nes96

Heejung Shim

Consider an example drawn from a subset of the 1996 American National Election Study. We consider only the age (age), education level(educ), and income group (income) as predictors. Our response variable will be party identification (PID): Democrat, Independent, or Republican. The original data involved more than three categories; we collapse this to three.

```
library(faraway)
data(nes96)
?nes96
str(nes96)
```

```
## 'data.frame':    944 obs. of  10 variables:
##  $ popul : int  0 190 31 83 640 110 100 31 180 2800 ...
##  $ TVnews: int  7 1 7 4 7 3 7 1 7 0 ...
##  $ selfLR: Ord.factor w/ 7 levels "extLib"<"Lib"<..: 7 3 2 3 5 3 5 5 4 3 ...
##  $ ClinLR: Ord.factor w/ 7 levels "extLib"<"Lib"<..: 1 3 2 4 6 4 6 4 6 3 ...
##  $ DoleLR: Ord.factor w/ 7 levels "extLib"<"Lib"<..: 6 5 6 5 4 6 4 5 3 7 ...
##  $ PID   : Ord.factor w/ 7 levels "strDem"<"weakDem"<..: 7 2 2 2 1 2 2 5 4 1 ...
##  $ age   : int  36 20 24 28 68 21 77 21 31 39 ...
##  $ educ  : Ord.factor w/ 7 levels "MS"<"HSdrop"<..: 3 4 6 6 6 4 4 4 4 3 ...
##  $ income: Ord.factor w/ 24 levels "$3Kminus"<"$3K-$5K"<..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ vote  : Factor w/ 2 levels "Clinton","Dole": 2 1 1 1 1 1 1 1 1 1 ...
```

```
# The original data involved more than three categories for PID; we collapse this to three.
levels(nes96$PID)
```

```
## [1] "strDem"  "weakDem" "indDem"  "indind"  "indRep"  "weakRep" "strRep"
```

```
sPID <- nes96$PID
levels(sPID) <- c("Democrat","Democrat","Independent","Independent", "Independent","Republican","Republi

# The income variable in the original data was an ordered factor with income ranges.
#Convert this to a numeric variable by taking the midpoint of each range.
levels(nes96$income)
```

```
##  [1] "$3Kminus"   "$3K-$5K"    "$5K-$7K"    "$7K-$9K"    "$9K-$10K"
##  [6] "$10K-$11K"  "$11K-$12K"  "$12K-$13K"  "$13K-$14K"  "$14K-$15K"
## [11] "$15K-$17K"  "$17K-$20K"  "$20K-$22K"  "$22K-$25K"  "$25K-$30K"
## [16] "$30K-$35K"  "$35K-$40K"  "$40K-$45K"  "$45K-$50K"  "$50K-$60K"
## [21] "$60K-$75K"  "$75K-$90K"  "$90K-$105K" "$105Kplus"
```

```
inca <- c(1.5,4,6,8,9.5,10.5,11.5,12.5,13.5,14.5,16,18.5,21,23.5,
          27.5,32.5,37.5,42.5,47.5,55,67.5,82.5,97.5,115)
nincome <- inca[unclass(nes96$income)]
```

Plotting voting preference against education.

```
str(nes96$educ)
```

```
##  Ord.factor w/ 7 levels "MS"<"HSdrop"<..: 3 4 6 6 6 4 4 4 4 3 ...
```

