MA678 Homework 03

Logistic Regression

Your name

September 15, 2017

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.

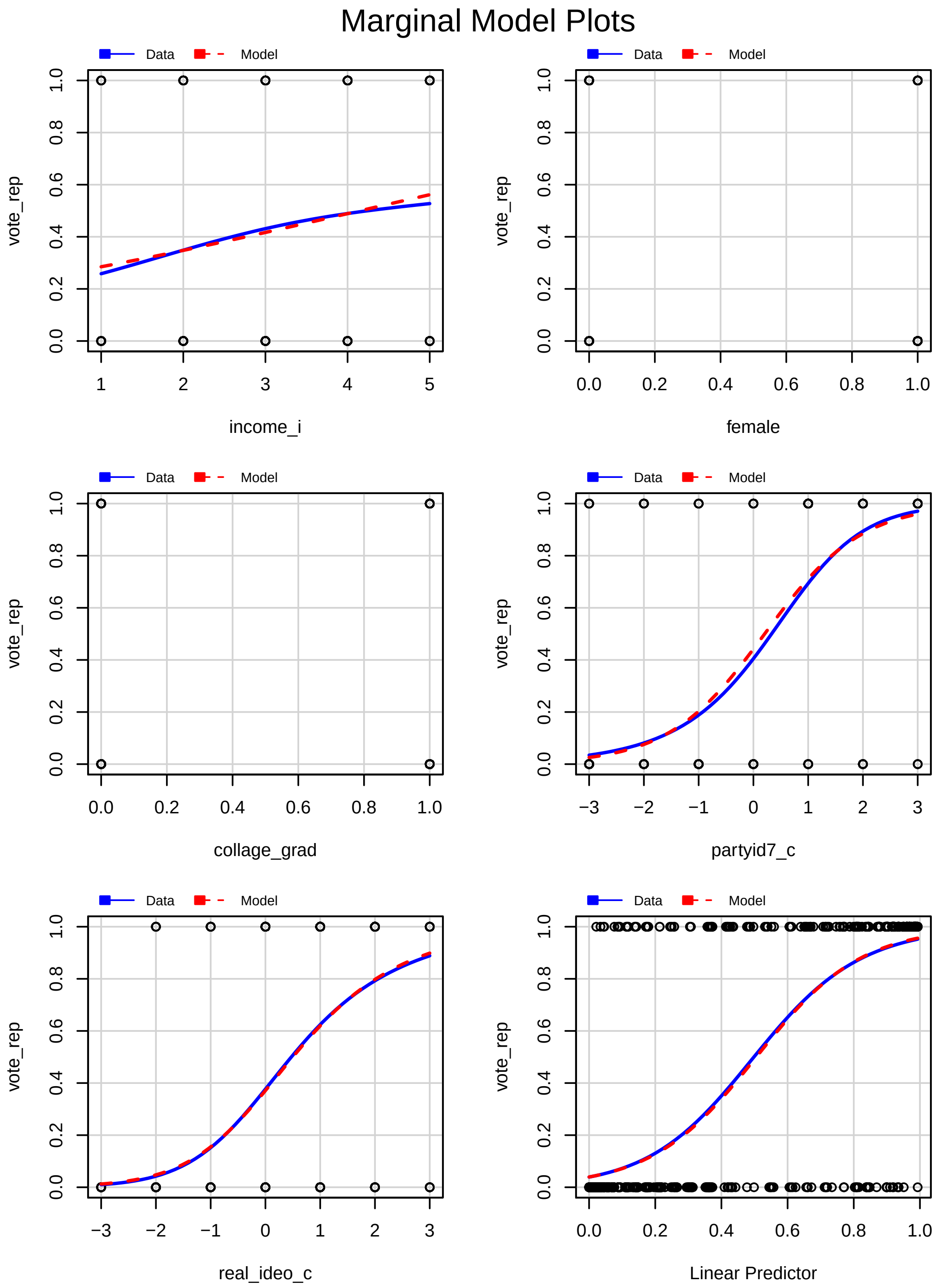
```
nes5200_dt_s$partyid7_c = (as.integer(nes5200_dt_s$partyid7) - 5)
nes5200_dt_s$collage_grad <- (nes5200_dt_s$educ1) == "4. college or advanced degree (no cases"
nes5200_dt_s$income_i<-as.integer(nes5200_dt_s$income)
nes5200_dt_s$real_ideo_c = (as.integer(nes5200_dt_s$real_ideo) - 4)

fit_vote_1<-glm(vote_rep ~ income_i+race+female+collage_grad+partyid7_c+real_ideo_c,
                data=nes5200_dt_s,family=binomial)
display(fit_vote_1)
```

```
## glm(formula = vote_rep ~ income_i + race + female + collage_grad +
##     partyid7_c + real_ideo_c, family = binomial, data = nes5200_dt_s)
##                        coef.est coef.se
## (Intercept)            -0.48     0.41
## income_i               -0.02     0.11
## race2. black           -2.79     0.65
## race3. asian           -0.15     0.85
## race4. native american -0.05     0.72
## race5. hispanic         0.97     0.53
## female                  0.21     0.23
## collage_gradTRUE        0.27     0.26
## partyid7_c              1.00     0.07
## real_ideo_c             0.76     0.10
## ---
##   n = 948, k = 10
##   residual deviance = 512.7, null deviance = 1296.3 (difference = 783.6)
```

We can look at the model fit using `marginalModelPlots`

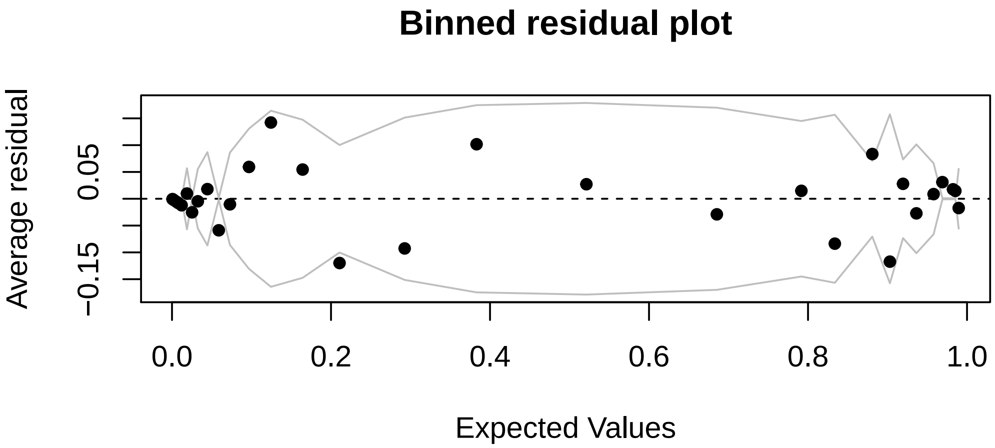
```
marginalModelPlots(fit_vote_1)
```

```
## Warning in mmps(...): Interactions and/or factors skipped
```



The model fits surprisingly well. However, note that `marginalModelPlots` has a problem with binary predictor variable.

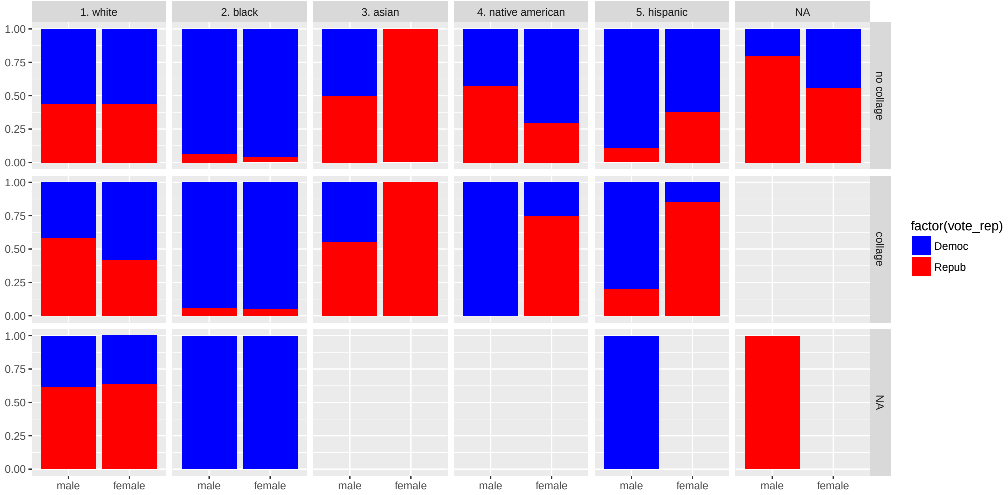
```
binnedplot(fitted(fit_vote_1),residuals(fit_vote_1,type="response"))
```



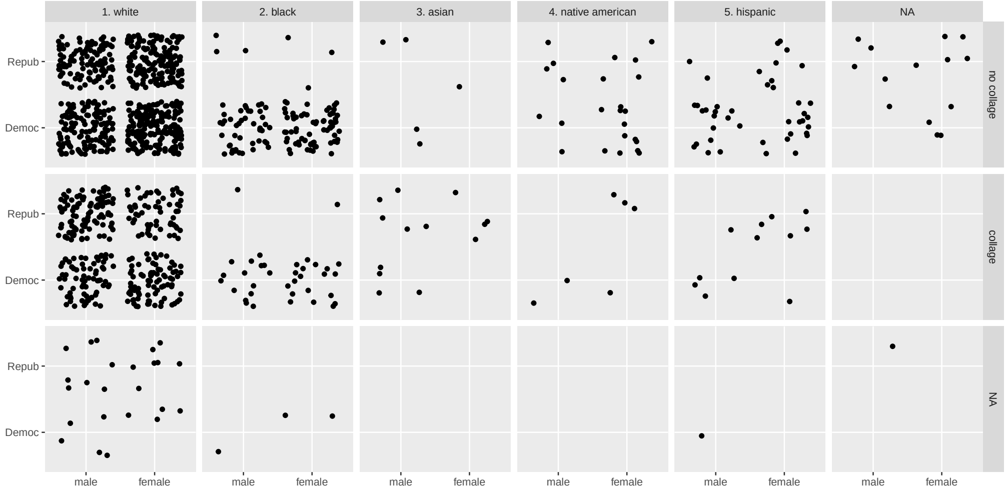
The binned residual looks OK.

