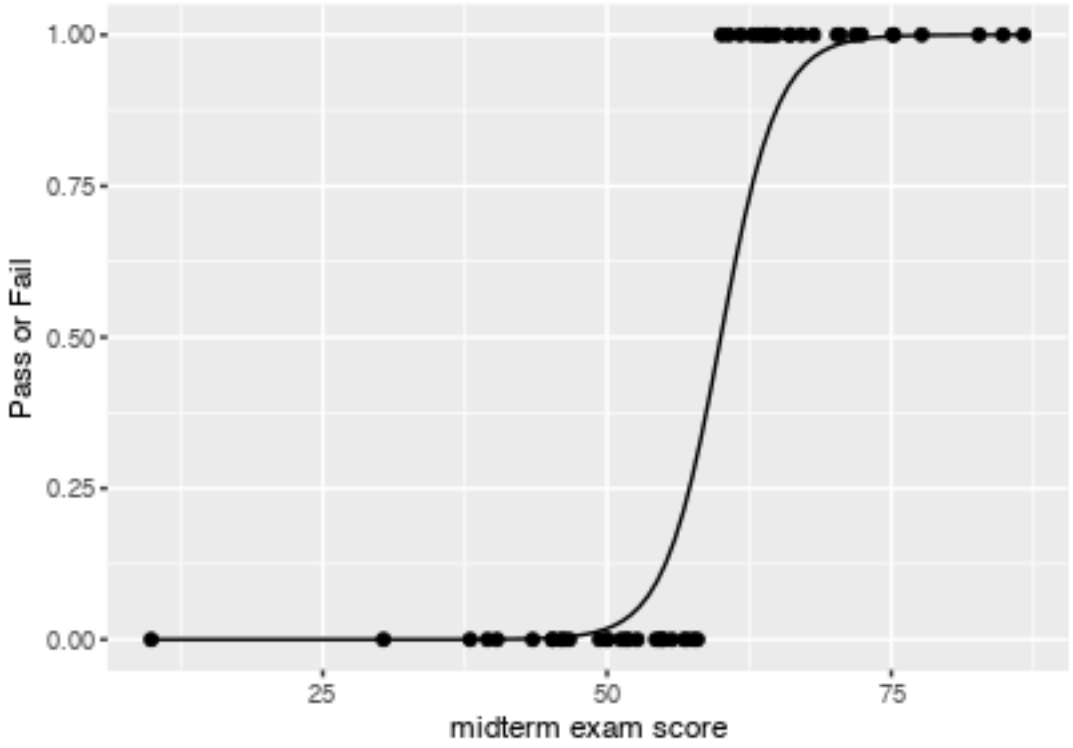
1. $Pr(y = 1) = logit^{-1}(x)$
2. $Pr(y = 1) = logit^{-1}(2 + x)$
3. $Pr(y = 1) = logit^{-1}(2x)$
4. $Pr(y = 1) = logit^{-1}(2 + 2x)$
5. $Pr(y = 1) = logit^{-1}(-2x)$

Please see attached "logit_sketch.pdf"

In a class of 50 students, a logistic regression is performed of course grade (pass or fail) on midterm exam score (continuous values with mean 60 and standard deviation 15). The fitted model is $Pr(pass) = logit^{-1}(-24 + 0.4x)$.

1. Graph the fitted model. Also on this graph put a scatterplot of hypothetical data consistent with the information given.

```
score = rnorm(50, mean=60, sd = 15)
y = invlogit(-24 + 0.4*score)
pass = ifelse(y>0.5,1,0)
ggplot(data.frame(score, pass), aes(x=score, y = pass)) +
  geom_point() +
  stat_function(fun=function(x) invlogit(-24 + 0.4 * x)) +
  labs(x="midterm exam score", y="Pass or Fail")
```



2. Suppose the midterm scores were transformed to have a mean of 0 and standard deviation of 1. What would be the equation of the logistic regression using these transformed scores as a predictor?

plug $c.score = (score - 60)/15$ back to the model, we get

$$Pr(pass) = logit^{-1}(6x)$$

3. Create a new predictor that is pure noise (for example, in R you can create `newpred <- rnorm(n,0,1)`). Add it to your model. How much does the deviance decrease?

```
newpred = rnorm(n = 50, mean = 0,sd = 1)
#original model
deviance(glm(y ~ score , family = "binomial"))
```

```
## [1] 9.306262e-16
```

