Logistics Regression

September 27, 2017

This writing explores logistic regression with the National Election Study data from Gelman & Hill (GH). (See Chapter 4.7 for descriptions of some of the variables and 5.1 of GH for initial model fitting).

[*The following code will read in the data and perform some filtering/recoding. Remove this text and modify the code chunk options so that the code does not appear in the output.*]

1. Summarize the data for 1992 noting which variables have missing data. Which variables are categorical but are coded as numerically?

```
##pick coloumns with NA
df_1=as.data.frame.matrix(summary(nes1992))
condition_1 <- !is.na(unlist(df_1[7,]))
df_with_missing_data=df_1[,condition_1]
print(colnames(df_with_missing_data))
```

```
##  [1] "    occup1"      "    union"       "   religion"
##  [4] "martial_status"  "    occup2"      "  icpsr_cty"
##  [7] "   partyid7"     "   partyid3"     "  partyid3_b"
## [10] " str_partyid"    " father_party"   " mother_party"
## [13] "  dem_therm"     "  rep_therm"     "    regis"
## [16] "presvote_intent" "  ideo_feel"     "    ideo7"
## [19] "     ideo"       "      cd"        "rep_pres_intent"
## [22] "  real_ideo"     "  presapprov"    "   perfin1"
## [25] "   perfin2"      "    perfin"      "   newfathe"
## [28] "   newmoth"      " parent_party"
```

29 variables have missing data, `including occup1`, `union`, `religion`, `martial_status`, `occup2`, `icpsr_cty`, `partyid7`, `partyid3`, `partyid3_b`, `str_partyid`, `father_party`, `mother_party`, `dem_therm`, `rep_therm`, `regis`, `presvote_intent`, `ideo_feel`, `ideo7`, `ideo`, `cd`, `rep_pres_intent`, `real_ideo`, `presapprov`, `perfin1`, `perfin2`, `perfin3`, `newfathe`, `newmoth`, `parent_party`.

variable `gender`, `race`, `educ1`, `urban`, `region`, `income`, `occup1`, `union`, `religion`, `educ2`, `educ3`, `martial_status`, `occup2`, `partyid7`, `partyid3`, `partyid3_b`, `str_partyid`, `father_party`, `mother_party`, `dlikes`, `rlikes`, `presvote`, `presvote_2party`, `presvote_intent`, `ideo7`, `ideo`, `cd`, `state`, `inter_pre`, `inter_post`, `female`, `rep_presvote`, `rep_pres_intent`, `south`, `real_ideo`, `presapprov`, `perfin1`, `perfin2`, `presadm`, `newfathe`, `newmoth`, `parent_party`, `white` are categorical but are coded as numerically.

2. Fit the logistic regression to estimate the probability that an individual would vote Bush (Republican) as a function of income and provide a summary of the model.

```
# income is continuous number
vote = factor(nes1992$vote)
glm.fit=glm(vote ~ income, data = nes1992, family = binomial(link = "logit"))
summary(glm.fit)
```

```
##
## Call:
## glm(formula = vote ~ income, family = binomial(link = "logit"),
##     data = nes1992)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2756  -1.0034  -0.8796   1.2194   1.6550
```

```
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.40213    0.18946  -7.401 1.35e-13 ***
## income       0.32599    0.05688   5.731 9.97e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1591.2  on 1178  degrees of freedom
## Residual deviance: 1556.9  on 1177  degrees of freedom
## AIC: 1560.9
##
## Number of Fisher Scoring iterations: 4
```

The coefficient of Intercept is -1.4, and the coefficient of income is 0.32599, which means an unit change in income will result in 0.32599 increasing in $logP(vote)$. Both of these coeffcients are statistically significant, and the residual deviance of the model is 1556.9.

```
#income as factor
glm.fit1=glm(vote ~ factor(income), data = nes1992, family = binomial(link = "logit"))
summary(glm.fit1)
```

```
##
## Call:
## glm(formula = vote ~ factor(income), family = binomial(link = "logit"),
##     data = nes1992)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2005  -1.0000  -0.9005   1.2027   1.7034
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -1.1838     0.2087  -5.673 1.40e-08 ***
## factor(income)2   0.4906     0.2555   1.920  0.05482 .
## factor(income)3   0.7509     0.2345   3.202  0.00136 **
## factor(income)4   1.1243     0.2312   4.863 1.15e-06 ***
## factor(income)5   1.2378     0.3125   3.962 7.45e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1591.2  on 1178  degrees of freedom
## Residual deviance: 1555.8  on 1174  degrees of freedom
## AIC: 1565.8
##
## Number of Fisher Scoring iterations: 4
```

3. Obtain a point estimate and create a 95% confidence interval for the odds ratio for voting Republican for a rich person (income category 5) compared to a poor person (income category 1). *Hint this is more than a one unit change; calculate manually and then show how to modify the output from confint.* Provide a sentence interpreting the result.

```r
odds_ratio = exp(4*glm.fit$coefficients[2])
ciodds_ratio = exp(4*confint(glm.fit)[2,])
odds_ratio
```

```
##   income
## 3.683925
```

```r
ciodds_ratio
```

```
##    2.5 %   97.5 %
## 2.367648 5.779917
```

The point estimator for odds ratio is `odds_ratio`, and the CI for it is `ciodds_ratio`.

```r
#income as continuous
beta1 = summary(glm.fit)[["coefficients"]][,1][2]
beta1_se = summary(glm.fit)[["coefficients"]][,2][2]
critval = qnorm(0.975)
#point estimate
point_estimate = exp(4*beta1)
odds_ratio_CI = matrix(c(exp(4*(beta1 - critval * beta1_se)),
                  exp(4*(beta1 + critval * beta1_se))), nrow = 1)
dimnames(odds_ratio_CI)=list(c("Confidence Interval(hand)"),
                             c("2.5%", "97.5%"))
odds_ratio_CI_confint = suppressMessages(t(as.matrix(exp(4*confint(glm.fit)[2,]))))
dimnames(odds_ratio_CI_confint)=list(c("Confidence Interval(confint)"),c("2.5%", "97.5%"))
odds_ratio = rbind(odds_ratio_CI, odds_ratio_CI_confint)
odds_ratio
```

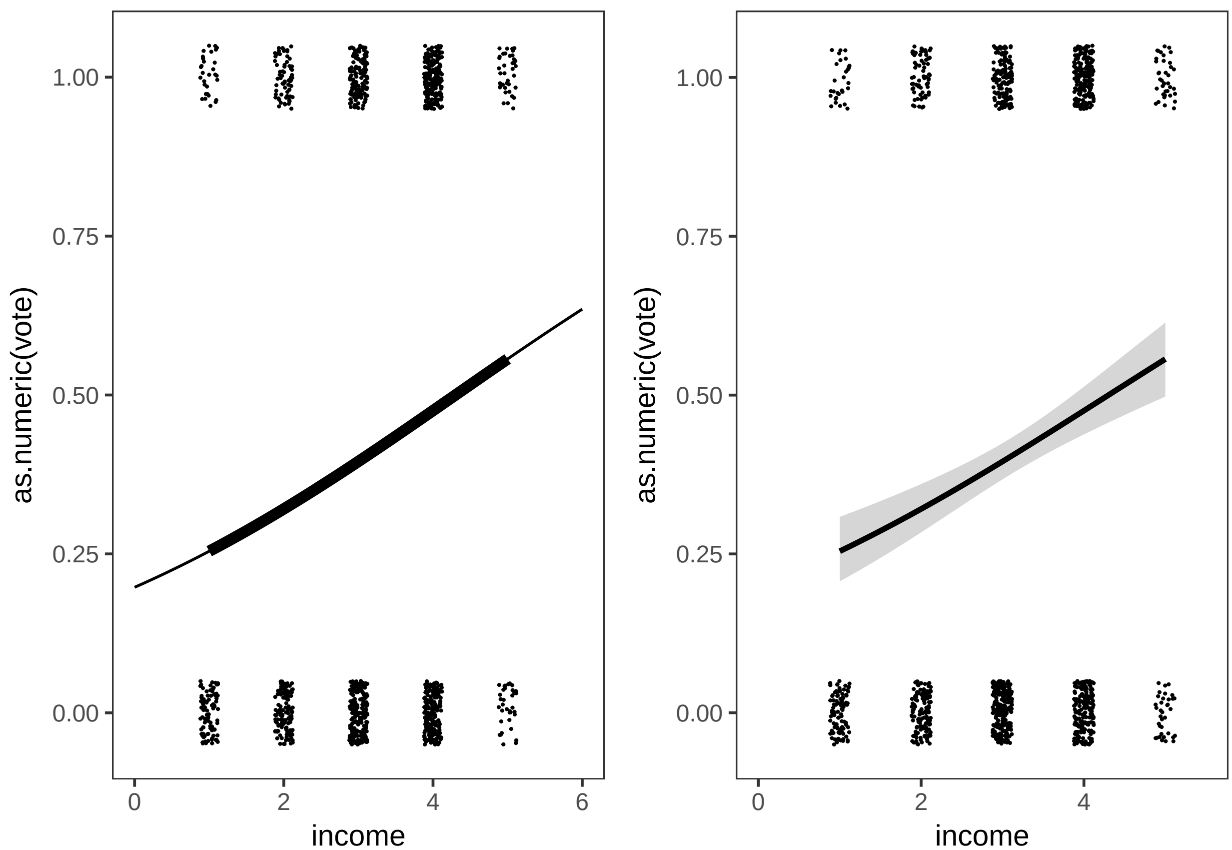
```
##                                  2.5%    97.5%
## Confidence Interval(hand)    2.358539 5.754114
## Confidence Interval(confint) 2.367648 5.779917
```