Is it interaction between gender, race, and/or education?

```
nnss<- nes5200_dt_s
nnss$female<-factor(nnss$female,labels=c("male","female"))
nnss$collage_grad<-factor(nnss$collage_grad,labels=c("no collage","collage"))
nnss$vote_rep<-factor(nnss$vote_rep,labels=c("Democ","Repub"))
ggplot(nnss)+aes(x=female,fill=factor(vote_rep))+
  geom_bar(position="fill")+facet_grid(collage_grad~race)+
  scale_fill_manual(values=c("blue","red"))+ylab("")+xlab("")
```



```
ggplot(nnss)+aes(x=female,y=factor(vote_rep))+
  geom_jitter()+facet_grid(collage_grad~race)+
  scale_fill_manual(values=c("blue","red"))+ylab("")+xlab("")
```

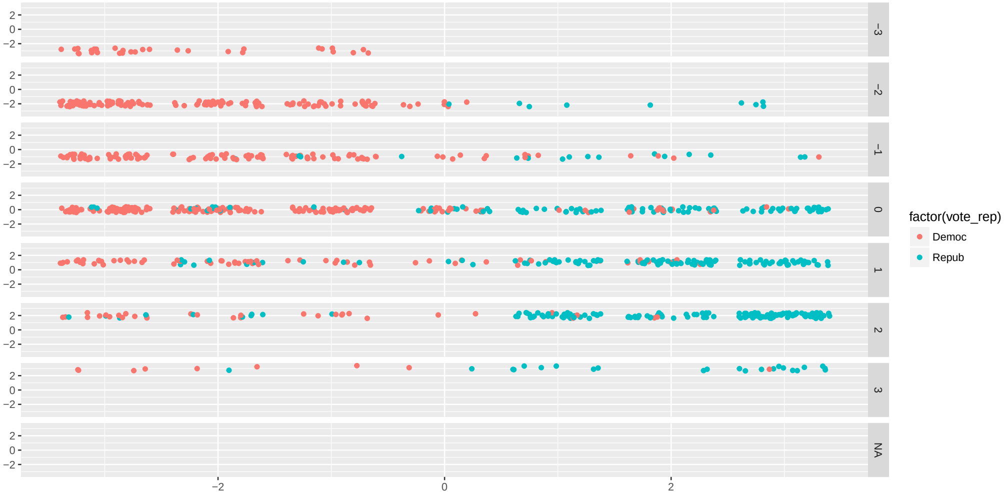


There doesn't seem to be a really strong interaction.

How about party identification and ideology?

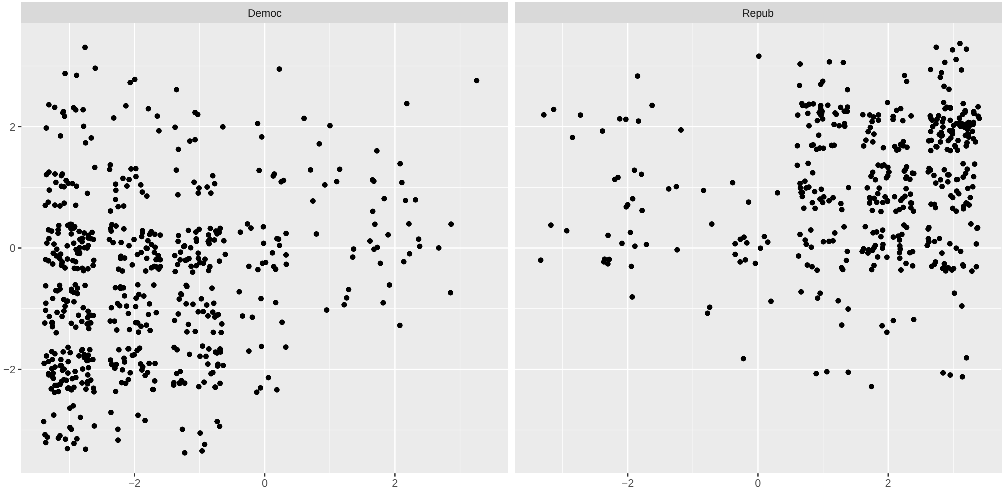
```
ggplot(nnss)+aes(x=partyid7_c,y=real_ideo_c,color=factor(vote_rep))+
  geom_jitter()+facet_grid(real_ideo_c~.)+
  scale_fill_manual(values=c("blue","red"))+ylab("")+xlab("")
```

```
## Warning: Removed 243 rows containing missing values (geom_point).
```



```
ggplot(nnss)+aes(x=partyid7_c,y=real_ideo_c)+
  geom_jitter()+facet_grid(.~vote_rep)+
  scale_fill_manual(values=c("blue","red"))+ylab("")+xlab("")
```

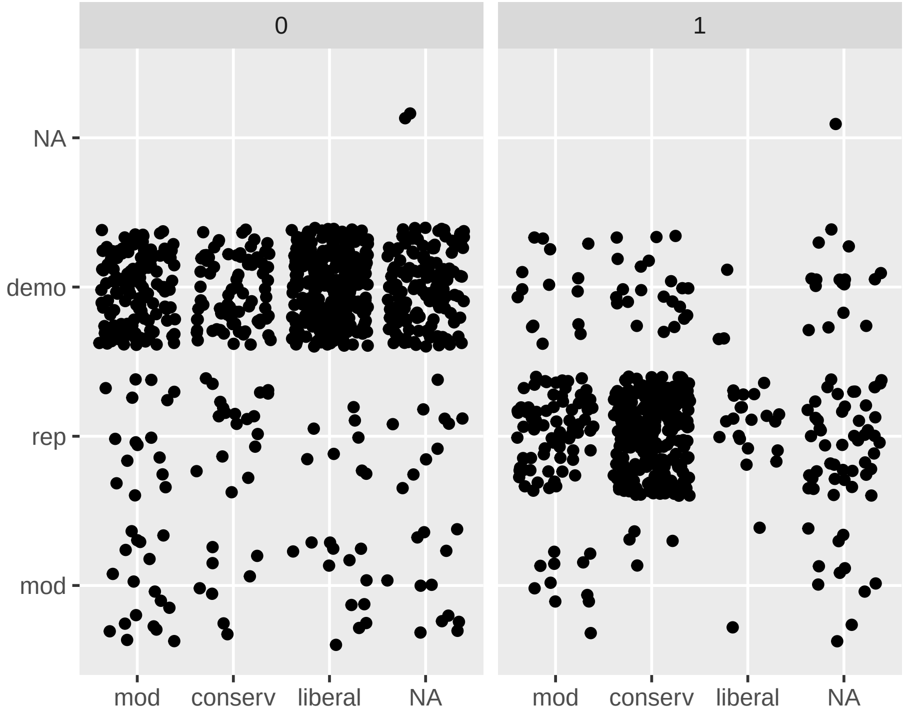
```
## Warning: Removed 243 rows containing missing values (geom_point).
```



```
nes5200_dt_s$party_ident<-factor(
  ifelse(nes5200_dt_s$partyid7_c >= 1, "rep",
  ifelse(nes5200_dt_s$partyid7_c <= -1,"demo", "mod") ),
  levels=c("mod","rep","demo"
))
nes5200_dt_s$ideo_ident<-factor(ifelse(nes5200_dt_s$real_ideo_c >= 1, "conserv",
  ifelse(nes5200_dt_s$real_ideo_c <= -1,"liberal", "mod")
),levels=c("mod","conserv","liberal"))

table(nes5200_dt_s$party_ident,nes5200_dt_s$ideo_ident)
```

```
##
##         mod conserv liberal
##   mod    29      12      15
##   rep    96     289      28
##   demo  157     101     252
```

```
ggplot(nes5200_dt_s)+aes(x=ideo_ident,y=party_ident)+
  geom_jitter()+facet_grid(.~vote_rep)+
  scale_fill_manual(values=c("blue","red"))+ylab("")+xlab("")
```

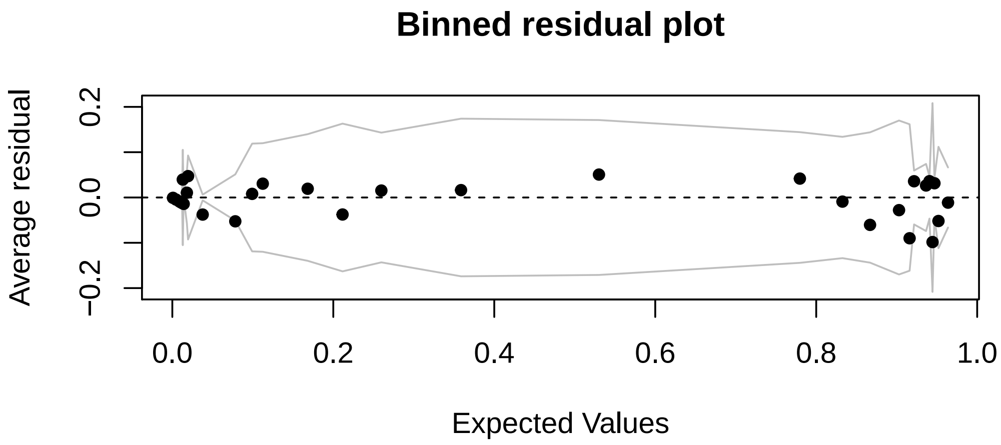


Ideally I would split the categories into 5 groups but when I do the coefficient estimates will blow up and there's nothing we can do about it with the skills you learned in class.

```
fit_vote_2<-glm(vote_rep ~ income_i+race+female+collage_grad+party_ident*ideo_ident,
                data=nes5200_dt_s,family=binomial)
display(fit_vote_2)
```