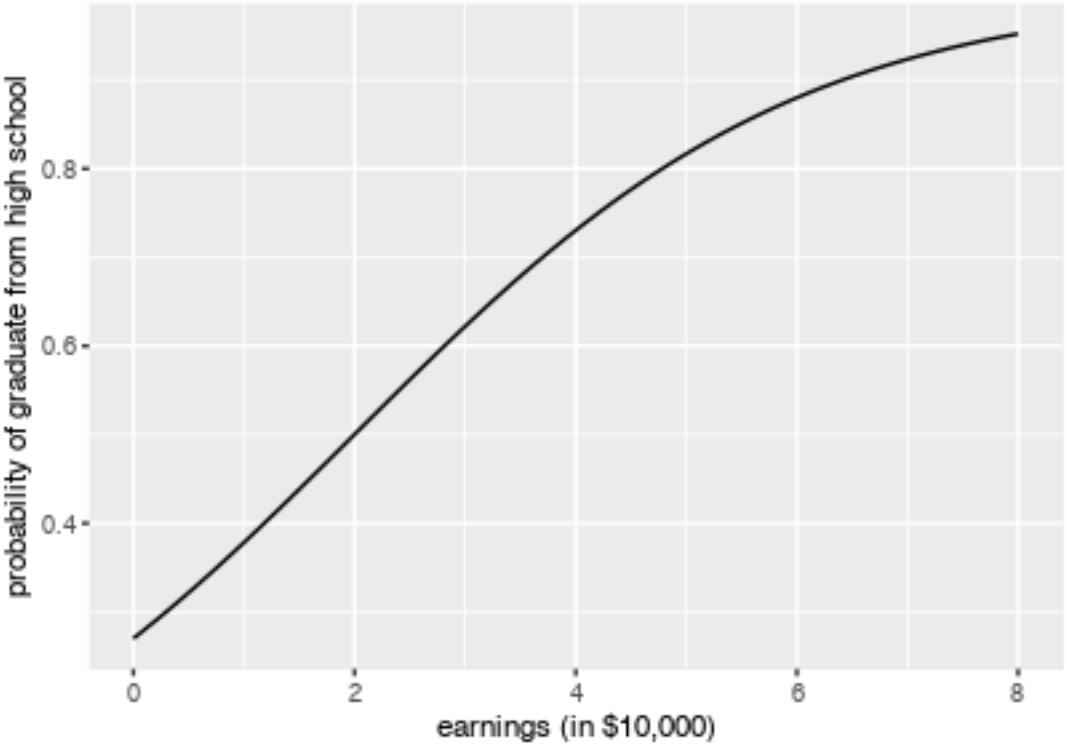
```
#add pure noise
deviance(glm(y ~ score + newpred, family = "binomial"))
```

```
## [1] 6.887834e-16
```

Logistic regression

You are interested in how well the combined earnings of the parents in a child's family predicts high school graduation. You are told that the probability a child graduates from high school is 27% for children whose parents earn no income and is 88% for children whose parents earn $60,000. Determine the logistic regression model that is consistent with this information. (For simplicity you may want to assume that income is measured in units of $10,000).

```
ggplot(data.frame(x=c(0, 8)), aes(x)) +
  stat_function(fun=function(x) invlogit(logit(0.27) + (logit(0.88)-logit(0.27))/6 * x)) +
  labs(x="earnings (in $10,000)", y="probability of graduate from high school")
```