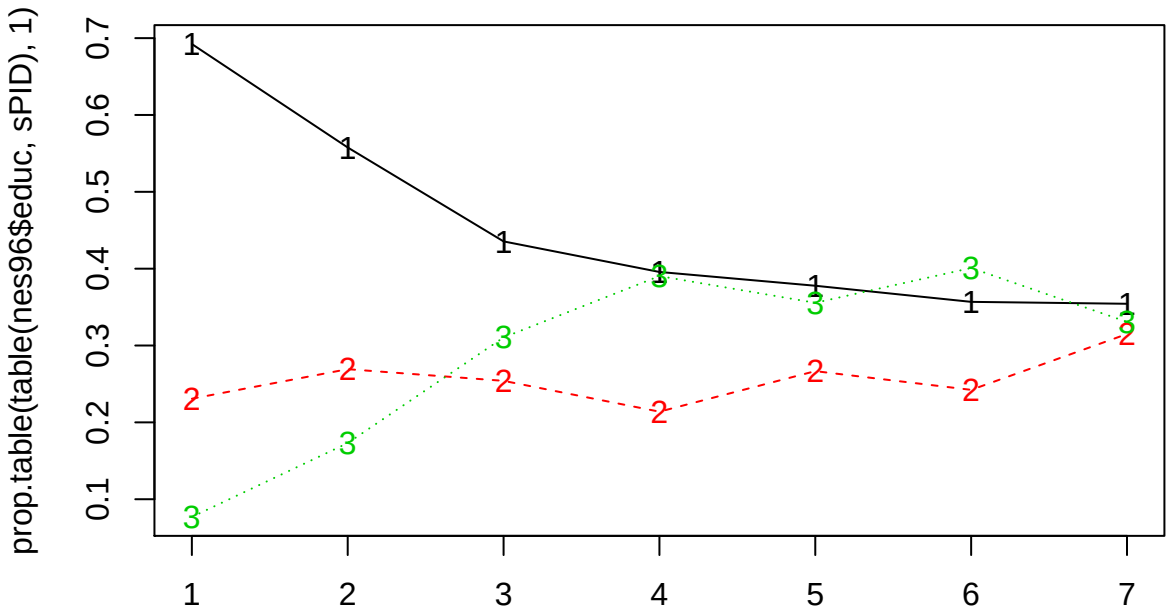
```
table(nes96$educ, sPID)
```

```
##          sPID
##           Democrat Independent Republican
##    MS            9           3          1
##    HSdrop       29          14          9
##    HS          108          63         77
##    Coll         74          40         73
##    CCdeg        34          24         32
##    BAdeg        81          55         91
##    MAdeg        45          40         42
```

```
prop.table(table(nes96$educ, sPID), 1)
```

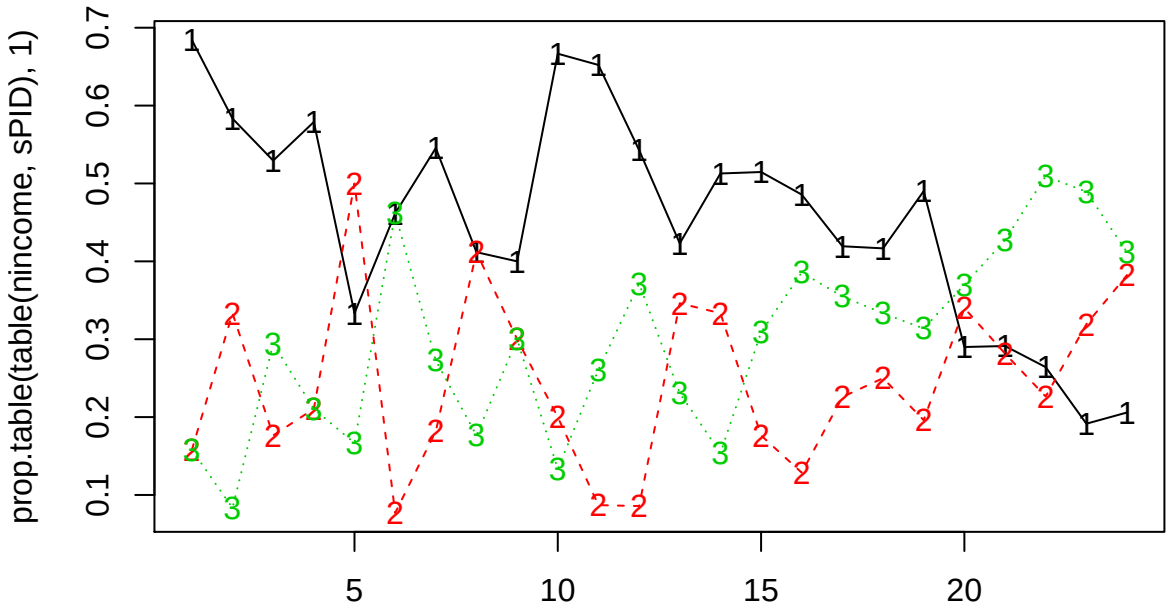
```
##          sPID
##             Democrat Independent Republican
##    MS     0.69230769  0.23076923 0.07692308
##    HSdrop 0.55769231  0.26923077 0.17307692
##    HS     0.43548387  0.25403226 0.31048387
##    Coll   0.39572193  0.21390374 0.39037433
##    CCdeg  0.37777778  0.26666667 0.35555556
##    BAdeg  0.35682819  0.24229075 0.40088106
##    MAdeg  0.35433071  0.31496063 0.33070866
```

```
matplot(prop.table(table(nes96$educ, sPID), 1), type="o")
```