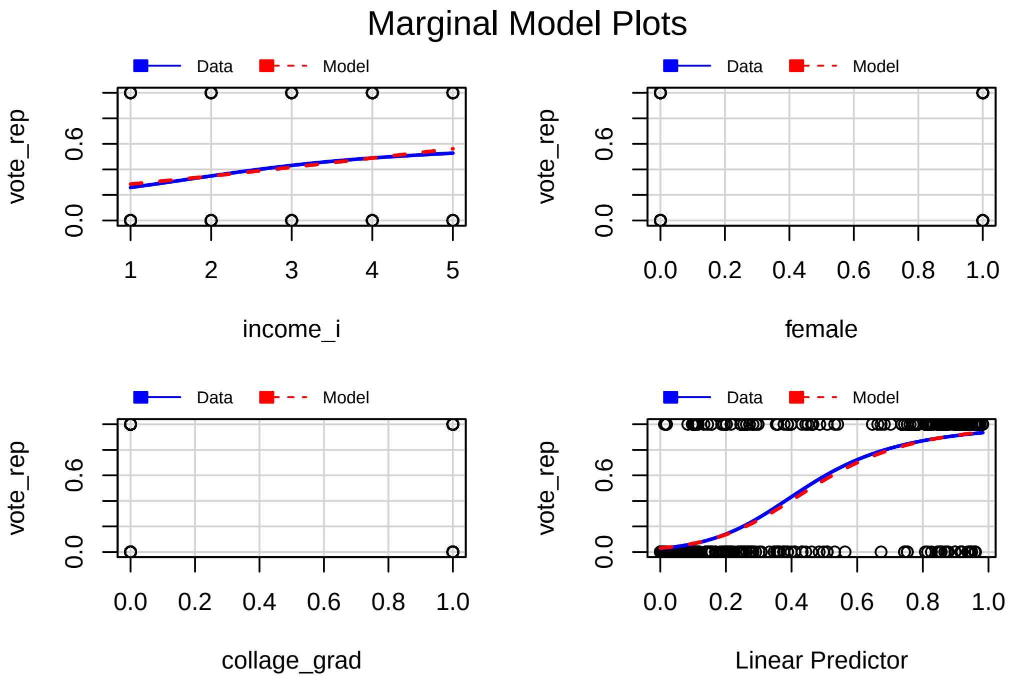
```
## glm(formula = vote_rep ~ income_i + race + female + collage_grad +
##     party_ident * ideo_ident, family = binomial, data = nes5200_dt_s)
##                                   coef.est coef.se
## (Intercept)                       -0.32     0.54
## income_i                          -0.05     0.12
## race2. black                      -2.57     0.61
## race3. asian                      -0.71     0.86
## race4. native american             0.09     0.75
## race5. hispanic                    0.76     0.53
## female                             0.30     0.25
## collage_gradTRUE                   0.49     0.28
## party_identrep                     1.94     0.50
## party_identdemo                   -2.02     0.51
## ideo_identconserv                 -0.44     0.75
## ideo_identliberal                 -1.63     0.89
## party_identrep:ideo_identconserv   1.40     0.84
## party_identdemo:ideo_identconserv  1.55     0.85
## party_identrep:ideo_identliberal   0.91     1.04
## party_identdemo:ideo_identliberal -0.62     1.10
## ---
##   n = 948, k = 16
##   residual deviance = 503.5, null deviance = 1296.3 (difference = 792.8)
```

```
binnedplot(fitted(fit_vote_2),resid(fit_vote_2,type="response"))
```



```
marginalModelPlots(fit_vote_2)
```

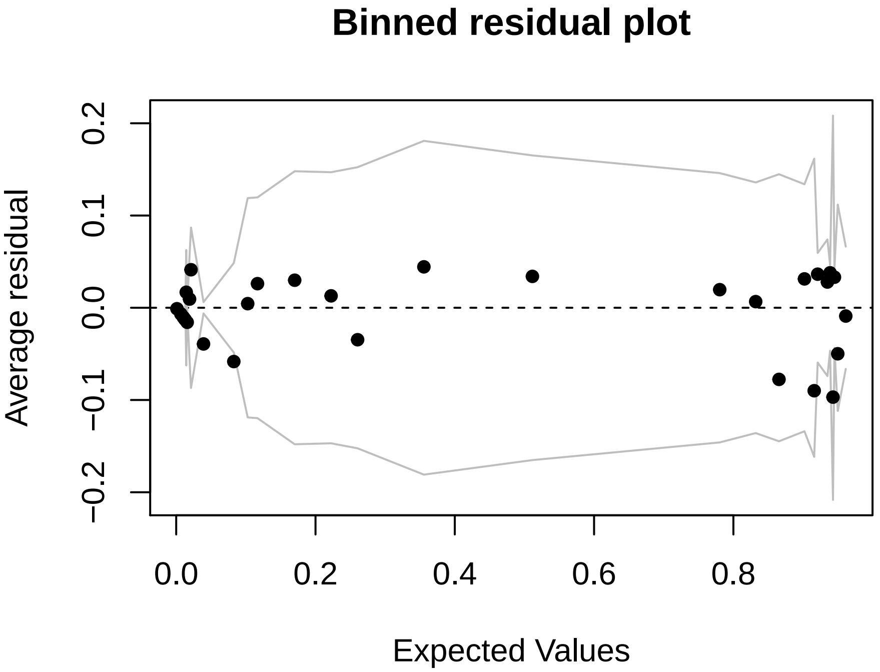
```
## Warning in mmps(...): Interactions and/or factors skipped
```



```
fit_vote_3<-bayesglm(vote_rep ~ income_i+race+female+collage_grad+party_ident*ideo_ident,
                data=nes5200_dt_s,family=binomial)
display(fit_vote_3)
```

```
## bayesglm(formula = vote_rep ~ income_i + race + female + collage_grad +
##     party_ident * ideo_ident, family = binomial, data = nes5200_dt_s)
##                                                 coef.est coef.se
## (Intercept)                                     -0.46     0.51
## income_i                                        -0.04     0.12
## race2. black                                    -2.42     0.58
## race3. asian                                    -0.58     0.79
## race4. native american                           0.07     0.69
## race5. hispanic                                  0.71     0.51
## race7. other                                  0.00     2.50
## race9. dk/na/no pre iw(1948,52)/sht-form new  0.00     2.50
## female                                        0.28     0.24
## collage_gradTRUE                              0.46     0.27
## party_identrep                                2.07     0.45
## party_identdemo                              -1.84     0.45
## ideo_identconserv                            -0.08     0.61
## ideo_identliberal                            -1.33     0.69
## party_identrep:ideo_identconserv              1.02     0.69
## party_identdemo:ideo_identconserv             1.13     0.69
## party_identrep:ideo_identliberal              0.59     0.82
## party_identdemo:ideo_identliberal            -0.87     0.86
## ---
## n = 948, k = 18
## residual deviance = 503.9, null deviance = 1296.3 (difference = 792.4)
```

```
binnedplot(fitted(fit_vote_3),resid(fit_vote_3,type="response"))
```



2. Evaluate and compare the different models you have fit. Consider coefficient estimates and standard errors, residual plots, and deviances.
3. For your chosen model, discuss and compare the importance of each input variable in the prediction.

Graphing logistic regressions:

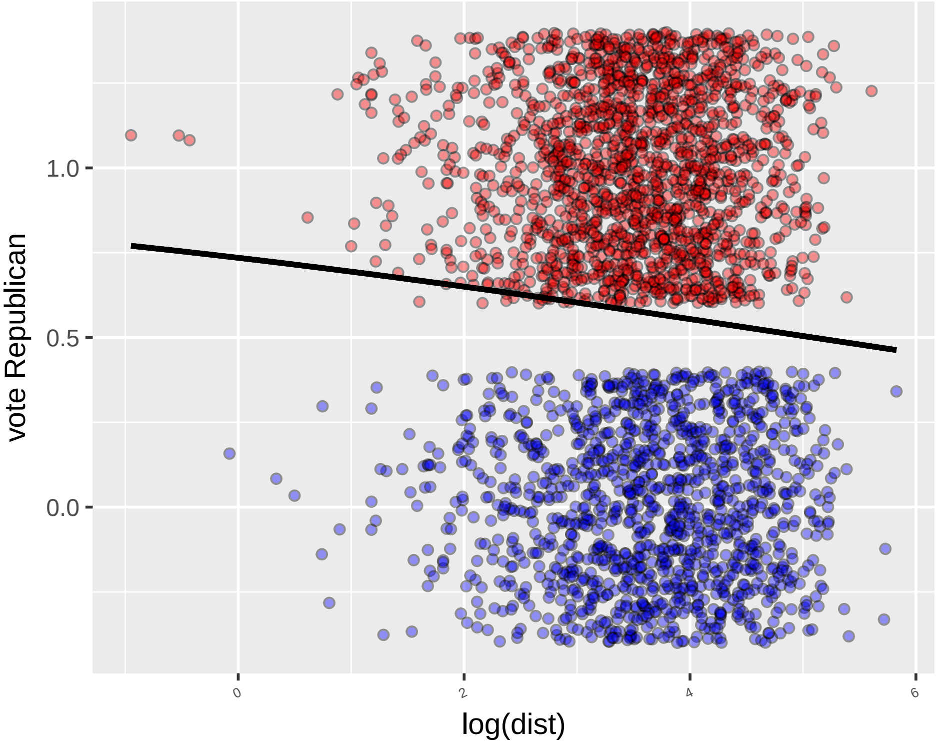
the well-switching data described in Section 5.4 of the Gelman and Hill are in the folder `arsenic`.

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

```
glm_wells<-glm(switch~log(dist),data=wells_dt,family=binomial)
```

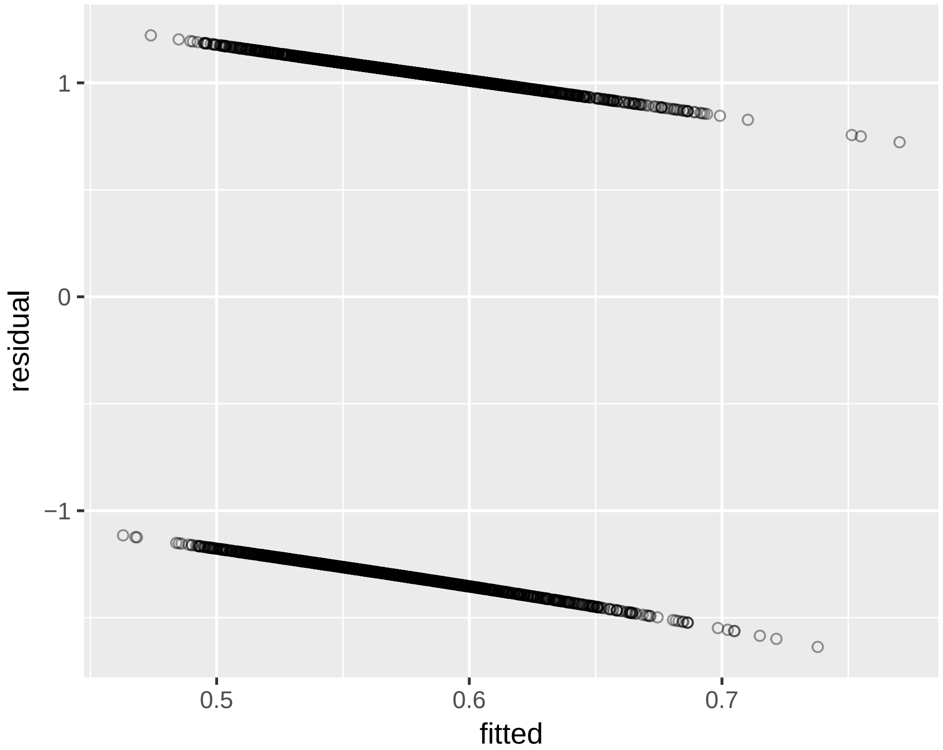
2. Make a graph similar to Figure 5.9 of the Gelman and Hill displaying Pr(switch) as a function of distance to nearest safe well, along with the data.

```
ggplot(wells_dt)+
  aes(x=log(dist),y=switch,fill=factor(switch)) + geom_jitter(shape=21,alpha=0.4)+
  theme(axis.text.x = element_text(angle = 25, hjust = 1,size=5))+
  theme(legend.position="none")+
  scale_fill_manual(values=c("blue","red"))+ylab("vote Republican")+
  stat_function(fun=function(x) invlogit(coef(glm_wells)[1]+coef(glm_wells)[2]*x),lwd=1)
```