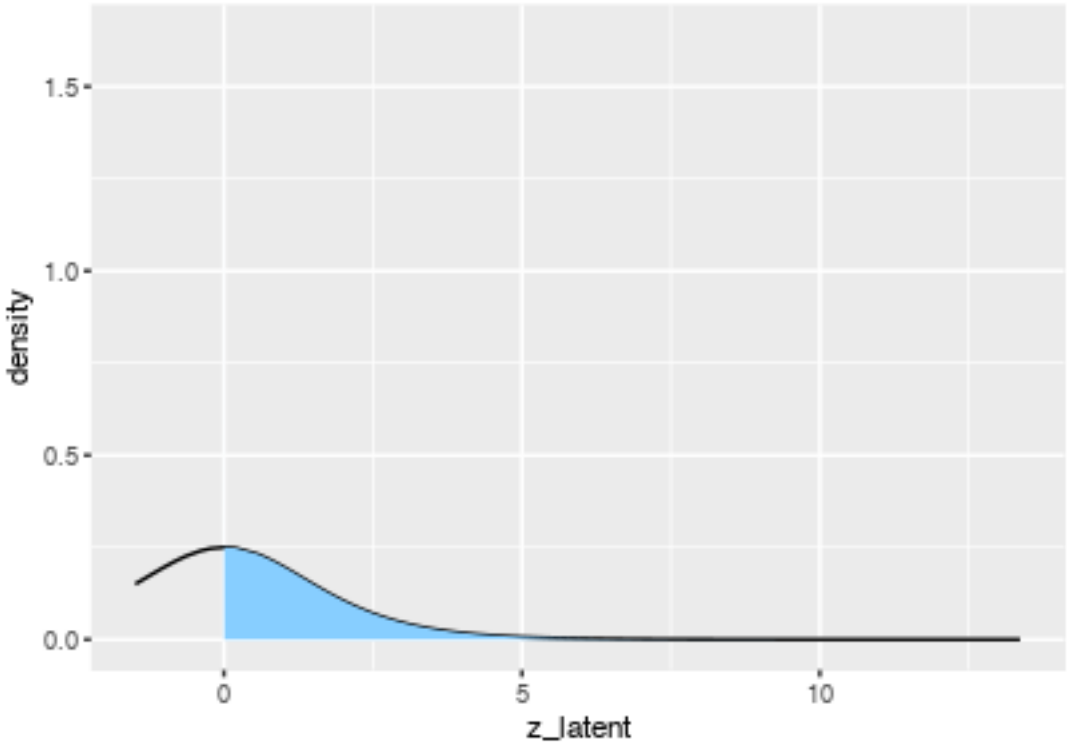


Latent-data formulation of the logistic model:

take the model $Pr(y = 1) = logit^{-1}(1 + 2x_1 + 3x_2)$ and consider a person for whom $x_1 = 1$ and $x_2 = 0.5$. Sketch the distribution of the latent data for this person. Figure out the probability that $y = 1$ for the person and shade the corresponding area on your graph.

```
e = rlogis(1000,0,1)
z_latent = 1 + 2*1 + 3*0.5 + e
density = dlogis(z_latent)
mydata = data.frame(cbind(e,z_latent,density))
```

```
p = ggplot(mydata,aes(x=z_latent,y=density))+
  geom_line()+
  geom_area(aes(x=ifelse(z_latent>0,z_latent,0)),fill="skyblue1")
p
```