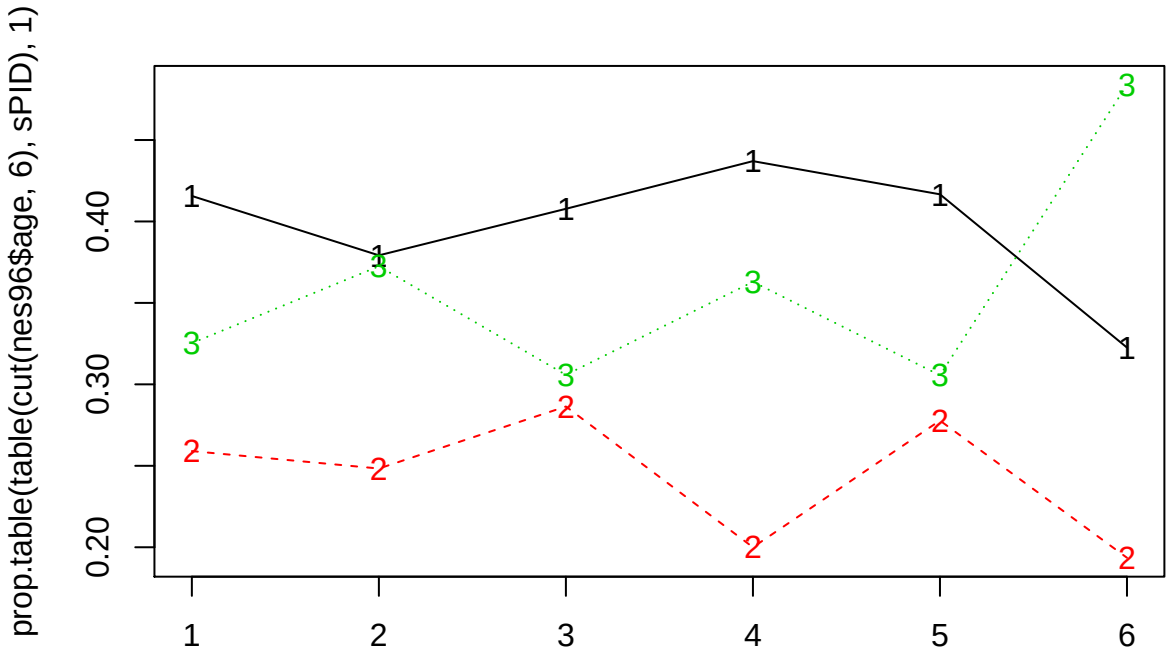
Plotting voting preference against income

```
matplot(prop.table(table(nincome, sPID), 1), type="o")
```



Plotting voting preference agains age; need to group age values. The relationship between sPID to age is not clear.

```
matplot(prop.table(table(cut(nes96$age, 6), sPID), 1), type="o")
```




```
levels(cut(nes96$age, 6))
```

```
## [1] "(18.9,31]" "(31,43]"   "(43,55]"   "(55,67]"   "(67,79]"   "(79,91.1]"
```

Fitting a model

```
library(nnet)
mmod <- multinom(sPID ~ age + educ + nincome, nes96)
```

```
## # weights:  30 (18 variable)
## initial  value 1037.090001
## iter  10 value 990.568608
## iter  20 value 984.319052
## final  value 984.166272
## converged
```

```
summary(mmod)
```

```
## Call:
## multinom(formula = sPID ~ age + educ + nincome, data = nes96)
##
## Coefficients:
##             (Intercept)          age     educ.L     educ.Q    educ.C
## Independent   -1.197260 0.0001534525 0.06351451 -0.1217038 0.1119542
## Republican    -1.642656 0.0081943691 1.19413345 -1.2292869 0.1544575
##                  educ^4     educ^5      educ^6    nincome
## Independent -0.07657336  0.1360851  0.15427826 0.01623911
## Republican  -0.02827297 -0.1221176 -0.03741389 0.01724679
##
## Std. Errors:
##             (Intercept)         age    educ.L    educ.Q    educ.C
## Independent   0.3265951 0.005374592 0.4571884 0.4142859 0.3498491
## Republican    0.3312877 0.004902668 0.6502670 0.6041924 0.4866432
##                educ^4    educ^5    educ^6     nincome
## Independent 0.2883031 0.2494706 0.2171578 0.003108585
## Republican  0.3605620 0.2696036 0.2031859 0.002881745
##
## Residual Deviance: 1968.333
## AIC: 2004.333
```