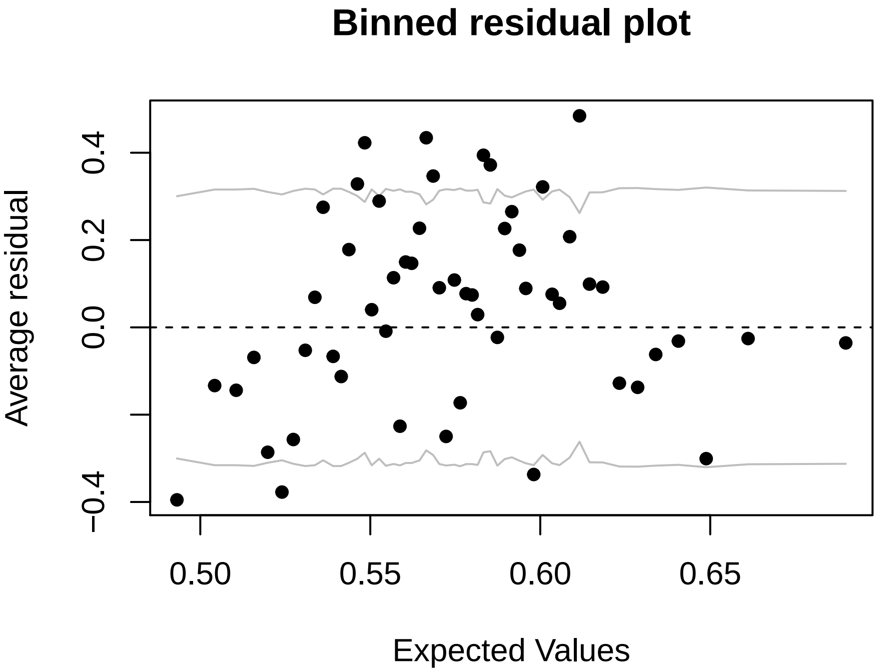


3. Make a residual plot and binned residual plot as in Figure 5.13.

```
invisible(wells_dt[,residual:=resid(glm_wells)])
invisible(wells_dt[,fitted:=fitted(glm_wells)])
ggplot(wells_dt)+
  aes(x=fitted,y=residual) + geom_jitter(shape=21,alpha=0.4)
```



```
binnedplot(wells_dt$fitted,wells_dt$residual)
```



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

```
n<- nrow(wells_dt)
wells_dt[(fitted>0.5&switch==0)|(fitted<0.5&switch==1),.N/n]
```

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

```
glm_wells_0<-glm(switch~1,data=wells_dt,family=binomial)
invisible(wells_dt[,residual0:=resid(glm_wells_0)])
invisible(wells_dt[,fitted0:=fitted(glm_wells_0)])
wells_dt[(fitted0>0.5&switch==0)|(fitted0<0.5&switch==1),.N/n]
```

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

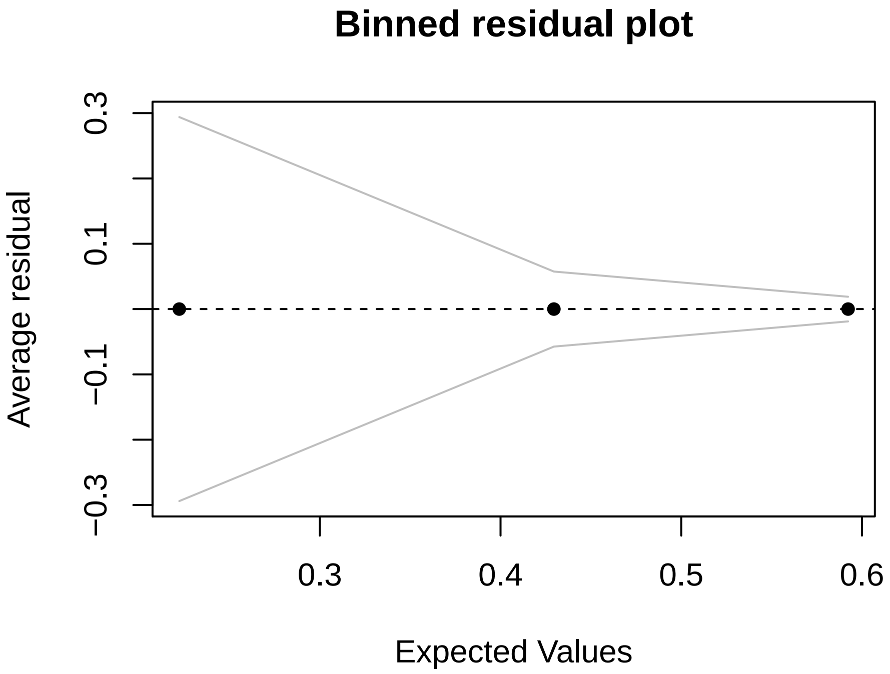
5. Create indicator variables corresponding to `dist < 100`, `100 =< dist < 200`, and `dist > 200`. Fit a logistic regression for Pr(switch) using these indicators. With this new model, repeat the computations and graphs for part (1) of this exercise.

```
invisible(wells_dt[,dist100:=dist<100])
invisible(wells_dt[,dist100200:=dist>=100&dist<200])
invisible(wells_dt[,dist200:=dist>200])
glm_wells_i<-glm(switch~dist100200+dist200,data=wells_dt,family=binomial)
display(glm_wells_i)
```

```
## glm(formula = switch ~ dist100200 + dist200, family = binomial,
##     data = wells_dt)
##                coef.est coef.se
## (Intercept)     0.37     0.04
## dist100200TRUE -0.66     0.12
## dist200TRUE    -1.63     0.80
## ---
##   n = 3020, k = 3
##   residual deviance = 4084.7, null deviance = 4118.1 (difference = 33.4)
```

The binned residual plot shows no trend

```
binnedplot(fitted(glm_wells_i),resid(glm_wells_i,type="response"))
```



The error rate is slightly lower.

```
n<- nrow(wells_dt)
invisible(wells_dt[,residual5:=resid(glm_wells_i,type="response")])
invisible(wells_dt[,fitted5:=fitted(glm_wells_i)])
wells_dt[(fitted5>0.5&switch==0)|(fitted5<0.5&switch==1),.N/n]
```

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

Model building and comparison:

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

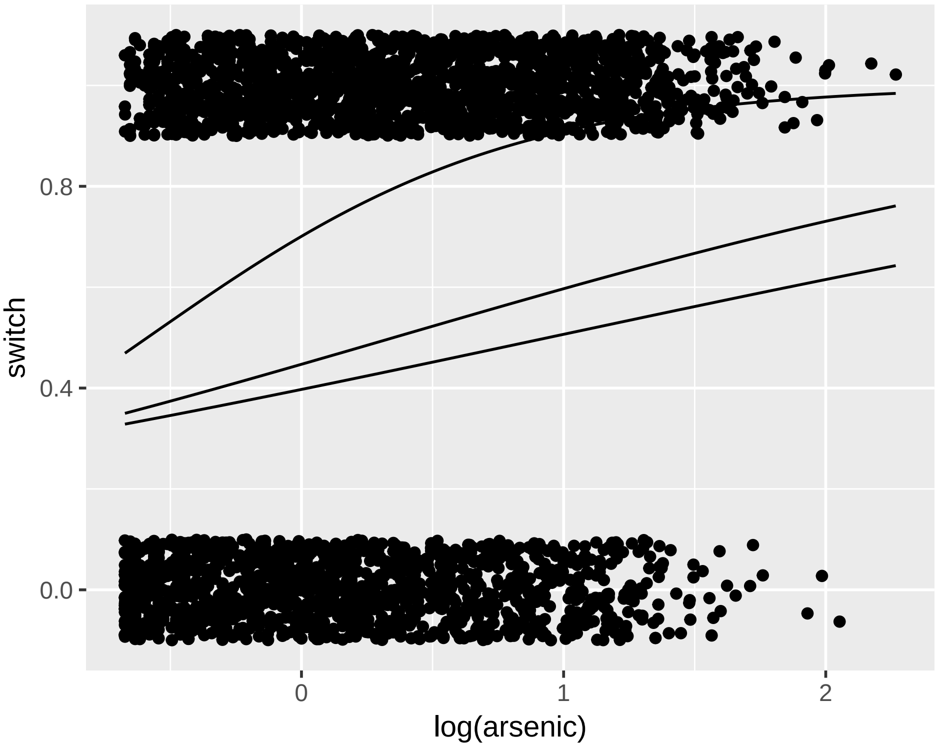
1. Fit a logistic regression for the probability of switching using, as predictors, distance, `log(arsenic)`, and their interaction. Interpret the estimated coefficients and their standard errors.

```
glm_wells_i<-glm(switch~log(dist)*log(arsenic),data=wells_dt,family=binomial)
display(glm_wells_i)
```

```
## glm(formula = switch ~ log(dist) * log(arsenic), family = binomial,
##     data = wells_dt)
##                        coef.est coef.se
## (Intercept)             1.15     0.18
## log(dist)              -0.29     0.05
## log(arsenic)            1.68     0.30
## log(dist):log(arsenic) -0.23     0.08
## ---
##   n = 3020, k = 4
##   residual deviance = 3924.8, null deviance = 4118.1 (difference = 193.3)
```