4. Obtain fitted probabilities and 95% confidence intervals for the income categories using the `predict` function. Use `ggplot` to recreate the plots in figure 5.1 of Gelman & Hill. *write a general function?*

```r
fitted_CI = as.data.frame(matrix(NA, nrow = 5, ncol = 3))
colnames(fitted_CI) = c("fitted probability", "0.025 lower bound", "0.975 upper bound")
for(i in 1:5){
  predict = predict(glm.fit, data.frame(income= i), type="response", se.fit=TRUE)
  fitted_CI[i,1] = predict$fit
  se.fit = predict$se.fit
  fitted_CI[i,2] = fitted_CI[i,1] - qnorm(0.975)*se.fit
  fitted_CI[i,3] = fitted_CI[i,1] + qnorm(0.975)*se.fit
}
fitted_CI = cbind(c(1:5),fitted_CI)
names(fitted_CI)[1] = 'income'
kable(fitted_CI)
```

income	fitted probability	0.025 lower bound	0.975 upper bound
1	0.2542381	0.2034288	0.3050474
2	0.3207907	0.2826795	0.3589019
3	0.3955251	0.3670002	0.4240501
4	0.4754819	0.4378802	0.5130836
5	0.5567158	0.4982716	0.6151599

```r
plot1 = ggplot(nes1992, aes(x = income, y = as.numeric(vote)))+
  geom_jitter(width = 0.12, height = 0.05, size = 0.05) +
  xlim(0,6) +
  stat_smooth(method = "glm", method.args = list(family = "binomial"),
              se = FALSE, size = 2, col = "black") +
  stat_smooth(method = "glm", method.args = list(family = "binomial"),
              se = FALSE, size = 0.5, fullrange = TRUE, col = "black") +
  theme_bw() +
  theme(plot.background = element_blank()
   ,panel.grid.major = element_blank()
   ,panel.grid.minor = element_blank())

plot2 = ggplot(nes1992, aes(x = income, y = as.numeric(vote)))+
    geom_jitter(width = 0.12, height = 0.05, size = 0.05) +
    xlim(0,5.5) +
    stat_smooth(aes(y = as.numeric(vote)), method="glm", method.args =list(family="binomial"),
                se=TRUE, col = "black") +
    theme_bw() +
    theme(
      plot.background = element_blank()
     ,panel.grid.major = element_blank()
     ,panel.grid.minor = element_blank()
    )

grid.arrange(plot1,plot2,ncol = 2)
```

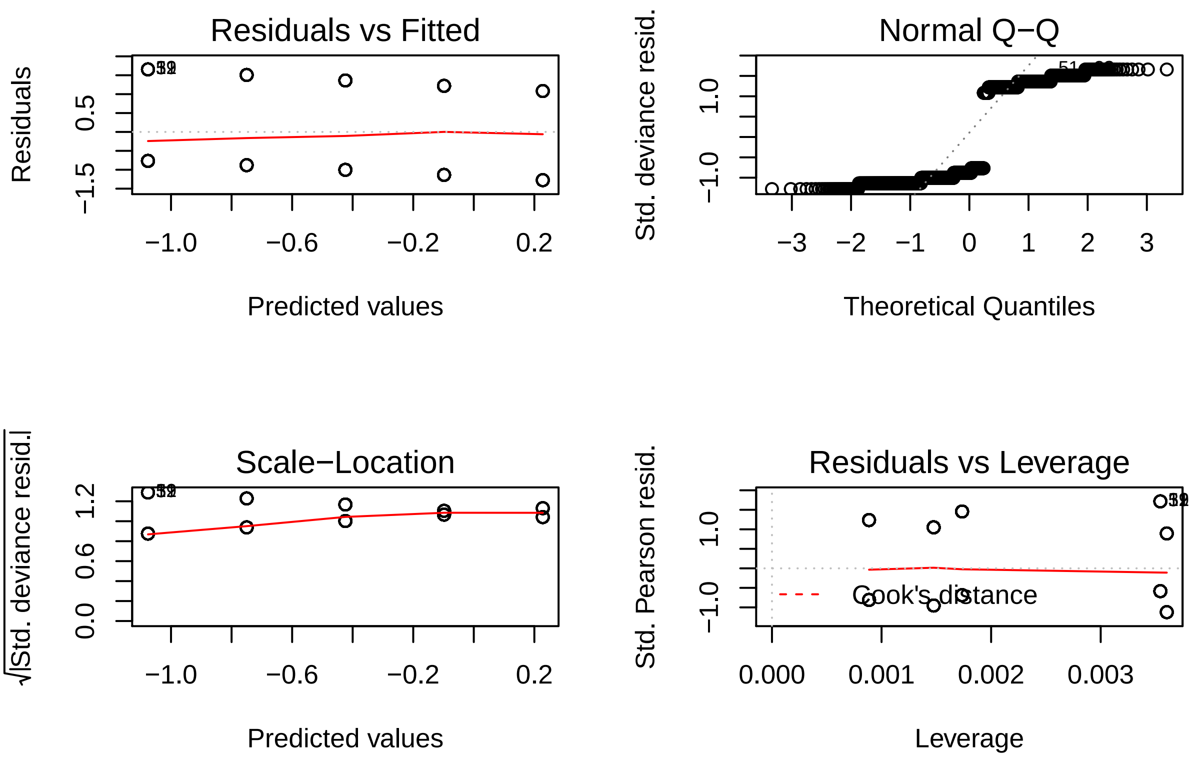


5. What does the residual deviance or any diagnostic plots suggest about the model? (provide code for p-values and output and plots)

```
pchisq(glm.fit$deviance, glm.fit$df.residual, lower = FALSE)
```

```
## [1] 4.971696e-13
```

```
par(mfrow=c(2,2))
plot(glm.fit)
```



Since the p-value of the residual deviance test is very small, we use `pchisq` to obtain the p-value and conclude that deviance is much larger than expected, which indicates the model is lack of fit. In addition, we plot diagonostic plots. However, these plots are hard to explained in the case of binary regression. Deviance analysis is a better diagnostic tool.

6. Create a new data set by the filtering and mutate steps above, but now include years between 1952 and 2000.

```
nes<-read.dta("nes5200_processed_voters_realideo.dta", convert.factors=F)
nesnew=nes %>%
            filter(year>=1952 & year<=2000) %>%
            filter(!is.na(black)) %>%
            filter(!is.na(female)) %>%
            filter(!is.na(educ1))  %>%
            filter(!is.na(age)) %>%
            filter(!is.na(state)) %>%
            filter(!is.na(income)) %>%
            filter(presvote %in% 1:2) %>%
            mutate(female = gender - 1,
```

```
                    black =race==2,
                    vote=presvote==2)
```

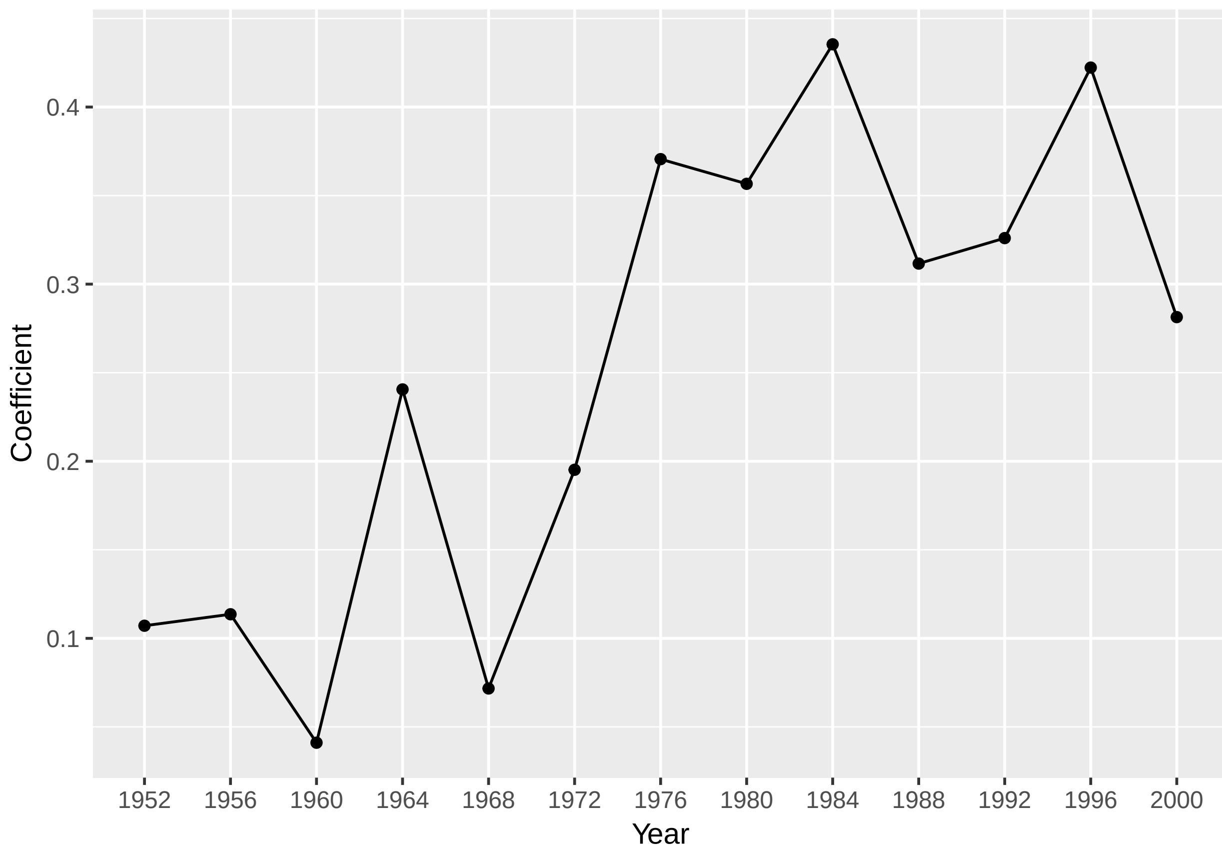
7. Fit a separate logistic regression for each year from 1952 to 2000, using the `subset` option in `glm`, i.e. add `subset=year==1952`. For each find the 95% Confidence interval for the odds ratio of voting republican for rich compared to poor for each year in the data set from 1952 to 2000.

```
# year
Year = unique(nesnew$year)
# empty dataframe
res = as.data.frame(matrix(NA, nrow = length(Year), ncol = 6))
colnames(res) = c("year", "coefficient", "1sd lower bound", "1sd upper bound", "Lower 95% CI of odds ra

res$year = paste(Year)
for (i in 1:length(Year)){
  glm.fit=glm(as.factor(vote) ~ income, data = nesnew, family = binomial, subset=year==Year[i])
  summary = summary(glm.fit)
  ci_beta = confint(glm.fit)[2,]
  se = summary$s
  res[i, 2] = glm.fit$coefficients[2]
  se = summary$coefficients[2,2]
  res[i, 3] = res[i, 2] - 1 * se
  res[i, 4] = res[i, 2] + 1 * se
  odd_ratio = exp(4*ci_beta)
  res[i, 5] = odd_ratio[1]
  res[i, 6] = odd_ratio[2]
}
res
```

```
##    year coefficient 1sd lower bound 1sd upper bound
## 1  1952  0.10711975     0.054536648       0.1597029
## 2  1956  0.11359766     0.061361417       0.1658339
## 3  1960  0.04107133    -0.020368027       0.1025107
## 4  1964  0.24055655     0.183964982       0.2971481
## 5  1968  0.07170134     0.009420254       0.1339824
## 6  1972  0.19518265     0.147719604       0.2426457
## 7  1976  0.37058020     0.314885854       0.4262745
## 8  1980  0.35664702     0.288369991       0.4249241
## 9  1984  0.43542325     0.378122346       0.4927242
## 10 1988  0.31160593     0.251328994       0.3718829
## 11 1992  0.32599471     0.269114036       0.3828754
## 12 1996  0.42227653     0.355838433       0.4887146
## 13 2000  0.28138078     0.218351614       0.3444099
##    Lower 95% CI of odds ratio Upper 95% CI of odds ratio
## 1                   1.0164940                   2.319557
## 2                   1.0466246                   2.375336
## 3                   0.7281162                   1.909628
## 4                   1.6854944                   4.096353
## 5                   0.8177762                   2.173288
## 6                   1.5056952                   3.170386
## 7                   2.8557874                   6.842821
## 8                   2.4501171                   7.154788
## 9                   3.6566346                   8.985704
## 10                  2.1753578                   5.601818
## 11                  2.3676482                   5.779917
```

```
## 12                      3.2382830                     9.187169
## 13                      1.8865722                     5.072581
```