Model selection using AIC. Model with only income has been selected.

```
mmodi <- step(mmod)
```

```
## Start:  AIC=2004.33
## sPID ~ age + educ + nincome
##
## trying - age
## # weights:  27 (16 variable)
## initial  value 1037.090001
## iter  10 value 988.896864
## iter  20 value 985.822223
## final  value 985.812737
## converged
## trying - educ
## # weights:  12 (6 variable)
## initial  value 1037.090001
```

```
## iter  10 value 992.269502
## final  value 992.269484
## converged
## trying - nincome
## # weights:  27 (16 variable)
## initial  value 1037.090001
## iter  10 value 1009.025560
## iter  20 value 1006.961593
## final  value 1006.955275
## converged
##           Df      AIC
## - educ     6 1996.539
## - age     16 2003.625
## <none>    18 2004.333
## - nincome 16 2045.911
## # weights:  12 (6 variable)
## initial  value 1037.090001
## iter  10 value 992.269502
## final  value 992.269484
## converged
##
## Step:  AIC=1996.54
## sPID ~ age + nincome
##
## trying - age
## # weights:  9 (4 variable)
## initial  value 1037.090001
## final  value 992.712152
## converged
## trying - nincome
## # weights:  9 (4 variable)
## initial  value 1037.090001
## final  value 1020.425203
## converged
##           Df      AIC
## - age      4 1993.424
## <none>     6 1996.539
## - nincome  4 2048.850
## # weights:  9 (4 variable)
## initial  value 1037.090001
## final  value 992.712152
## converged
##
## Step:  AIC=1993.42
## sPID ~ nincome
##
## trying - nincome
## # weights:  6 (2 variable)
## initial  value 1037.090001
## final  value 1020.636052
## converged
##           Df      AIC
## <none>     4 1993.424
## - nincome  2 2045.272
```

```
summary(mmodi)
```

```
## Call:
## multinom(formula = sPID ~ nincome, data = nes96)
##
## Coefficients:
##             (Intercept)    nincome
## Independent  -1.1749331 0.01608683
## Republican   -0.9503591 0.01766457
##
## Std. Errors:
##             (Intercept)     nincome
## Independent   0.1536103 0.002849738
## Republican    0.1416859 0.002652532
##
## Residual Deviance: 1985.424
## AIC: 1993.424
```

Model selection using likelihood ratios. For example, we can fit a model without education and then compare the deviances.The education is not significant. This may seem somewhat surprising given the scatter plot, but the large differences between proportions of Democrats and Republicans occur for groups with low education which represent only a small number of people.

```
mmode <- multinom(sPID ~ age + nincome, nes96)
```

```
## # weights:  12 (6 variable)
## initial  value 1037.090001
## iter  10 value 992.269502
## final  value 992.269484
## converged
```

```
deviance(mmode) - deviance(mmod)
```

```
## [1] 16.20642
```

```
pchisq(16.206, mmod$edf - mmode$edf, lower=FALSE)
```

```
## [1] 0.181982
```

Obtain predicted values for specified values of income. For example, suppose we pick the midpoints of the income groups we selected for the earlier plot.

```
predict(mmodi, data.frame(nincome=inca), type="probs")
```

```
##     Democrat Independent Republican
## 1  0.5836466   0.1846557  0.2316977
## 2  0.5733047   0.1888271  0.2378682
## 3  0.5649837   0.1921708  0.2428455
## 4  0.5566253   0.1955183  0.2478565
## 5  0.5503347   0.1980300  0.2516353
## 6  0.5461317   0.1997045  0.2541638
## 7  0.5419219   0.2013787  0.2566993
```

```
## 8  0.5377060   0.2030524  0.2592415
## 9  0.5334846   0.2047254  0.2617901
## 10 0.5292582   0.2063972  0.2643446
## 11 0.5229106   0.2089023  0.2681871
## 12 0.5123151   0.2130684  0.2746165
## 13 0.5017076   0.2172194  0.2810730
## 14 0.4910976   0.2213511  0.2875513
## 15 0.4741402   0.2279116  0.2979482
## 16 0.4530281   0.2360027  0.3109692
## 17 0.4320800   0.2439428  0.3239772
## 18 0.4113683   0.2517021  0.3369297
## 19 0.3909623   0.2592525  0.3497852
## 20 0.3610676   0.2701312  0.3688012
## 21 0.3136199   0.2868931  0.3994870
## 22 0.2614599   0.3044513  0.4340888
## 23 0.2152314   0.3190178  0.4657508
## 24 0.1691487   0.3322310  0.4986204
```

```
matplot(predict(mmodi, data.frame(nincome=inca), type="probs"), type="l", ylim=c(0, .7))
matplot(prop.table(table(nincome, sPID), 1), add=TRUE)
```