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

```
func<- function(x,cp){
  coeffs<-as.vector(coef(glm_wells_i))
  invlogit(coeffs[1]+coeffs[2]*cp+coeffs[3]*x+coeffs[4]*cp*x)
}
ggplot(wells_dt)+geom_jitter(height=0.1)+aes(x=log(arsenic),y=switch)+
  stat_function(fun=func,args = list(cp = log(100)))+
  stat_function(fun=func,args = list(cp = log(200)))+
  stat_function(fun=func,args = list(cp = 1))
```



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.

```
b <- coef(glm_wells_i)

 # for distance to nearest safe well

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

```
## [1] -0.2593998
```

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

```
## [1] -0.06219585
```

```
hi <- log(1)
lo <- log(0.5)
delta <- invlogit (b[1] + b[2]*log(wells_dt$dist) + b[3]*hi +b[4]*log(wells_dt$dist)*hi ) -
         invlogit (b[1] + b[2]*log(wells_dt$dist) + b[3]*lo+ b[4]*log(wells_dt$dist)*lo )
```

```
print (mean(delta))
```

```
## [1] 0.1436887
```

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

```
## [1] 0.1357249
```

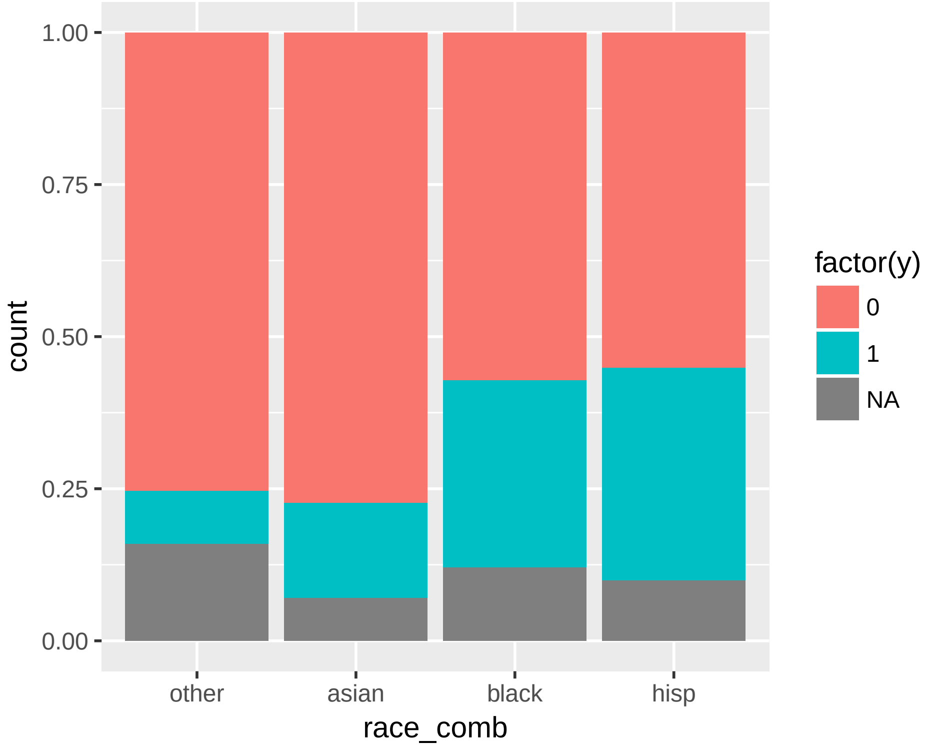
Building a logistic regression model:

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.

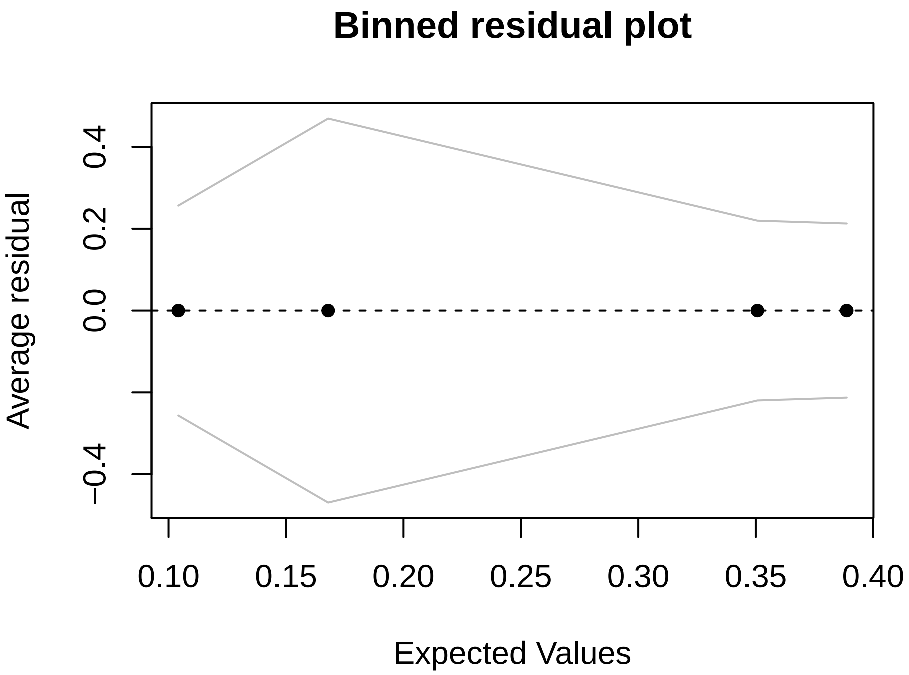
```
apt_dt$race_comb<- "other"
apt_dt$race_comb[apt_dt$asian]<-"asian"
apt_dt$race_comb[apt_dt$black]<-"black"
apt_dt$race_comb[apt_dt$hisp]<-"hisp"
apt_dt$race_comb<-factor(apt_dt$race_comb,levels=c("other","asian","black","hisp"))

ggplot(apt_dt)+
  geom_bar(position = "fill")+
  aes(x=race_comb,fill=factor(y))
```



```r
fitapt<-glm(y~asian+black+ hisp,family=binomial,data=apt_dt)
display(fitapt)
```

```
## glm(formula = y ~ asian + black + hisp, family = binomial, data = apt_dt)
##             coef.est coef.se
## (Intercept) -2.15     0.13
## asianTRUE    0.55     0.27
## blackTRUE    1.54     0.17
## hispTRUE     1.70     0.17
## ---
##   n = 1522, k = 4
##   residual deviance = 1526.3, null deviance = 1672.2 (difference = 145.9)
```

```r
binnedplot(fitapt$fitted,fitapt$residuals)
```

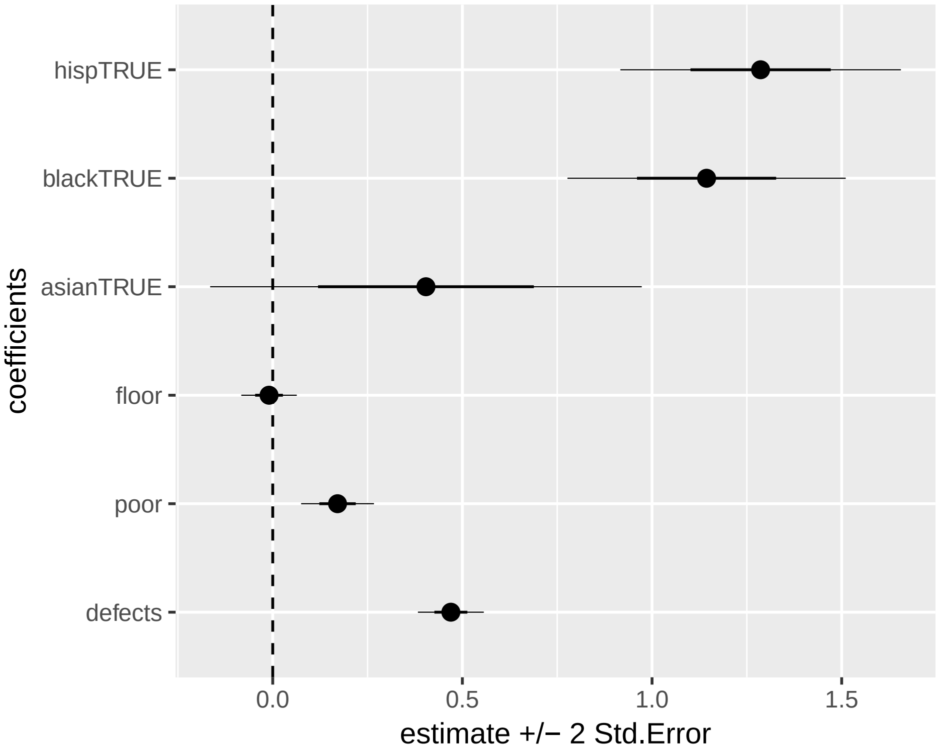


2. Add to your model some other potentially relevant predictors describing the apartment, building, and community district. Build your model using the general principles explained in Section 4.6 of the Gelman and Hill. Discuss the coefficients for the ethnicity indicators in your model.

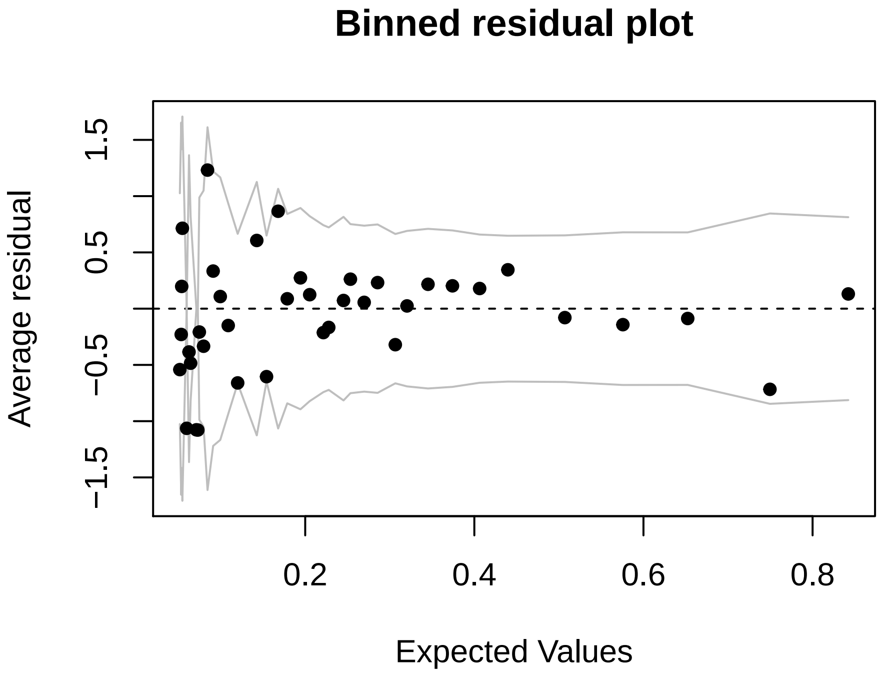
```r
fitapt2<-glm(y~defects+poor+floor+asian+black+ hisp,family=binomial,data=apt_dt)
display(fitapt2)
```