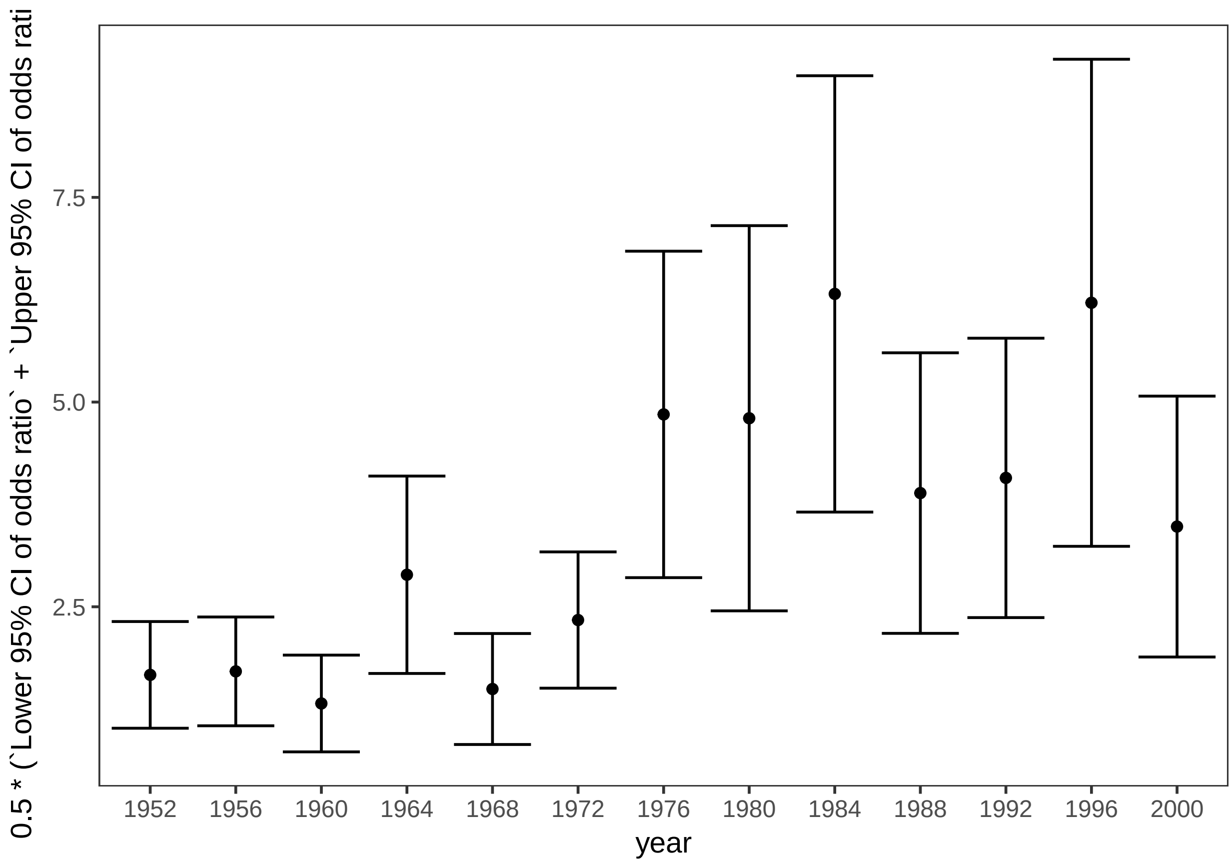
```
qplot(res[,1],res[,2],geom = c("line","point"), group = 1,xlab = "Year", ylab = "Coefficient")
```



From the figure above, we could found that the coefficient over year has an increasing trend.

8. Using `ggplot` plot the confidence intervals over time similar to the display in Figure 5.4.

```
##v_8 <- subset(df_8, select = c(""))

ggplot(res, aes(x = year, y = 0.5*(`Lower 95% CI of odds ratio`+`Upper 95% CI of odds ratio`))) +
geom_point() +
geom_errorbar(aes(ymax = `Lower 95% CI of odds ratio`, ymin=`Upper 95% CI of odds ratio`)) +
theme_bw() +
theme(
  plot.background = element_blank()
 ,panel.grid.major = element_blank()
 ,panel.grid.minor = element_blank()
)
```



The pattarn of richer voter supporting republicanshas increased since 1970. This plot shows the coefficients of income(1-5 scale) with ±1 standard error bounds in logistic regression predicting Replublican preference for president.

9. Fit a logistic regression using income and year as a factor with an interaction i.e. `income*factor(year)` to the data from 1952-2000. Find the log odds ratio for income for each year by combining parameter estimates and show that these are the same as in the respective individual logistic regression models fit separately to the data for each year.

```
# fit model with interaction
glm.fit_IcYr = glm(vote ~ income * factor(year), data = nesnew, family = binomial)
summary(glm.fit_IcYr)
```

```
##
## Call:
## glm(formula = vote ~ income * factor(year), family = binomial,
##     data = nesnew)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6947  -1.1838   0.8667   1.0823   1.7341
##
## Coefficients:
##                            Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -0.017512   0.173405  -0.101 0.919561
## income                     0.107120   0.052583   2.037 0.041635 *
## factor(year)1956           0.046479   0.241891   0.192 0.847627
```

```
## factor(year)1960        -0.090709   0.267884  -0.339 0.734900
## factor(year)1964        -1.451075   0.256633  -5.654 1.57e-08 ***
## factor(year)1968        -0.035074   0.264418  -0.133 0.894474
## factor(year)1972         0.022388   0.230313   0.097 0.922563
## factor(year)1976        -1.144608   0.253984  -4.507 6.59e-06 ***
## factor(year)1980        -0.775574   0.274770  -2.823 0.004763 **
## factor(year)1984        -0.995267   0.253911  -3.920 8.86e-05 ***
## factor(year)1988        -0.818493   0.261529  -3.130 0.001750 **
## factor(year)1992        -1.384618   0.256835  -5.391 7.00e-08 ***
## factor(year)1996        -1.656746   0.281238  -5.891 3.84e-09 ***
## factor(year)2000        -0.982592   0.265677  -3.698 0.000217 ***
## income:factor(year)1956  0.006478   0.074119   0.087 0.930354
## income:factor(year)1960 -0.066048   0.080869  -0.817 0.414080
## income:factor(year)1964  0.133437   0.077250   1.727 0.084108 .
## income:factor(year)1968 -0.035418   0.081511  -0.435 0.663907
## income:factor(year)1972  0.088063   0.070836   1.243 0.213796
## income:factor(year)1976  0.263460   0.076595   3.440 0.000582 ***
## income:factor(year)1980  0.249527   0.086179   2.895 0.003786 **
## income:factor(year)1984  0.328304   0.077771   4.221 2.43e-05 ***
## income:factor(year)1988  0.204486   0.079989   2.556 0.010576 *
## income:factor(year)1992  0.218875   0.077462   2.826 0.004720 **
## income:factor(year)1996  0.315157   0.084729   3.720 0.000200 ***
## income:factor(year)2000  0.174261   0.082083   2.123 0.033756 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 19057  on 13756  degrees of freedom
## Residual deviance: 18303  on 13731  degrees of freedom
## AIC: 18355
##
## Number of Fisher Scoring iterations: 4
```

```r
# extract log odds ratio for each year by combining parameters
a = coef(glm.fit_IcYr)
Sum_IcYr = rep(0,length(Year))
for(i in 1:length(Year)){
  if(i == 1){
    Sum_IcYr[i] = sum(a[c(1,2)])
  }else{
    Sum_IcYr[i] = sum(a[c(1,2,(i+1),(i+13))])
  }
}

# fit separately for each year
Sum_eachYear = rep(0,length(Year))
confInt = data.frame(matrix(c(0,0),ncol = 2))
coef = rep(0,length(Year))
for(i in 1:length(Year)){
  y = Year[i]
  glm.fit_sep = glm(vote ~ income, data = nesnew, subset = year == y, family = binomial)
  Sum_eachYear[i] = sum(coef(glm.fit_sep))
  coef[i] = coefficients(glm.fit_sep)[2]
```

```r
  confInt[i,] = confint(glm.fit_sep)[2,]
}

confInt = cbind(Year,coef,confInt)
names(confInt) = c("Year","coefficient","lower","upper")
# bind dataframe
Compare = data.frame(rbind(Sum_IcYr,Sum_eachYear))
names(Compare) = Year
rownames(Compare) = c("With Interaction term","Fit by each_Year")
Compare = t(Compare)
kable(Compare)
```

	With Interaction term	Fit by each_Year
1952	0.0896081	0.0896081
1956	0.1425647	0.1425647
1960	-0.0671497	-0.0671497
1964	-1.2280301	-1.2280301
1968	0.0191160	0.0191160
1972	0.2000588	0.2000588
1976	-0.7915391	-0.7915391
1980	-0.4364385	-0.4364385
1984	-0.5773551	-0.5773551
1988	-0.5243984	-0.5243984
1992	-1.0761352	-1.0761352
1996	-1.2519809	-1.2519809
2000	-0.7187228	-0.7187228