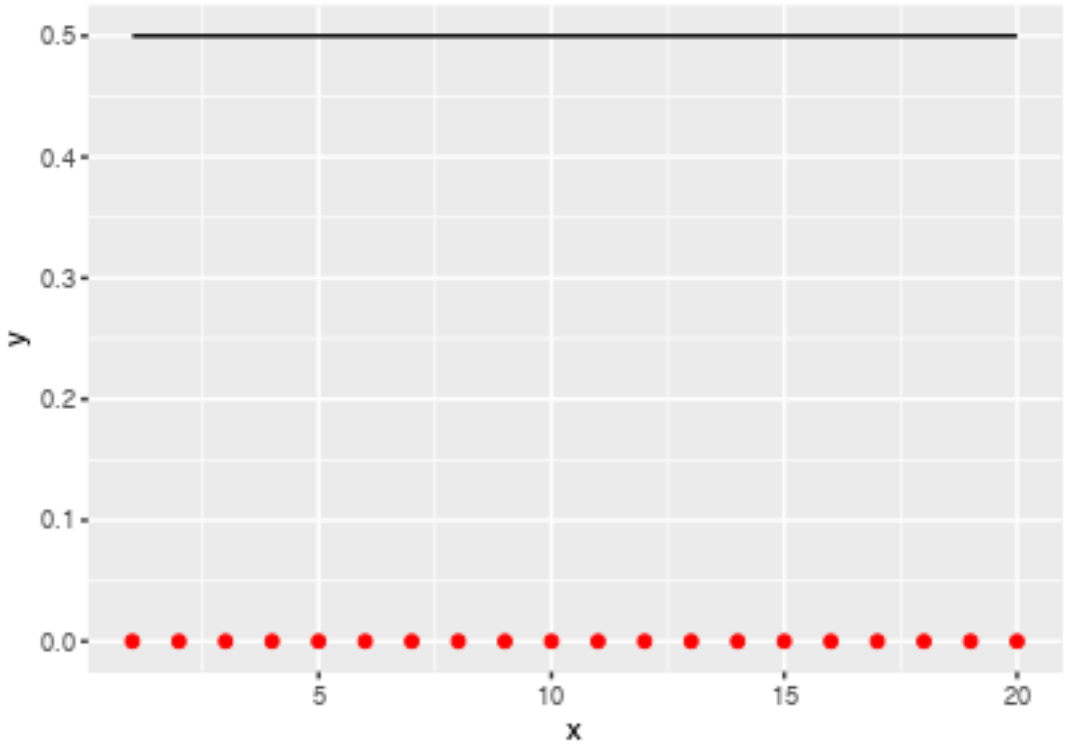

```
#p_df = ggplot_build(p)$data[[1]]
# p = p + geom_area(data = subset(p_df, x >= 0 & x <=15),
#                           aes(x=x,y=y),
#                           fill = "skyblue1",
#                           color = "black")
# p
```

Limitations of logistic regression:

consider a dataset with $n = 20$ points, a single predictor x that takes on the values $1, \ldots, 20$, and binary data y. Construct data values $y_1, \ldots, y_{20}$ that are inconsistent with any logistic regression on x. Fit a logistic regression to these data, plot the data and fitted curve, and explain why you can say that the model does not fit the data.

```
x = c(1:20)
y = rep(0,20)
model.glm = glm(y ~ x)

ggplot(data.frame(x,y), aes(x=x, y=y)) +
  geom_point(color="red") +
  stat_function(fun=function(x) invlogit(coef(model.glm)[1] + coef(model.glm)[2] * x))
```



Identifiability:

the folder nes has data from the National Election Studies that were used in Section 5.1 of the Gelman and Hill to model vote preferences given income. When we try to fit a similar model using ethnicity as a predictor, we run into a problem. Here are fits from 1960, 1964, 1968, and 1972:

```
## glm(formula = vote_rep ~ female + black + income, family = binomial(link = "logit"),
##     data = nes5200_dt_d, subset = (year == 1960))
##             coef.est coef.se
## (Intercept) -0.16     0.23
## female       0.24     0.14
## black       -1.06     0.36
## income       0.03     0.06
## ---
##   n = 877, k = 4
##   residual deviance = 1202.6, null deviance = 1215.7 (difference = 13.1)
```

```
## glm(formula = vote_rep ~ female + black + income, family = binomial(link = "logit"),
##     data = nes5200_dt_d, subset = (year == 1964))
##             coef.est coef.se
## (Intercept)  -1.16     0.22
## female       -0.08     0.14
## black       -16.83   420.51
## income        0.19     0.06
## ---
##   n = 1062, k = 4
##   residual deviance = 1254.0, null deviance = 1337.7 (difference = 83.7)
```

```
## glm(formula = vote_rep ~ female + black + income, family = binomial(link = "logit"),
```

```
##     data = nes5200_dt_d, subset = (year == 1968))
##             coef.est coef.se
## (Intercept)  0.48     0.24
## female      -0.03     0.15
## black       -3.64     0.59
## income      -0.03     0.07
## ---
##   n = 851, k = 4
##   residual deviance = 1066.8, null deviance = 1173.8 (difference = 107.0)
```

```
## glm(formula = vote_rep ~ female + black + income, family = binomial(link = "logit"),
##     data = nes5200_dt_d, subset = (year == 1972))
##             coef.est coef.se
## (Intercept)  0.70     0.18
## female      -0.25     0.12
## black       -2.58     0.26
## income       0.08     0.05
## ---
##   n = 1518, k = 4
##   residual deviance = 1808.3, null deviance = 1973.8 (difference = 165.5)
```