```
## glm(formula = y ~ defects + poor + floor + asian + black + hisp,
##     family = binomial, data = apt_dt)
##             coef.est coef.se
## (Intercept) -3.02     0.22
## defects      0.47     0.04
## poor         0.17     0.05
## floor       -0.01     0.04
## asianTRUE    0.40     0.28
## blackTRUE    1.14     0.18
## hispTRUE     1.29     0.18
## ---
##   n = 1522, k = 7
##   residual deviance = 1349.5, null deviance = 1672.2 (difference = 322.7)
```

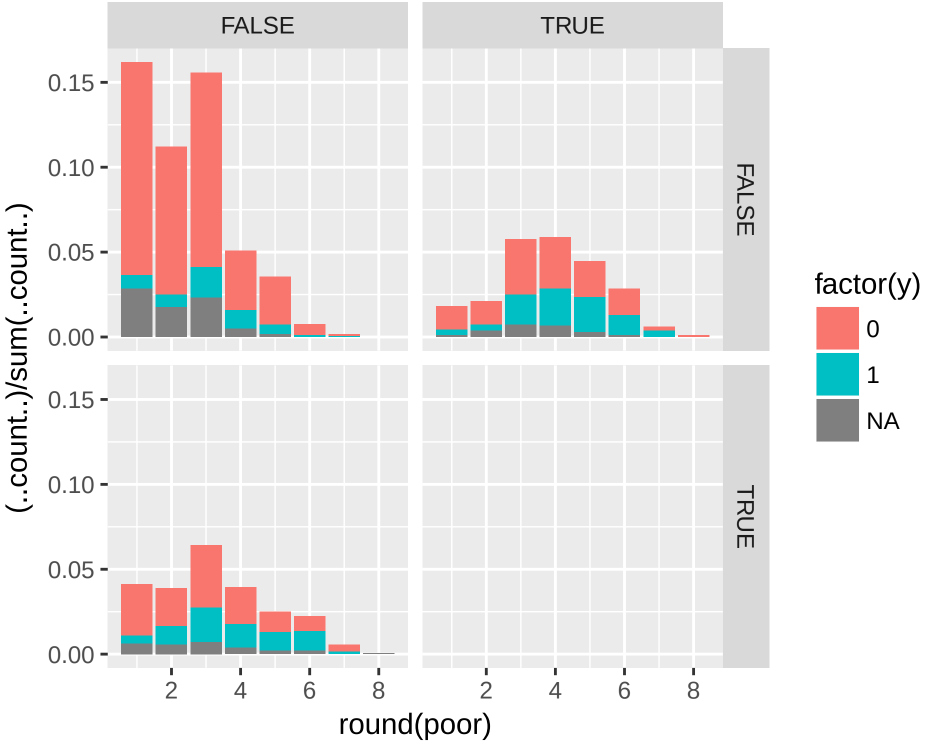
```
coefplot_my(fitapt2)
```



```
binnedplot(fitapt2$fitted,fitapt2$residuals)
```



```
stat = "identity"
ggplot(apt_dt )+geom_bar()+
  aes(x=round(poor),y=(..count..)/sum(..count..), fill=factor(y))+facet_grid(black~hisp)
```



Taking into account the other information, we still see there is difference in the ethnicity.

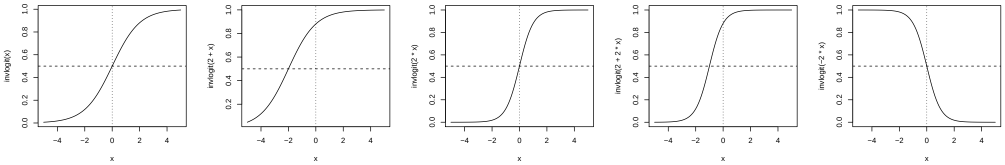
Conceptual exercises.

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. $Pr(y=1) = logit^{-1}(-2x)$

```
par(mfrow=c(1,5))
curve(invlogit(x),from=-5,to=5);abline(v=0,lty=3);abline(h=0.5,lty=2)
curve(invlogit(2+x),from=-5,to=5);abline(v=0,lty=3);abline(h=0.5,lty=2)
curve(invlogit(2*x),from=-5,to=5);abline(v=0,lty=3);abline(h=0.5,lty=2)
curve(invlogit(2+2*x),from=-5,to=5);abline(v=0,lty=3);abline(h=0.5,lty=2)
curve(invlogit(-2*x),from=-5,to=5);abline(v=0,lty=3);abline(h=0.5,lty=2)
```

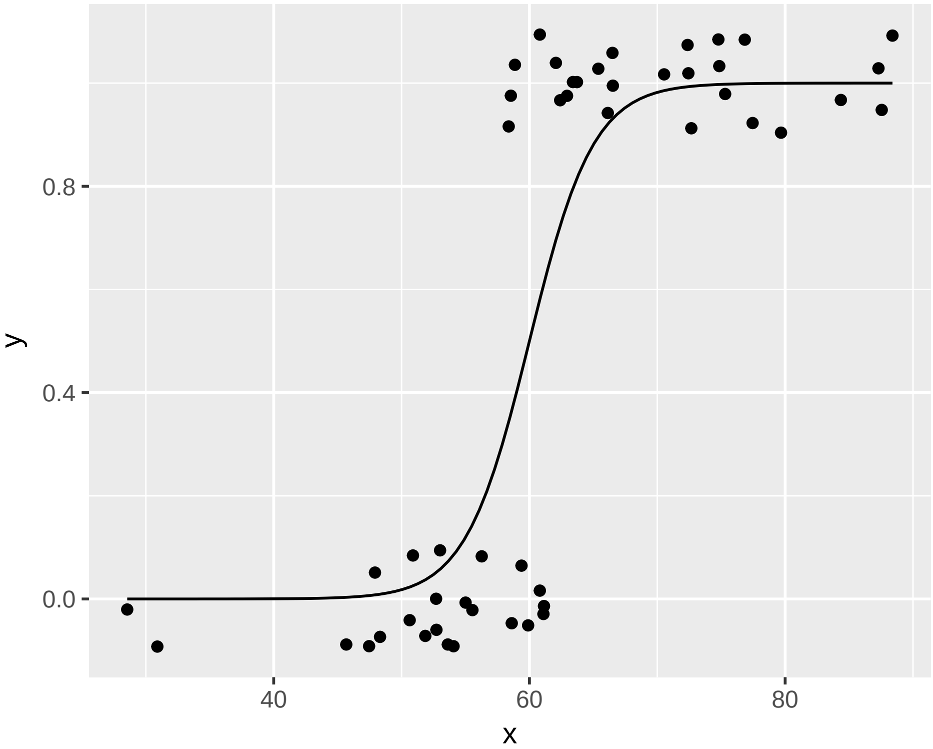


Hypothetical model

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 $Pr(pass) = logit^{-1}(-24 + 0.4x)$.

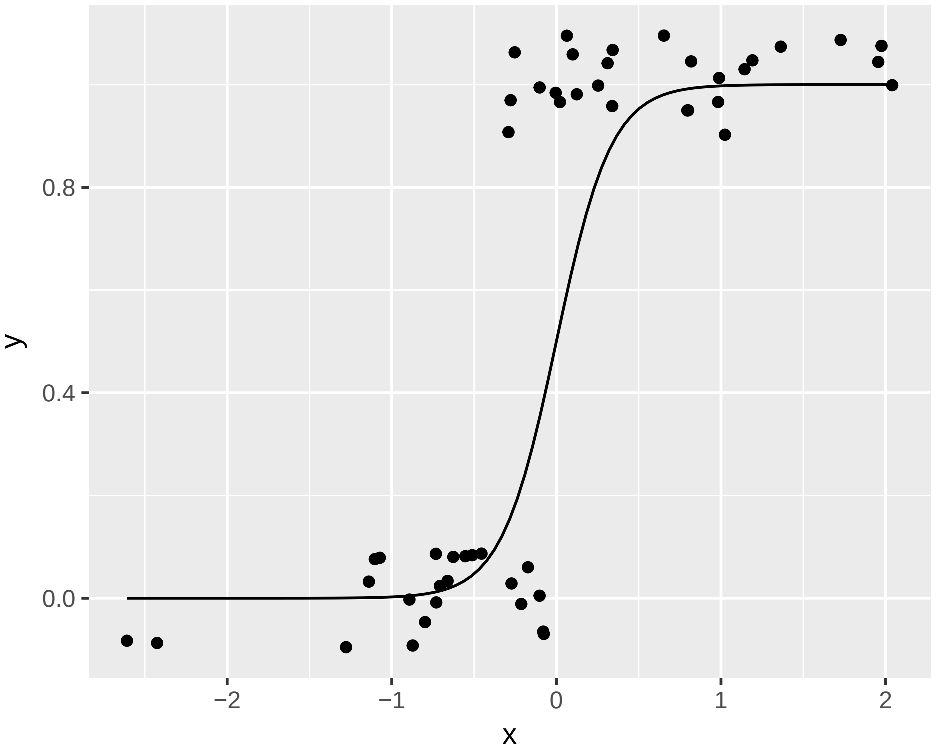
1. Graph the fitted model. Also on this graph put a scatterplot of hypothetical data consistent with the information given.