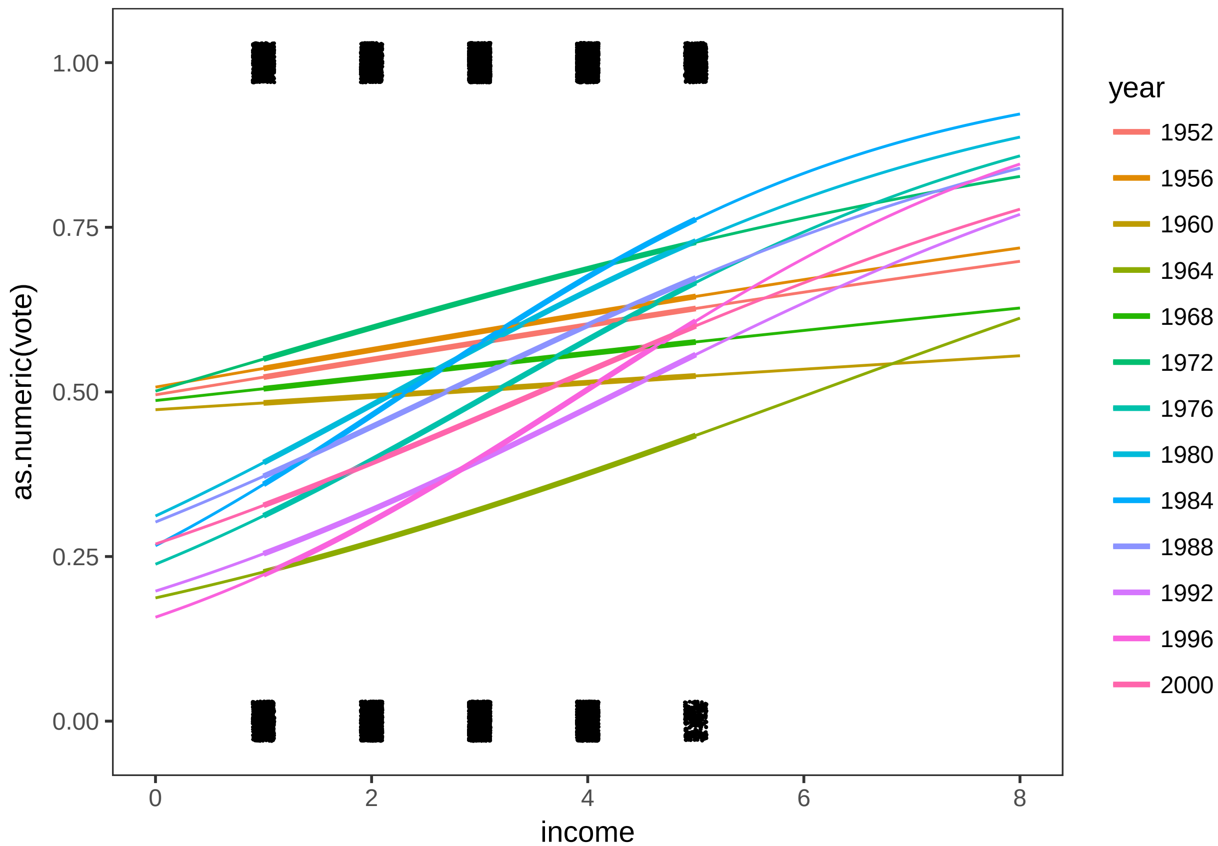
In order to generate the comparison table, we fit the interaction model `glm.fit_IcYr`, and `glm.fit_sep` respectively. In the `glm.fit_sep` model, we fit with the model respect to each year individually in order to obtain the `year + income` coefficient parameters.In the `glm.fit_IcYr`, we obtain each year's coefficient parameters by adding coefficient values of `year` and interaction term `year:income` together. The results is recorded in the `Compare` table. This shows that log odds ratio for income for each year by combining parameter estimates and is as same as in the respective individual logistic regression models fit separately to the data for each year.

10. Create a plot of fitted probabilities and confidence intervals as in question 4, with curves for all years in the same plot.

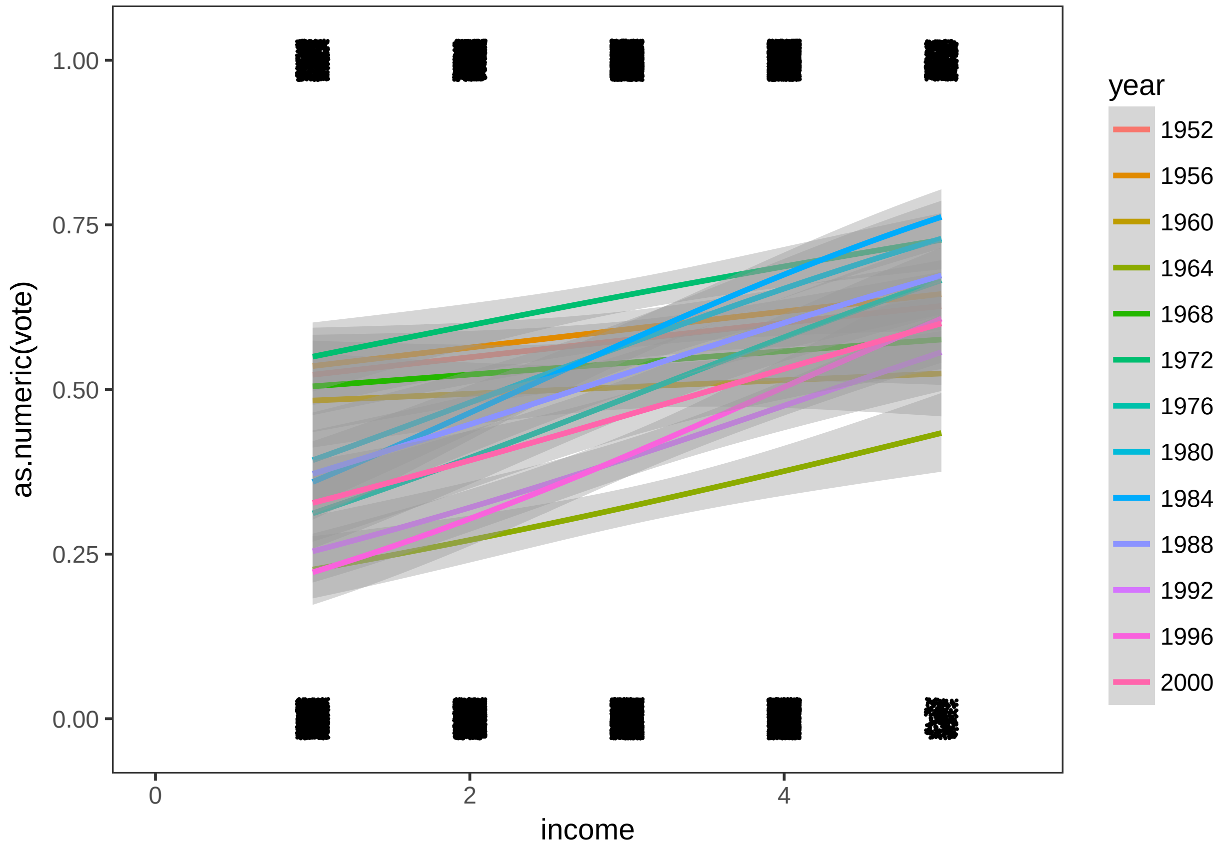
```r
nesall = nes %>% filter(!is.na(black)) %>%
              filter(!is.na(female)) %>%
              filter(!is.na(educ1)) %>%
              filter(!is.na(age)) %>%
              filter(!is.na(state)) %>%
              filter(!is.na(income)) %>%
              filter(presvote %in% 1:2) %>%
# limit to year 19922 t0 2000 and add new varialbes
              mutate(female = gender -1,
                     black=race ==2,
# recode vote so that vote = 1 corresponds to a vote for Bush, and vote=0 is a vote for Clinton, where
                     vote = presvote == 2)
```

```r
#Use ggplot to plot probability and confidence interval for "nesall"
nesall$year = as.factor(nesall$year)
```

```r
ggplot(nesall, aes(x = income, y = as.numeric(vote), group = year,color =year))+
  geom_jitter(width = 0.1, height = 0.03,color = 'black', size = 0.01) +
  geom_point(color = 'black') +
  xlim(0,8) +
  stat_smooth(method = "glm",method.args =list("binomial"), se = FALSE, size = 1) +
  stat_smooth(method = "glm",method.args =list("binomial"),
            se = FALSE, size = 0.5, fullrange = TRUE) +
  theme_bw() +
  theme(
    plot.background = element_blank()
   ,panel.grid.major = element_blank()
   ,panel.grid.minor = element_blank()
  )
```




```r
ggplot(nesall, aes(x = income, y = as.numeric(vote), group = year,color = year))+
  geom_jitter(width = 0.1, height = 0.03,color = 'black', size = 0.01) +
  geom_point(color = 'black') +
  xlim(0,5.5) +
  stat_smooth(method = "glm",method.args =list("binomial"), se = TRUE, size = 1) +
  theme_bw() +
  theme(
    plot.background = element_blank()
   ,panel.grid.major = element_blank()
   ,panel.grid.minor = element_blank()
  )
```



11. Return to the 1992 year data. Filter out rows of `nes1992` with NA's in the variables below and recode as factors using the levels in parentheses:
    - gender (1 = "male", 2 = "female"),
    - race (1 = "white", 2 = "black", 3 = "asian", 4 = "native american", 5 = "hispanic", 7 = "other"),
    - education (use `educ1` with levels 1 = "no high school", 2 = "high school graduate", 3 = "some college", 4 = "college graduate"),
    - party identification (`partyid3` with levels 1= "democrats", 2 = "independents", 3 = "republicans", 4 = "apolitical" , and
    - political ideology (`ideo` 1 = "liberal", 2 ="moderate", 3 = "conservative")

```
nes1992 = nes1992 %>% filter(!is.na(gender)) %>%
              filter(!is.na(race))  %>%
              filter(!is.na(educ1)) %>%
              filter(!is.na(partyid3)) %>%
              filter(!is.na(ideo))  %>%
              mutate(gender=recode_factor(gender,'1'="male",'2'="female"),
                     race=recode_factor(race,"1"="white","2"="black",
                                        "3"="asian","4"="native american",
                                        "5"="hispanic","7"="other"),
                     educ1=recode_factor(educ1,"1"="no high school",
                                         "2"="high school graduate",
                                         "3"="some college",
                                         "4"= "college graduate"),
                     partyid3=recode_factor(partyid3,"1"="democrats",
                                            "2"="independents",
                                            "3"="republicans",
                                            "9"="apolitical"),
```

```
                      ideo=recode_factor(ideo,"1"="liberal",
                                         "3"="moderate",
                                         "5"="conservative")
                      )
```

12. Fit a logistic regression model predicting support for Bush given the the variables above and income as predictors and also consider interactions among the predictors. You do not need to consider all possible interactions or use model selection, but suggest a couple from the predictors above that might make sense intuitively.

```
nes1992$race = factor(nes1992$race)
nes1992$gender = factor(nes1992$gender)
nes1992$educ1 = factor(nes1992$educ1)
nes1992$partyid3 = factor(nes1992$partyid3)
nes1992$ideo = factor(nes1992$ideo)
glm.full = glm(vote ~ (income + gender + race + educ1 + partyid3 + ideo)^2, data = nes1992, family = "bi
backwards = step(glm.full,trace=0)
summary(backwards)
```