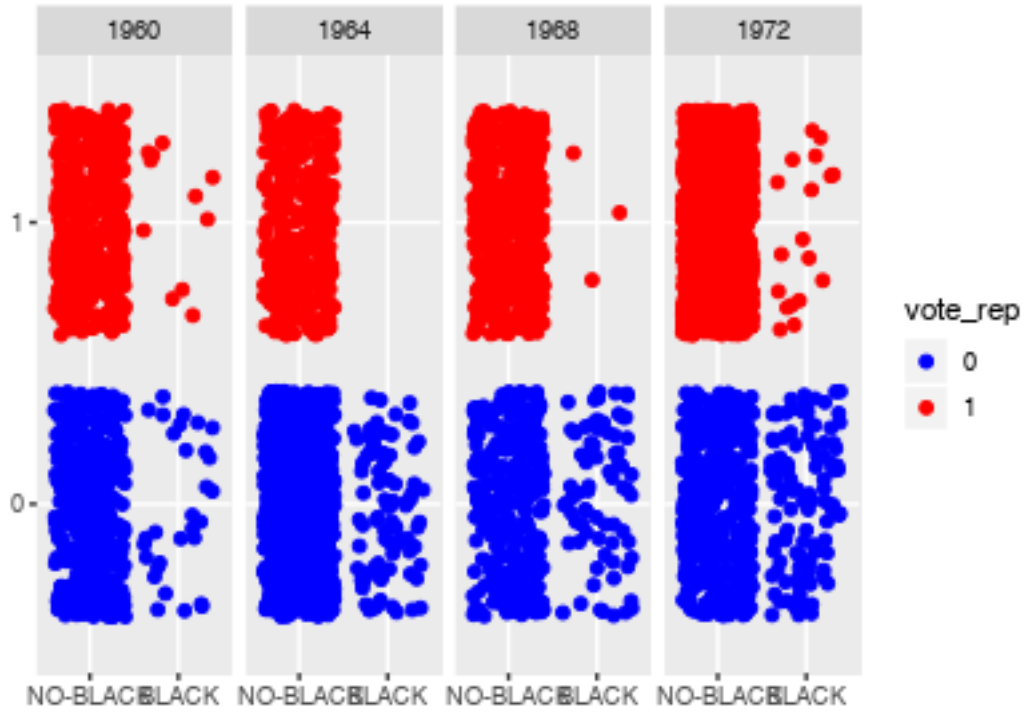
What happened with the coefficient of black in 1964? Take a look at the data and figure out where this extreme estimate came from. What can be done to fit the model in 1964?

```
sub_nes =  nes5200_dt_d %>%
  select(vote_rep, year, female ,black ,income) %>%
  subset( year%in% c(1960, 1964, 1968, 1972) & !is.na(black))
sub_nes$vote_rep <- factor(sub_nes$vote_rep)
sub_nes$female <- factor(sub_nes$female, label=c("MALE","FEMALE"))
sub_nes$black <- factor(sub_nes$black, labels = c("NO-BLACK", "BLACK"))

str(sub_nes)
```

```
## Classes 'data.table' and 'data.frame':   4308 obs. of  5 variables:
##  $ vote_rep: Factor w/ 2 levels "0","1": 1 1 1 2 1 1 1 1 1 1 ...
##  $ year    : num  1960 1960 1960 1960 1960 1960 1960 1960 1960 1960 ...
##  $ female  : Factor w/ 2 levels "MALE","FEMALE": 1 1 1 2 2 2 1 1 1 2 ...
##  $ black   : Factor w/ 2 levels "NO-BLACK","BLACK": 1 1 1 1 1 1 1 1 2 2 ...
##  $ income  : int  4 4 2 1 1 2 1 1 3 1 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```
ggplot(sub_nes)+
  aes(x=black,y=vote_rep,color=vote_rep)+geom_jitter()+
  facet_grid(.~year)+scale_color_manual(values=c("blue","red"))+
  ylab("")+xlab("")
```



Feedback comments etc.

If you have any comments about the homework, or the class, please write your feedback here. We love to hear your opinions.