```
x=rnorm(50,60,15)
y=rbinom(50,1,invlogit(-24 + 0.4*x))
ggplot(data.frame(x=x,y=y))+
  geom_jitter(height=0.1)+ aes(x=x,y=y)+
  stat_function(fun=function(x) invlogit(-24 + 0.4*x))
```



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?

```
x<- scale(x,center=TRUE)
ggplot(data.frame(x,y))+ aes(x=x,y=y)+geom_jitter(height=0.1) + stat_function(fun=function(x) invlogit(
```



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?

We expect the deviance to decrease by 1. However, when we use the simulated example the decrease was even less.

```
newpred <- rnorm (50,0,1)
deviance(glm(y~x,family="binomial"))
```

```
## [1] 20.07077
```

```
deviance(glm(y~x+newpred,family="binomial"))
```

```
## [1] 20.0574
```

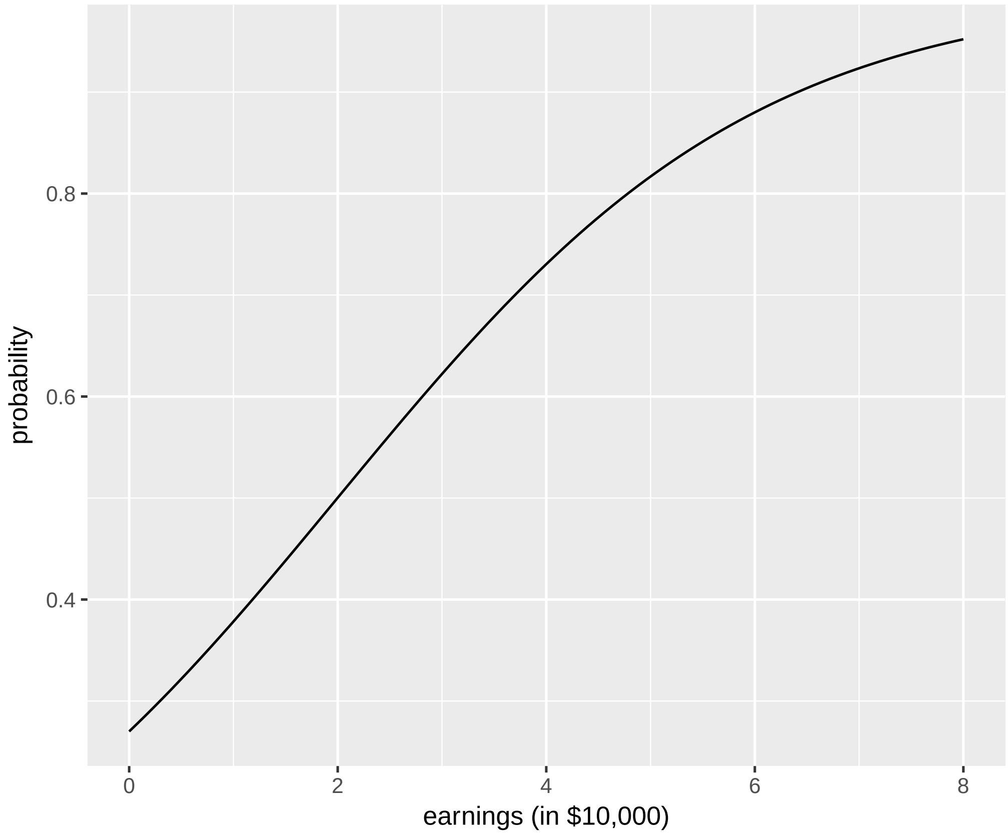
Logistic regression

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).

We can find the y-intercept solving the equation $logit(0.27) = -0.9946$. On the same way, we can find the coefficient for the earnings:

$$logit(0.88) = logit(0.27) + x6 \rightarrow x = (logit(0.88) - logit(0.27))/6 = 0.4978421$$

```
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")
```



This lead us to the final equation:

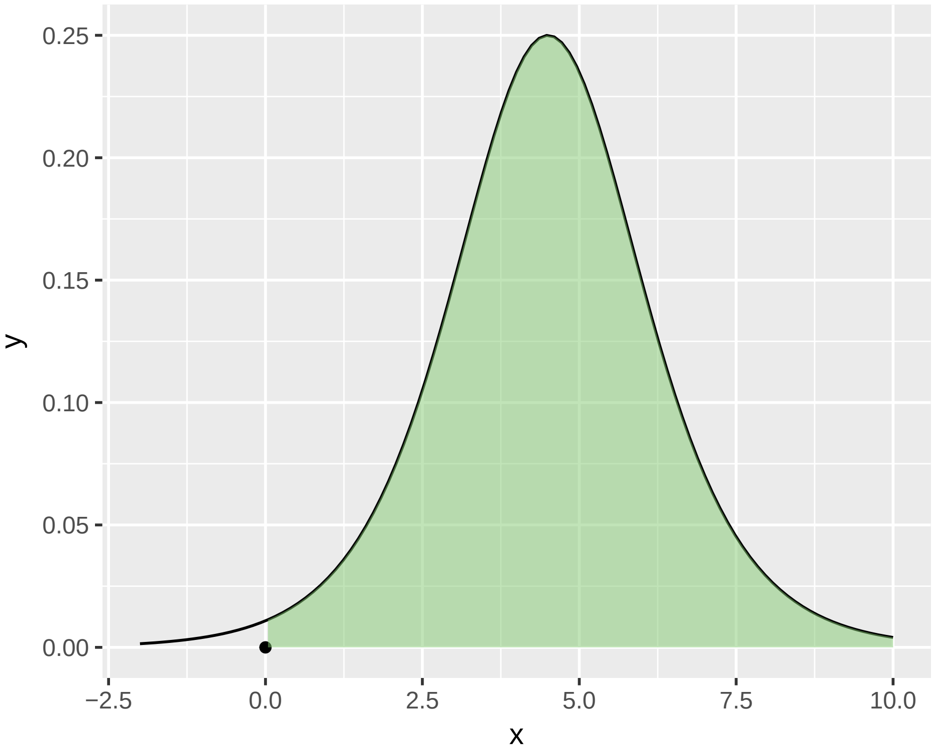
$$Pr(y = 1) = logit^{-1}(-0.99 + 0.5x)$$

.

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.

```
funcShaded <- function(x) {
    y <- dlogis(x, location=1+2*1+3*0.5, scale = 1)
    y[x < 0 | x > (100)] <- NA
    return(y)
}
ggplot(data.frame(x=0,y=0))+xlim(-2,10)+
  stat_function(fun=dlogis,args=list(location=1+2*1+3*0.5))+
  geom_point()+aes(x=x,y=y)+
  stat_function(fun=funcShaded, geom="area", fill="#84CA72", alpha=0.5)
```



Limitations of logistic regression:

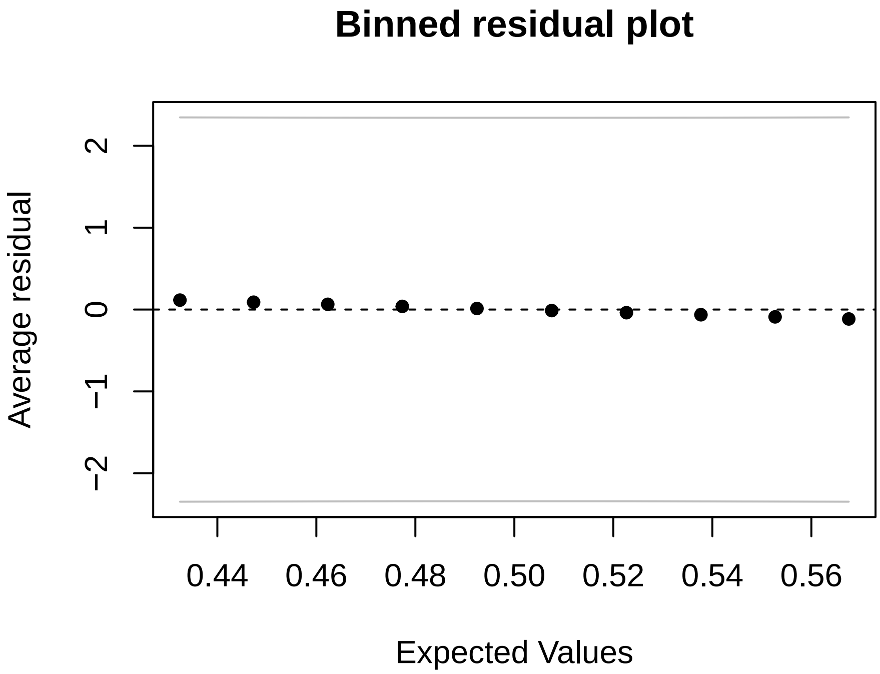
consider a dataset with $n = 20$ points, a single predictor x that takes on the values $1, \ldots, 20$, and binary data y. Construct data values $y_1, \ldots, y_{20}$ that are inconsistent with any logistic regression on x. Fit a logistic regression to these data, plot the data and fitted curve, and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

We create oscillating y that is repeating 0 and 1.

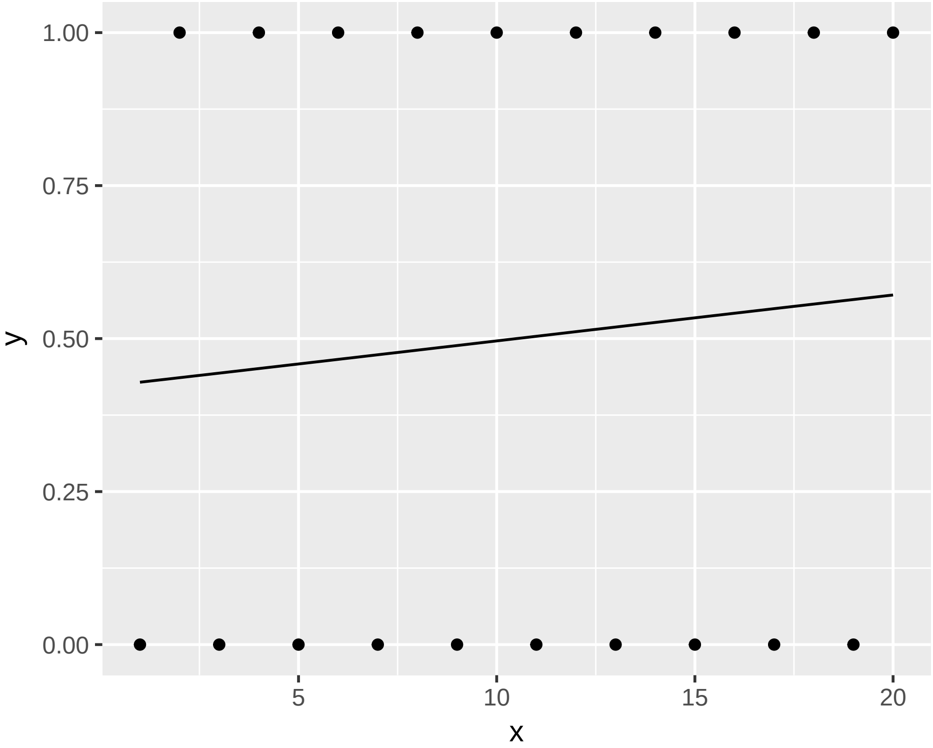
```r
set.seed(7654321)
y <- rep(c(0,1),10)
x <- 1:20
m1 <- glm(y ~ x, family=binomial(link="logit"))
display(m1)
```