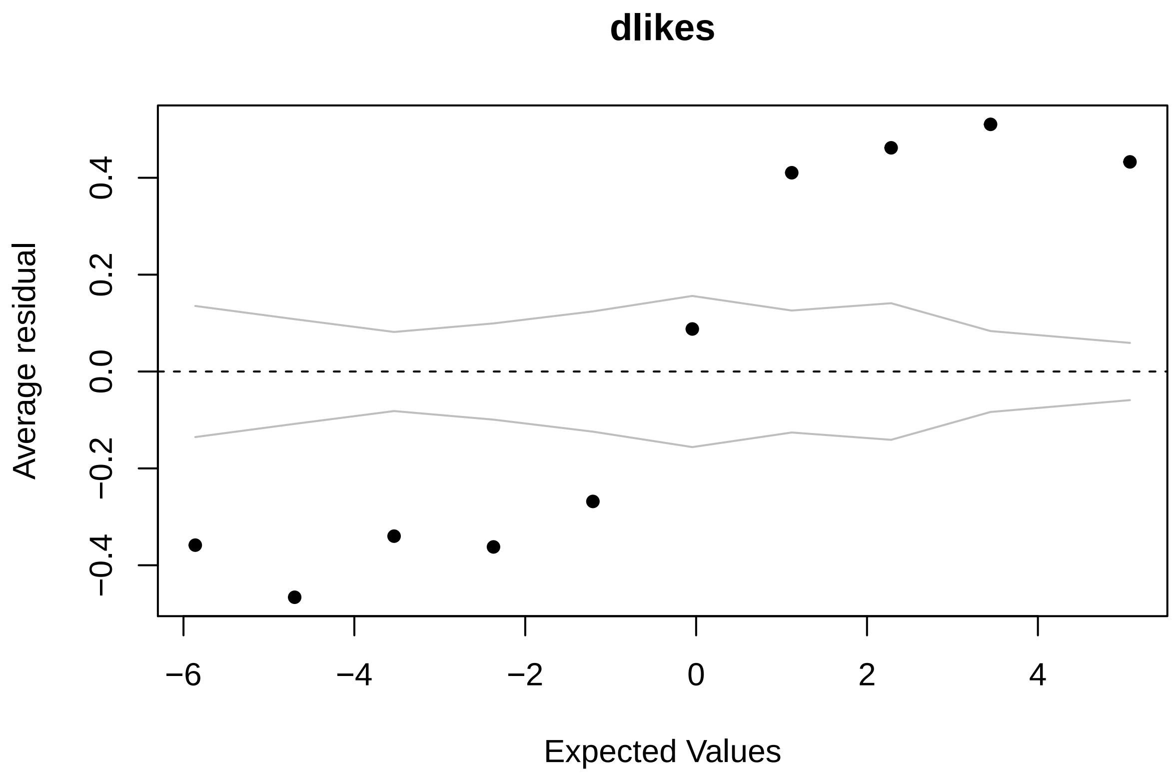
```
##
## Call:
## glm(formula = vote ~ income + gender + race + partyid3 + ideo +
##     income:partyid3 + gender:race + gender:partyid3, family = "binomial",
##     data = nes1992)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5413  -0.3417  -0.1593   0.3705   3.2193
##
## Coefficients: (2 not defined because of singularities)
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                        -4.60975    0.63195  -7.295 3.00e-13
## income                              0.24960    0.15318   1.629  0.10321
## genderfemale                        0.79965    0.38541   2.075  0.03800
## raceblack                          -1.52903    0.74809  -2.044  0.04096
## raceasian                          -1.08601    0.98780  -1.099  0.27158
## racenative american                 0.05502    1.52872   0.036  0.97129
## racehispanic                       -1.21741    1.09995  -1.107  0.26839
## partyid3independents                2.05254    1.11395   1.843  0.06539
## partyid3republicans                 6.85413    0.87054   7.873 3.45e-15
## partyid3apolitical                -13.83371 1455.39758  -0.010  0.99242
## ideomoderate                        0.91170    0.39451   2.311  0.02083
## ideoconservative                    1.87900    0.25062   7.498 6.50e-14
## income:partyid3independents        -0.13200    0.30076  -0.439  0.66074
## income:partyid3republicans         -0.61939    0.22448  -2.759  0.00579
## income:partyid3apolitical                NA         NA      NA       NA
## genderfemale:raceblack             -0.83542    1.00215  -0.834  0.40449
## genderfemale:raceasian             15.82631  567.00064   0.028  0.97773
## genderfemale:racenative american    0.61637    1.66059   0.371  0.71051
## genderfemale:racehispanic           2.75136    1.21428   2.266  0.02346
## genderfemale:partyid3independents   0.41795    0.69022   0.606  0.54482
## genderfemale:partyid3republicans   -1.36459    0.49924  -2.733  0.00627
## genderfemale:partyid3apolitical          NA         NA      NA       NA
##
```

```
## (Intercept)                    ***
## income
## genderfemale                   *
## raceblack                      *
## raceasian
## racenative american
## racehispanic
## partyid3independents           .
## partyid3republicans            ***
## partyid3apolitical
## ideomoderate                   *
## ideoconservative               ***
## income:partyid3independents
## income:partyid3republicans     **
## income:partyid3apolitical
## genderfemale:raceblack
## genderfemale:raceasian
## genderfemale:racenative american
## genderfemale:racehispanic      *
## genderfemale:partyid3independents
## genderfemale:partyid3republicans  **
## genderfemale:partyid3apolitical
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1534.10  on 1132  degrees of freedom
## Residual deviance:  629.68  on 1113  degrees of freedom
## AIC: 669.68
##
## Number of Fisher Scoring iterations: 14
```

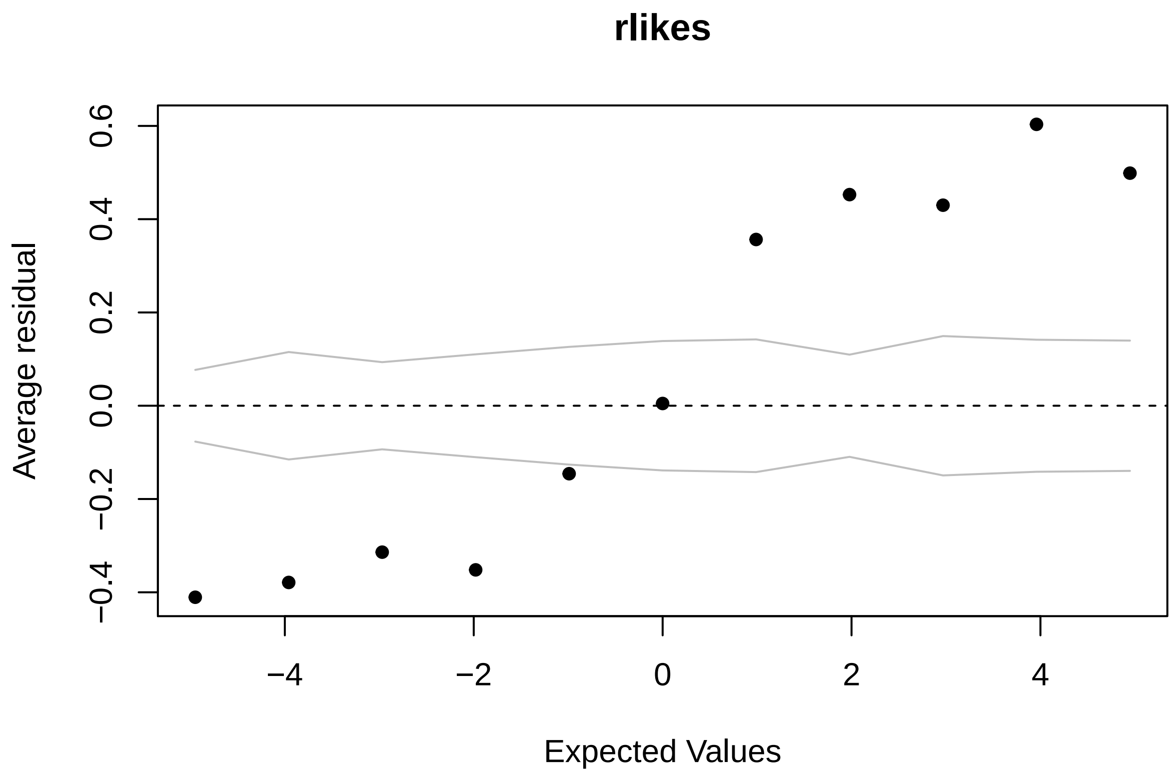
We use backward selection to select the variables, and the preserved variables are `income`, `gender`, `race`, `partyid3`, `ideo`, `income:partyid3`, `gender:race`, `gender:partyid3`. Initially, We construct the full model by assuming each of the main effects (`income`, `gender`, `race`, `partyid3`, `educ1`, `ideo`) have interactions with one another.

13. Plot binned residuals using the function `binnedplot` from package `arm` versus some of the additional predictors in the 1992 dataframe. Are there any suggestions that the mean or distribution of residuals is different across the levels of the other predictors and that they should be added to the model? (Provide plots and any other summaries to explain).

```r
x = predict(backwards)
y = resid(backwards)

## fit dlikes
fit_dlikes = glm(vote ~ dlikes, data = nes1992, family = binomial(link = "logit"))
x = predict(fit_dlikes)
y = resid(backwards)
binnedplot(x,y, main = "dlikes")
```



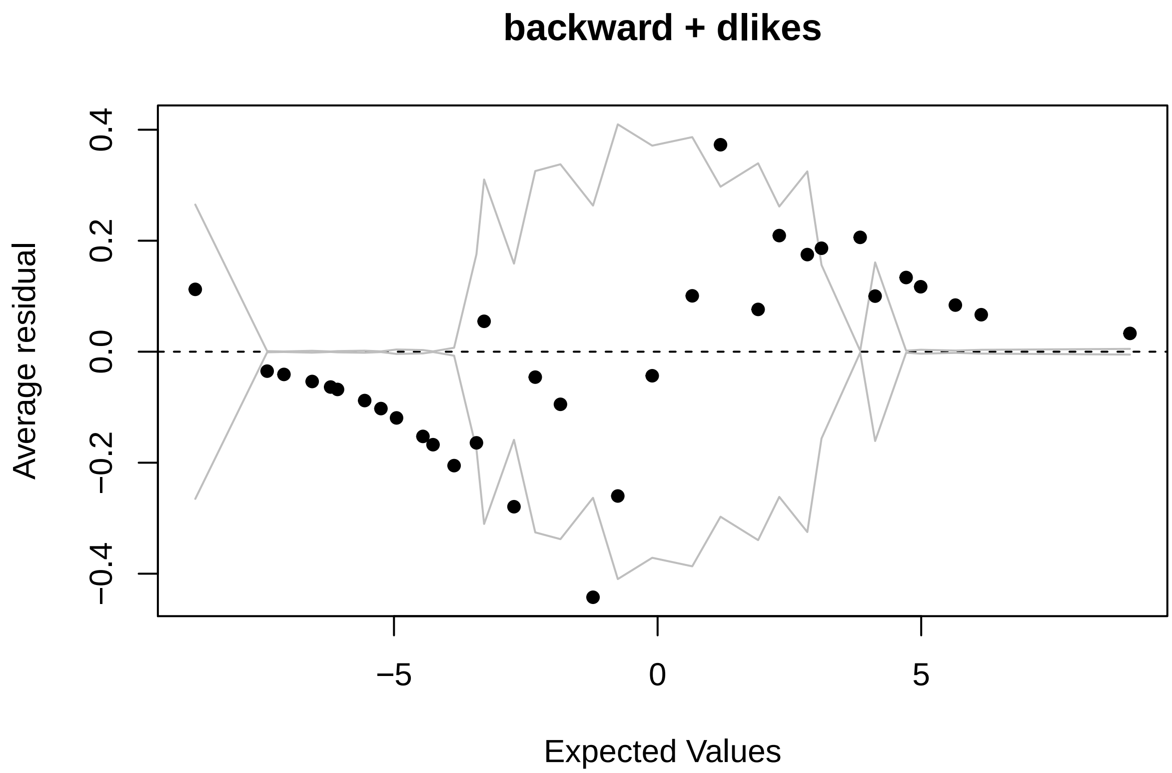

```r
## fit rlikes
fit_rlikes = glm(vote ~ rlikes, data = nes1992, family = binomial(link = "logit"))
x = predict(fit_rlikes)
y = resid(backwards)
binnedplot(x,y, main = "rlikes")
```




```
cor(nes1992$dlikes,nes1992$rlikes)
```

```
## [1] -0.6515694
```

```
# compare
fitt = step(glm(vote ~ (income + gender + race + partyid3 + ideo)^2 + dlikes, family = binomial(link =
x = predict(fitt)
y = resid(fitt)
binnedplot(x,y, main = "backward + dlikes")
```

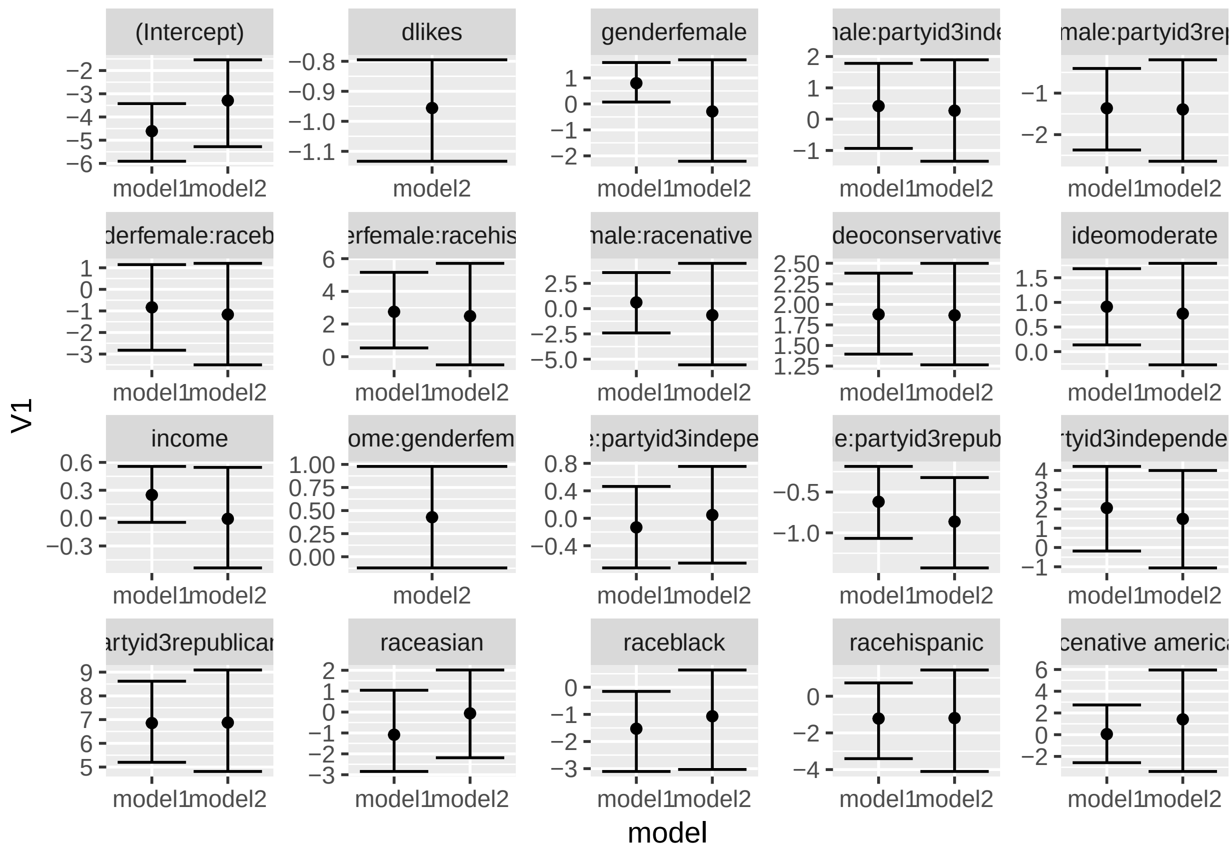


If there are more points fall inside the boundary, the model could be considered as a better model. By comparing the model fitted by `rlikes`, `dlikes` and backward selected model, we consider fitting `rlikes` and `dlikes`. In addition, due to the strong correlation between these two variables, we only keep one and added in the backward selected models. The binned plot shows that the model become better.

14. Evaluate and compare the different models you fit. Consider coefficient estimates (are they stable across models) and standard errors (any indications of identifiability problems), residual plots and deviances.

```
a = names(coefficients(fitt))
b = names(coefficients(backwards))


model2 = cbind(coefficients(fitt), confint(fitt))
model2 = cbind(model2, model = "model2")
model1 = cbind(coefficients(backwards), confint(backwards))
model1 = cbind(model1, model = "model1")


features = rownames(model1)
model1 = cbind(model1, features)

features = rownames(model2)
model2 = cbind(model2, features)

miss = matrix(NA,ncol = 5, nrow = 2 )
miss[,5] = c(a[a %in% b == FALSE])
```

```r
df = data.frame(rbind(model1,model2,miss))
df$V1 = as.numeric(as.character(df$V1))
df$X2.5 = as.numeric(as.character(df$X2.5))
df$X97.5 = as.numeric(as.character(df$X97.5))

df = na.omit(df)

ggplot(data = df, aes(x = model, y = V1 , group = features )) +
  geom_point() +
  geom_errorbar(aes(ymax = `X97.5`, ymin=`X2.5`)) +
  facet_wrap(~features,scales = "free")
```




```r
confint(fitt, method="boot")
```

```
##                                      2.5 %      97.5 %
## (Intercept)                     -5.2791104  -1.5375205
## income                          -0.5363101   0.5460988
## genderfemale                    -2.2092173   1.6996613
## raceblack                       -3.0298104   0.6368221
## raceasian                       -2.1868446   2.0152491
## racenative american             -3.3832287   5.9566557
## racehispanic                    -4.1015167   1.4312955
## partyid3independents            -1.0594406   4.0010015
## partyid3republicans              4.8152600   9.0911744
## partyid3apolitical                      NA 473.5105784
## ideomoderate                    -0.2680624   1.7923746
```

```
## ideoconservative                    1.2650182   2.4989420
## dlikes                              -1.1332118  -0.7950607
## income:genderfemale                 -0.1209976   0.9782042
## income:partyid3independents         -0.6525351   0.7551279
## income:partyid3republicans          -1.4295276  -0.3227824
## income:partyid3apolitical                   NA          NA
## genderfemale:raceblack              -3.4991030   1.2049145
## genderfemale:raceasian             -63.9849592          NA
## genderfemale:racenative american    -5.5561872   4.4749107
## genderfemale:racehispanic           -0.4961754   5.7122677
## genderfemale:partyid3independents  -1.3456622   1.8953922
## genderfemale:partyid3republicans   -2.6429290  -0.1963891
## genderfemale:partyid3apolitical             NA          NA
```

```
anova(backwards,fitt, test = 'Chi')
```

```
## Analysis of Deviance Table
##
## Model 1: vote ~ income + gender + race + partyid3 + ideo + income:partyid3 +
##     gender:race + gender:partyid3
## Model 2: vote ~ income + gender + race + partyid3 + ideo + dlikes + income:gender +
##     income:partyid3 + gender:race + gender:partyid3
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1      1113     629.68
## 2      1111     414.79  2   214.88 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From the previous question, we added 'dlikes' to the model. To check whether any coefficients become more unstable after we change the model, we plotted coefficients in both models with their intervals. From the figure above, we found that most of confidence interval of the coefficients become slightly larger. In addition, the coeffcient of variable `genderfemale`, `genderfemale:racenativeamerican` and `income:partyid3independent` change the sign. These variables can be considered unstable while we change the model.

Further, in order to observe the identifiability problems, we construct the \$ 95% \$ confidence interval and find that `partyid3apolitical` and `genderfemale:raceasian` has extremely large interval, which might be a indicator of identifiability problems. The reason might be the lack of the observations: only 1 for `partyid3(apolitical)` and 5 for `gender(female):race(asian)`.

In addition, we use anova test to test the deviance between these two models. The deviance was reduced by 214.88 which is much larger than 1. The goodness of fit test with this deviance indicates that there is no lack of fit issue in our model as well.

15. Compute the error rate of your model (see GH page 99) and compare it to the error rate of the null model. We can define a function for the error rate as:

The error rate in our model is weigh better than the rate in the null model. In our model, the error rate is 0.0723742 while in null model the rate is 0.4104148.

16. For your chosen model, discuss and compare the importance of each input variable in the prediction. Provide a neatly formatted table of odds ratios and 95% confidence intervals.