```
## glm(formula = y ~ x, family = binomial(link = "logit"))
##             coef.est coef.se
## (Intercept) -0.32     0.93
## x            0.03     0.08
## ---
##   n = 20, k = 2
##   residual deviance = 27.6, null deviance = 27.7 (difference = 0.2)
```

```r
binnedplot(fitted(m1),resid(m1))
```



```
fun<- function(x){invlogit(coef(m1)[1]+coef(m1)[2]*x)}
ggplot(data.frame(y, x)) + geom_point() +aes(x=x, y=y)+stat_function(fun=fun)
```



The logistic regression predicts positive relationship.

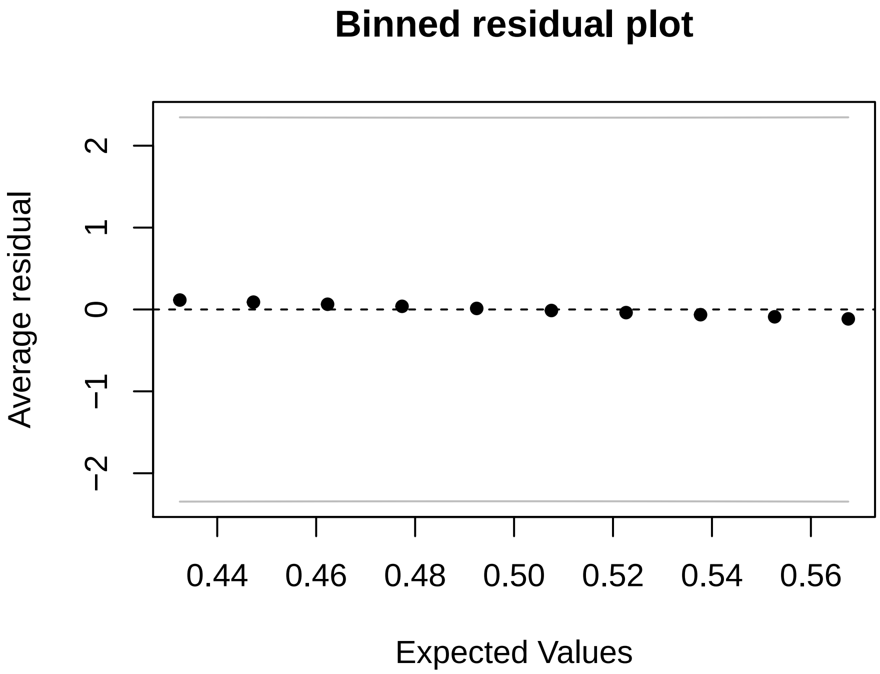
If we switch the order we get a negative relationship.

```
set.seed(7654321)
y <- rep(c(1,0),10)
x <- 1:20
```

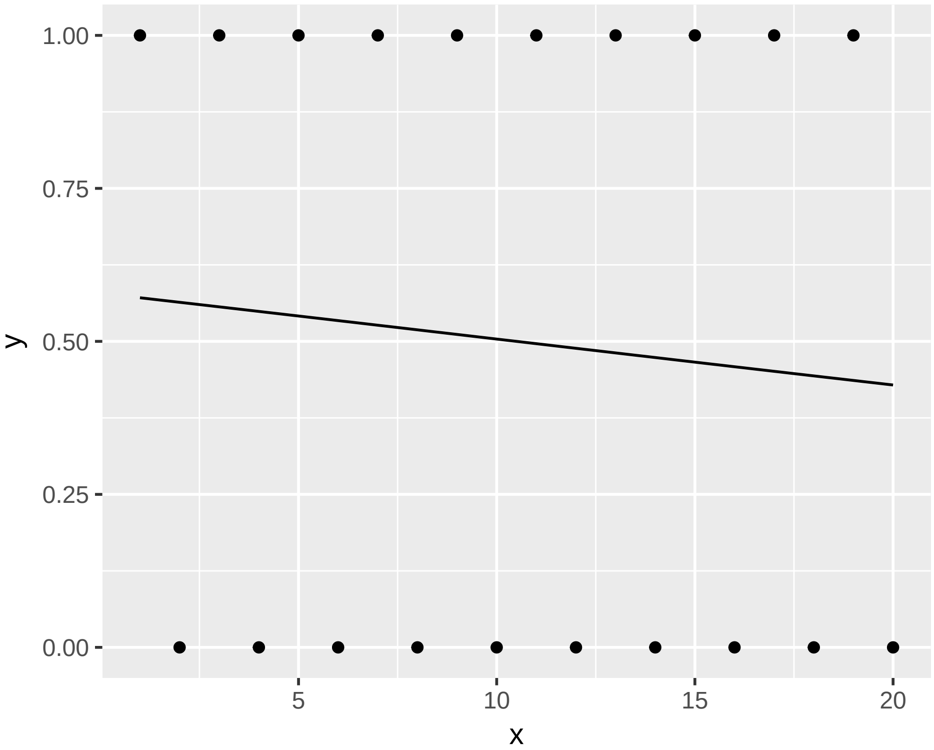
```
m2 <- glm(y ~ x, family=binomial(link="logit"))
display(m2)
```

```
## glm(formula = y ~ x, family = binomial(link = "logit"))
##             coef.est coef.se
## (Intercept)  0.32     0.93
## x           -0.03     0.08
## ---
##   n = 20, k = 2
##   residual deviance = 27.6, null deviance = 27.7 (difference = 0.2)
```

```
binnedplot(fitted(m2),resid(m2))
```



```
fun<- function(x){invlogit(coef(m2)[1]+coef(m2)[2]*x)}
ggplot(data.frame(y, x)) + geom_point() +aes(x=x, y=y)+stat_function(fun=fun)
```



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)
```

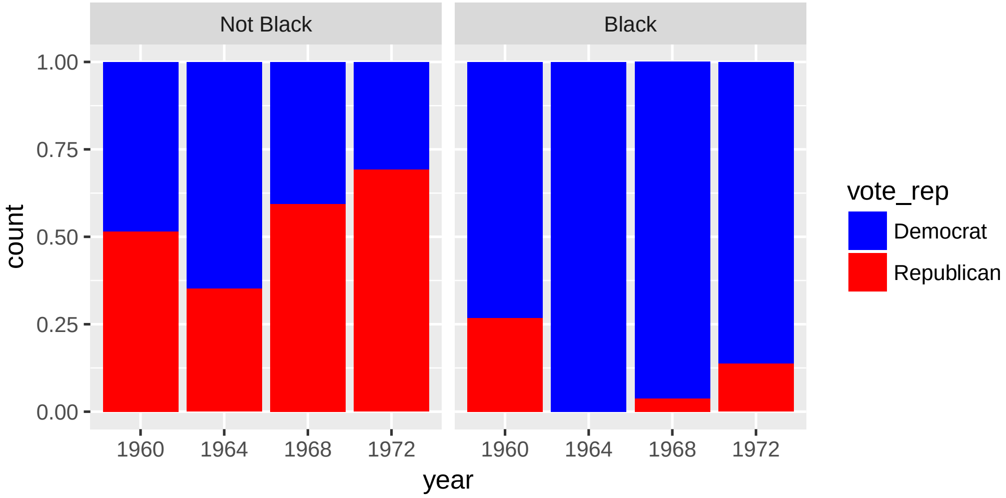
```
## glm(formula = vote_rep ~ female + black + income, family = binomial(link = "logit"),
##     data = nes5200_dt_d, subset = (year == 1964))
##             coef.est coef.se
## (Intercept)  -1.16     0.22
## female       -0.08     0.14
## black       -16.83   420.51
## income        0.19     0.06
## ---
##   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)
```

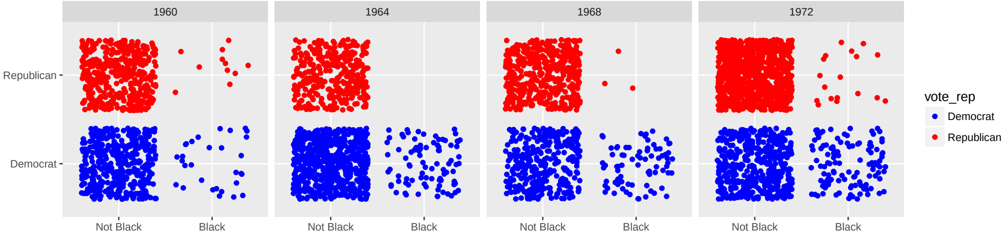
What happened with the coefficient of black in 1964? Take a look at the data and figure out where this extreme estimate came from. What can be done to fit the model in 1964?

```
ns<-subset(nes5200_dt_d,year%in%c(1960,1964,1968,1972)&!is.na(black))
ns$year<- factor(ns$year)
ns$vote_rep<- factor(ns$vote_rep,levels=c(0,1),labels=c("Democrat","Republican"))
ns$black<- factor(ns$black,levels=c(0,1),labels=c("Not Black","Black"))
ggplot(ns)+aes(x=year,fill=vote_rep)+
  geom_bar(position = "fill")+
  facet_grid(.~black)+
  scale_fill_manual(values=c("blue","red"))
```



In 1964 there was no black Republican vote.

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



We can do a subset analysis without considering the black population. Another thing we can do is to put a prior information in the estimate to regularize the coefficient estimates. We will get back to this point later in the semester.

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.