```
summary(fitt)
```

```
##
## Call:
## glm(formula = vote ~ income + gender + race + partyid3 + ideo +
##     dlikes + income:gender + income:partyid3 + gender:race +
```

```
##     gender:partyid3, family = binomial(link = "logit"), data = nes1992)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9550  -0.1799  -0.0452   0.1599   4.2224
##
## Coefficients: (2 not defined because of singularities)
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      -3.291e+00  9.511e-01  -3.460 0.000539
## income                           -7.825e-03  2.752e-01  -0.028 0.977313
## genderfemale                     -2.894e-01  9.925e-01  -0.292 0.770578
## raceblack                        -1.069e+00  9.391e-01  -1.138 0.254976
## raceasian                        -6.270e-02  1.084e+00  -0.058 0.953863
## racenative american               1.412e+00  3.734e+00   0.378 0.705427
## racehispanic                     -1.192e+00  1.550e+00  -0.769 0.441868
## partyid3independents              1.485e+00  1.282e+00   1.158 0.246828
## partyid3republicans               6.875e+00  1.088e+00   6.317 2.66e-10
## partyid3apolitical               -1.511e+01  2.400e+03  -0.006 0.994975
## ideomoderate                      7.712e-01  5.252e-01   1.468 0.142028
## ideoconservative                  1.868e+00  3.139e-01   5.949 2.70e-09
## dlikes                           -9.558e-01  8.606e-02 -11.106  < 2e-16
## income:genderfemale               4.282e-01  2.796e-01   1.531 0.125688
## income:partyid3independents       4.796e-02  3.571e-01   0.134 0.893162
## income:partyid3republicans       -8.626e-01  2.817e-01  -3.063 0.002193
## income:partyid3apolitical                NA         NA      NA       NA
## genderfemale:raceblack           -1.167e+00  1.193e+00  -0.978 0.327829
## genderfemale:raceasian            1.730e+01  8.171e+02   0.021 0.983106
## genderfemale:racenative american -6.448e-01  3.815e+00  -0.169 0.865779
## genderfemale:racehispanic         2.484e+00  1.692e+00   1.468 0.142223
## genderfemale:partyid3independents  2.708e-01  8.239e-01   0.329 0.742453
## genderfemale:partyid3republicans -1.393e+00  6.217e-01  -2.241 0.025050
## genderfemale:partyid3apolitical          NA         NA      NA       NA
##
## (Intercept)                      ***
## income
## genderfemale
## raceblack
## raceasian
## racenative american
## racehispanic
## partyid3independents
## partyid3republicans              ***
## partyid3apolitical
## ideomoderate
## ideoconservative                 ***
## dlikes                           ***
## income:genderfemale
## income:partyid3independents
## income:partyid3republicans       **
## income:partyid3apolitical
## genderfemale:raceblack
## genderfemale:raceasian
## genderfemale:racenative american
## genderfemale:racehispanic
```

```
## genderfemale:partyid3independents
## genderfemale:partyid3republicans  *
## genderfemale:partyid3apolitical
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1534.10  on 1132  degrees of freedom
## Residual deviance:  414.79  on 1111  degrees of freedom
## AIC: 458.79
##
## Number of Fisher Scoring iterations: 15
```

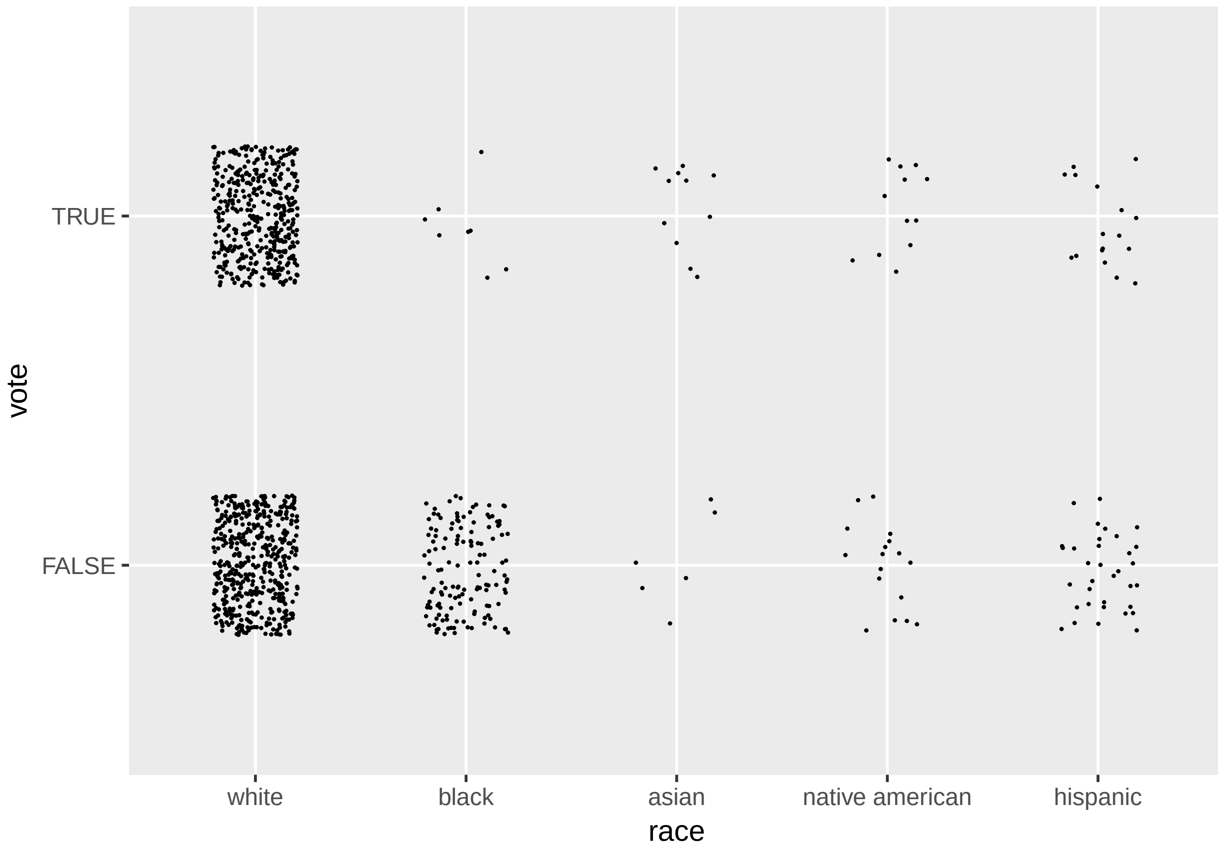
```r
ratio_table = as.data.frame(round(exp(confint(fitt)),4))
coef_ratio = as.data.frame(fitt$coefficients)
table = cbind(coef_ratio, ratio_table)
kable(table)
```

	fitt$coefficients	2.5 %	97.5 %
(Intercept)	-3.2911950	0.0051	2.149000e-01
income	-0.0078248	0.5849	1.726500e+00
genderfemale	-0.2894264	0.1098	5.472100e+00
raceblack	-1.0689657	0.0483	1.890500e+00
raceasian	-0.0627039	0.1123	7.502600e+00
racenative american	1.4116304	0.0339	3.863160e+02
racehispanic	-1.1923364	0.0165	4.184100e+00
partyid3independents	1.4850723	0.3466	5.465290e+01
partyid3republicans	6.8746331	123.3789	8.876605e+03
partyid3apolitical	-15.1111413	NA	4.395733e+205
ideomoderate	0.7711920	0.7649	6.003700e+00
ideoconservative	1.8675672	3.5432	1.216960e+01
dlikes	-0.9557684	0.3220	4.516000e-01
income:genderfemale	0.4281647	0.8860	2.659700e+00
income:partyid3independents	0.0479609	0.5207	2.127900e+00
income:partyid3republicans	-0.8626485	0.2394	7.241000e-01
income:partyid3apolitical	NA	NA	NA
genderfemale:raceblack	-1.1670234	0.0302	3.336500e+00
genderfemale:raceasian	17.3029478	0.0000	NA
genderfemale:racenative american	-0.6447958	0.0039	8.778680e+01
genderfemale:racehispanic	2.4835955	0.6089	3.025564e+02
genderfemale:partyid3independents	0.2707510	0.2604	6.655200e+00
genderfemale:partyid3republicans	-1.3929956	0.0712	8.217000e-01
genderfemale:partyid3apolitical	NA	NA	NA

According to the summary of model `fitt`, variables `partyid3republicans` and `ideoconservative` are significant compared with their base level. The main variable `dlikes`, and interaction terms `income:partyid3republicans`, and `genderfemale:partyid3republicans` are also statistically significant.

17. Provide a paragraph summarizing your findings and interpreting key coefficients (providing ranges of supporting values from above) in terms of the odds of voting for Bush. Attempt to write this at a level that readers of the New York Times Upshot column could understand.

```r
ggplot(data = nes1992, aes(x = race , y = vote)) +
  geom_jitter(size = 0.1, height = 0.2, width = 0.2)
```



Based on `table` from previous question and above figure, when other variables is constant and the baseline of varaible race is the white population. The log odds difference of voting Bush between black and white populations is -1.06. The log odds difference of voting Bush of between aisna and white populations is -0.06. The log odds difference of voting Bush between native american and white populations is 1.41. The log odds difference of voting Bush between hispanic and white populations is -1.19.

18. In the above analysis, we removed missing data. Repeat the data cleaning steps, but remove only the rows where the response variable, `presvote` is missing. Recode all of the predictors (including income) so that there is a level that is 'missing' for any NA's for each variable. How many observations are there now compared to the complete data?

```r
nes<-read.dta("nes5200_processed_voters_realideo.dta", convert.factors=F)

## chang NA as missing instead of getting rid of it
nes$black[which(is.na(nes$black))] = "Missing"
nes$female[which(is.na(nes$female))] = "Missing"
nes$educ1[which(is.na(nes$educ1))] = "Missing"
nes$age[which(is.na(nes$age))] = "Missing"
nes$state[which(is.na(nes$state))] = "Missing"
nes$income[which(is.na(nes$income))] = "Missing"

nesmiss=nes %>%
            filter(!is.na(presvote)) %>%
            filter(year == 1992) %>%
```

```r
            filter(presvote %in% 1:2) %>%
            mutate(female = gender - 1,
                   black =race==2,
                   vote=presvote==2)

n_new = nrow(nesmiss)
n_old = nrow(nes1992)
```

By label missing values, we now have `n_new` observations and in previous data set we only have `n_old` obervations.

19. For any of above variables, suggest possible reasons why they may be missing.

Possible reasons for the missing variables include: people in the survey provide no response. For example income in this database, it's a variable that is kind of private, so people may be not willing to share this information, gender and age are also this kind of information sometimes. Another reason why participates tend to provide no reply is that the measurement of certain variables is repeated after a certain period of time. For example the education in this database, people in the survey may drop out before the test ends and one or more measurements are missing. Missing variables also have relevance with research fields, some fields like politics and sociology, issues in these fields are critical and sensitive, this makes governments choose or totally fail to report relevant information. Sometimes researchers will make some mistakes in data collection as well as data entry, which leads to missing variables.

20. Rerun your selected model and create a table of parameter estimates and confidence intervals for the odds ratios. You should have an additional coefficient for any categorical variable with missing data. Comment on any changes in results for the model including the missing data and the previous one that used only complete data.

```r
nesmiss$income = as.factor(nesmiss$income)
nesmiss$race = factor(nesmiss$race)
nesmiss$gender = factor(nesmiss$gender)
nesmiss$educ1 = factor(nesmiss$educ1)
nesmiss$partyid3 = factor(nesmiss$partyid3)
nesmiss$ideo = factor(nesmiss$ideo)
summary(fitt)
```

```
##
## Call:
## glm(formula = vote ~ income + gender + race + partyid3 + ideo +
##     dlikes + income:gender + income:partyid3 + gender:race +
##     gender:partyid3, family = binomial(link = "logit"), data = nes1992)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9550  -0.1799  -0.0452   0.1599   4.2224
##
## Coefficients: (2 not defined because of singularities)
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      -3.291e+00  9.511e-01  -3.460 0.000539
## income                           -7.825e-03  2.752e-01  -0.028 0.977313
## genderfemale                     -2.894e-01  9.925e-01  -0.292 0.770578
## raceblack                        -1.069e+00  9.391e-01  -1.138 0.254976
## raceasian                        -6.270e-02  1.084e+00  -0.058 0.953863
## racenative american               1.412e+00  3.734e+00   0.378 0.705427
## racehispanic                     -1.192e+00  1.550e+00  -0.769 0.441868
## partyid3independents              1.485e+00  1.282e+00   1.158 0.246828
```

```
## partyid3republicans              6.875e+00  1.088e+00   6.317 2.66e-10
## partyid3apolitical              -1.511e+01  2.400e+03  -0.006 0.994975
## ideomoderate                     7.712e-01  5.252e-01   1.468 0.142028
## ideoconservative                 1.868e+00  3.139e-01   5.949 2.70e-09
## dlikes                          -9.558e-01  8.606e-02 -11.106  < 2e-16
## income:genderfemale              4.282e-01  2.796e-01   1.531 0.125688
## income:partyid3independents      4.796e-02  3.571e-01   0.134 0.893162
## income:partyid3republicans      -8.626e-01  2.817e-01  -3.063 0.002193
## income:partyid3apolitical               NA         NA      NA       NA
## genderfemale:raceblack          -1.167e+00  1.193e+00  -0.978 0.327829
## genderfemale:raceasian           1.730e+01  8.171e+02   0.021 0.983106
## genderfemale:racenative american -6.448e-01  3.815e+00  -0.169 0.865779
## genderfemale:racehispanic        2.484e+00  1.692e+00   1.468 0.142223
## genderfemale:partyid3independents  2.708e-01  8.239e-01   0.329 0.742453
## genderfemale:partyid3republicans -1.393e+00  6.217e-01  -2.241 0.025050
## genderfemale:partyid3apolitical         NA         NA      NA       NA
##
## (Intercept)                     ***
## income
## genderfemale
## raceblack
## raceasian
## racenative american
## racehispanic
## partyid3independents
## partyid3republicans             ***
## partyid3apolitical
## ideomoderate
## ideoconservative                ***
## dlikes                          ***
## income:genderfemale
## income:partyid3independents
## income:partyid3republicans      **
## income:partyid3apolitical
## genderfemale:raceblack
## genderfemale:raceasian
## genderfemale:racenative american
## genderfemale:racehispanic
## genderfemale:partyid3independents
## genderfemale:partyid3republicans  *
## genderfemale:partyid3apolitical
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1534.10  on 1132  degrees of freedom
## Residual deviance:  414.79  on 1111  degrees of freedom
## AIC: 458.79
##
## Number of Fisher Scoring iterations: 15
```

```
fitt.miss = glm(formula = vote ~ income + gender + race + partyid3 + ideo +
    dlikes + income:gender + income:partyid3 + gender:race +
```

```r
    gender:partyid3, family = binomial(link = "logit"), data = nesmiss)
summary(fitt.miss)
```

```
##
## Call:
## glm(formula = vote ~ income + gender + race + partyid3 + ideo +
##     dlikes + income:gender + income:partyid3 + gender:race +
##     gender:partyid3, family = binomial(link = "logit"), data = nesmiss)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7299  -0.2101  -0.0482   0.1910   4.1649
##
## Coefficients: (6 not defined because of singularities)
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -3.03920    0.93659  -3.245  0.00117 **
## income2                   -0.89974    1.25716  -0.716  0.47418
## income3                    0.58672    1.02162   0.574  0.56576
## income4                   -0.42098    1.04330  -0.404  0.68658
## income5                   -0.84183    1.49614  -0.563  0.57366
## incomeMissing              0.41719    1.42690   0.292  0.77000
## gender2                    0.11194    1.04478   0.107  0.91467
## race2                     -1.11748    0.93692  -1.193  0.23298
## race3                      0.22218    1.02352   0.217  0.82815
## race4                      0.76725    2.99011   0.257  0.79749
## race5                     -1.21567    1.49593  -0.813  0.41642
## partyid32                  0.51327    1.46769   0.350  0.72655
## partyid33                  5.22451    1.09858   4.756 1.98e-06 ***
## partyid39                -14.85349 2399.54478  -0.006  0.99506
## ideo3                      0.51472    0.50237   1.025  0.30556
## ideo5                      1.74760    0.29645   5.895 3.75e-09 ***
## dlikes                    -0.95951    0.08478 -11.317  < 2e-16 ***
## income2:gender2            0.63945    1.30610   0.490  0.62442
## income3:gender2            0.58382    1.10785   0.527  0.59820
## income4:gender2            1.76081    1.13109   1.557  0.11953
## income5:gender2            1.32203    1.38489   0.955  0.33977
## incomeMissing:gender2      0.67033    1.39698   0.480  0.63134
## income2:partyid32          1.82725    1.64582   1.110  0.26690
## income3:partyid32          0.45513    1.51812   0.300  0.76433
## income4:partyid32         -0.39826    1.62158  -0.246  0.80599
## income5:partyid32          2.58963    1.96391   1.319  0.18730
## incomeMissing:partyid32   -1.66118    2.63817  -0.630  0.52891
## income2:partyid33          1.33595    1.26754   1.054  0.29190
## income3:partyid33         -1.82286    1.12872  -1.615  0.10632
## income4:partyid33         -1.43618    1.10606  -1.298  0.19413
## income5:partyid33         -2.04395    1.49615  -1.366  0.17190
## incomeMissing:partyid33   -2.53783    1.38878  -1.827  0.06764 .
## income2:partyid39               NA         NA      NA       NA
## income3:partyid39               NA         NA      NA       NA
## income4:partyid39               NA         NA      NA       NA
## income5:partyid39               NA         NA      NA       NA
## incomeMissing:partyid39         NA         NA      NA       NA
## gender2:race2             -1.17055    1.17096  -1.000  0.31748
## gender2:race3             17.68402  743.02671   0.024  0.98101
```

```
## gender2:race4            -0.07664    3.08871  -0.025  0.98021
## gender2:race5             2.19685    1.63464   1.344  0.17897
## gender2:partyid32         0.70193    0.84506   0.831  0.40619
## gender2:partyid33        -1.60977    0.59356  -2.712  0.00669 **
## gender2:partyid39              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1671.82  on 1229  degrees of freedom
## Residual deviance:  467.57  on 1192  degrees of freedom
##   (74 observations deleted due to missingness)
## AIC: 543.57
##
## Number of Fisher Scoring iterations: 15
```

```r
ratio_table = as.data.frame(round(exp(confint(fitt)),4))
coef_ratio = as.data.frame(fitt$coefficients)
table = cbind(coef_ratio, ratio_table)

ratio_misstable = as.data.frame(round(exp(confint(fitt.miss)),4))
coef_missratio = as.data.frame(exp(fitt.miss$coefficients))
table_miss = cbind(coef_missratio, ratio_misstable)
kable(table_miss)
```

	exp(fitt.miss$coefficients)	2.5 %	97.5 %
(Intercept)	4.787300e-02	0.0058	2.431000e-01
income2	4.066734e-01	0.0355	5.360200e+00
income3	1.798075e+00	0.2793	1.652390e+01
income4	6.564063e-01	0.0947	6.136700e+00
income5	4.309196e-01	0.0226	8.596200e+00
incomeMissing	1.517687e+00	0.0952	2.701180e+01
gender2	1.118449e+00	0.1463	9.619400e+00
race2	3.271023e-01	0.0459	1.784300e+00
race3	1.248795e+00	0.1748	9.348800e+00
race4	2.153834e+00	0.0285	1.709995e+02
race5	2.965103e-01	0.0167	3.696500e+00
partyid32	1.670748e+00	0.0541	2.432420e+01
partyid33	1.857701e+02	25.1841	1.994613e+03
partyid39	4.000000e-07	NA	4.722785e+205
ideo3	1.673170e+00	0.6209	4.455000e+00
ideo5	5.740828e+00	3.2438	1.039930e+01
dlikes	3.830809e-01	0.3217	4.488000e-01
income2:gender2	1.895442e+00	0.1389	2.469260e+01
income3:gender2	1.792878e+00	0.1903	1.574680e+01
income4:gender2	5.817152e+00	0.6052	5.490280e+01
income5:gender2	3.751038e+00	0.2379	5.720540e+01
incomeMissing:gender2	1.954877e+00	0.1189	2.997940e+01
income2:partyid32	6.216742e+00	0.2895	2.401148e+02
income3:partyid32	1.576377e+00	0.0946	5.040360e+01
income4:partyid32	6.714857e-01	0.0311	2.438640e+01
income5:partyid32	1.332484e+01	0.3297	8.801582e+02
incomeMissing:partyid32	1.899144e-01	0.0011	2.933020e+01

	exp(fitt.miss$coefficients)	2.5 %	97.5 %
income2:partyid33	3.803607e+00	0.2820	4.360810e+01
income3:partyid33	1.615635e-01	0.0146	1.305200e+00
income4:partyid33	2.378349e-01	0.0223	1.824900e+00
income5:partyid33	1.295167e-01	0.0065	2.531200e+00
incomeMissing:partyid33	7.903800e-02	0.0047	1.191000e+00
income2:partyid39	NA	NA	NA
income3:partyid39	NA	NA	NA
income4:partyid39	NA	NA	NA
income5:partyid39	NA	NA	NA
incomeMissing:partyid39	NA	NA	NA
gender2:race2	3.101966e-01	0.0312	3.198800e+00
gender2:race3	4.787079e+07	0.0000	NA
gender2:race4	9.262263e-01	0.0080	9.705920e+01
gender2:race5	8.996673e+00	0.5079	2.146818e+02
gender2:partyid32	2.017642e+00	0.3884	1.081380e+01
gender2:partyid33	1.999337e-01	0.0606	6.264000e-01
gender2:partyid39	NA	NA	NA

Instead of deleting variables with NA, but recoding them by introducing a new label “missing”, we generate a new data frame called `nesmiss`. In this data frame, the variable is factorized with a new level “missing”. In order to compaare how the model is changed based on the modifying of the dataset, we fit a new model `fitt.miss` with the variables we selected in our previous final model(`fitt`). The summary results of our `fitt.miss` model shows that the main effects of `income` and `race`, and the interaction terms of income and party, income and gender, gender and race are no longer significant.