

Problem_set_5

Vinoth Aryan Nagabosshanam

October 27, 2017

SOLUtion For 1.A

```
library(foreign)
library(nnet)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.3
```

```
library(VGAM)
```

```
## Loading required package: stats4
```

```
## Loading required package: splines
```

```
library(reshape2)
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 3.3.3
```

```
library(arm)
```

```
## Warning: package 'arm' was built under R version 3.3.3
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 3.3.3
```

```
## Loading required package: lme4
```

```
## Warning: package 'lme4' was built under R version 3.3.3
```

```
##
```

```
## arm (Version 1.9-3, built: 2016-11-21)
```

```
## Working directory is C:/Users/admin/Desktop/EDA/assignment5
```

```
##
```

```
## Attaching package: 'arm'
```

```
## The following object is masked from 'package:VGAM':
```

```
##
```

```
##      logit
```

```
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 3.3.3
```

```
library(ggbiplot)
```

```
## Loading required package: plyr
```

```
## Loading required package: scales
```

```
## Warning: package 'scales' was built under R version 3.3.3
```

```
##
```

```
## Attaching package: 'scales'
```

```

## The following object is masked from 'package:arm':
##
##      rescale
## Loading required package: grid
library(ggbiplot)

#import the dataset
dataset <- read.dta("https://stats.idre.ucla.edu/stat/data/hsbdemo.dta")

#this step which used include a new colume name newprog
dataset$newprog<-relevel(dataset$prog, ref = "academic")

datasettest <- multinom(newprog ~ ses + write, data = dataset)

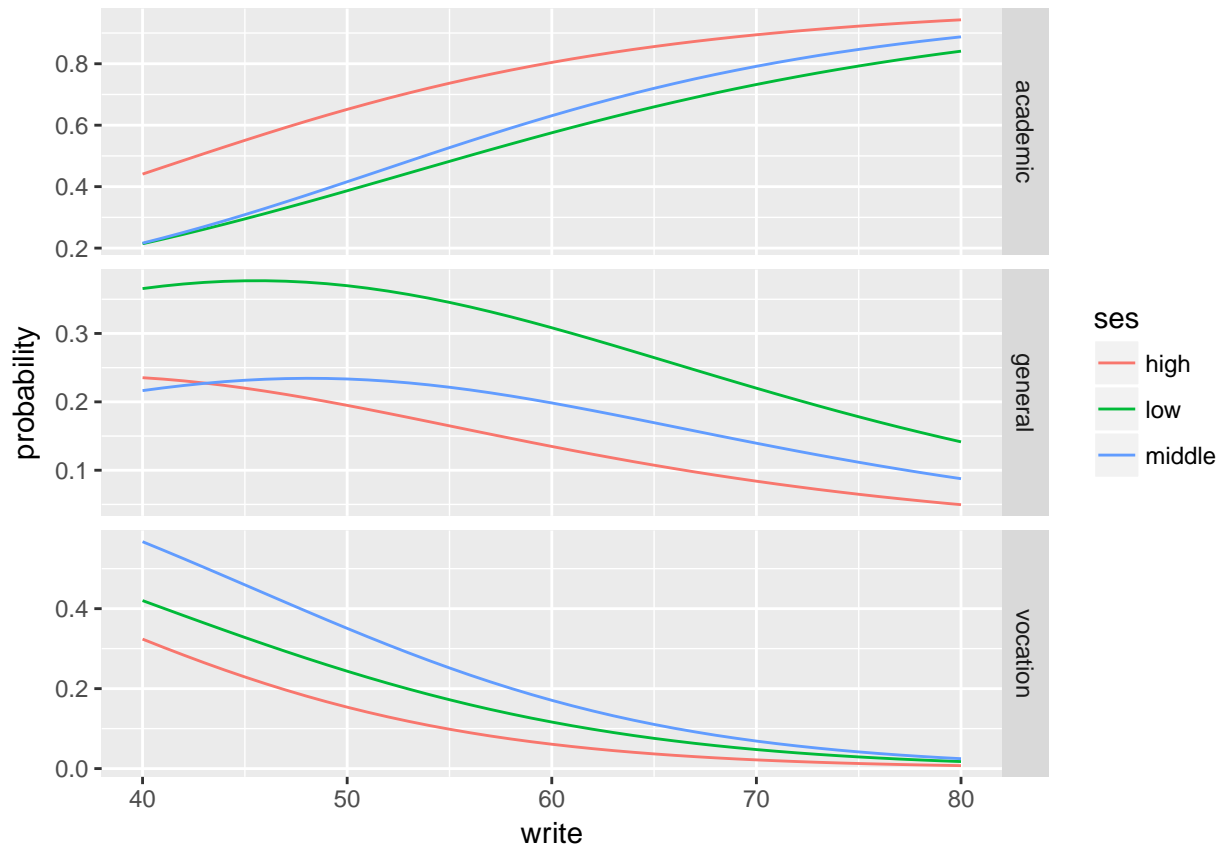
## # weights:  15 (8 variable)
## initial  value 219.722458
## iter   10 value 179.982880
## final   value 179.981726
## converged

hld <- data.frame(ses = c("low", "middle", "high"), write = mean(dataset$write))
writeframe <- data.frame(ses = rep(c("low", "middle", "high"), each = 41), write = rep(c(40:80),3))

# this step which used to predict the probability stores along with respect of write and see

predictwrite <- cbind(writeframe, predict(datasettest, newdata = writeframe, type = "probs", se = TRUE))
## data set melting to long for ggplot2
probplt <- melt(predictwrite, id.vars = c("ses", "write"), value.name = "probability")
ggplot(probplt, aes(x = write, y =probability, colour = ses)) + geom_line() + facet_grid(variable ~., s

```



1.b

```
probabilit<-predict(datasettest,data.frame(ses='middle',write=54),type ="probs")
probabilit
```

```
## academic general vocation
## 0.5049270 0.2247932 0.2702798
```

#ratios

```
atov=probabilit[1]/probabilit[3]
atov
```

```
## academic
## 1.868164
```

```
gtov=probabilit[2]/probabilit[3]
gtov
```

```
## general
## 0.8317055
```

2.solution

```
datasetpolr =polr(factor(prog) ~ ses+write,data =dataset)
display(datasetpolr)
```

```
##
```

```
## Re-fitting to get Hessian
```

```
## polr(formula = factor(prog) ~ ses + write, data = dataset)
```

```
##               coef.est coef.se
## sesmiddle      0.68      0.36
## seshigh        0.40      0.38
## write         -0.04      0.02
## general|academic -3.15      0.87
## academic|vocation -0.71      0.83
## ---
## n = 200, k = 5 (including 2 intercepts)
## residual deviance = 397.0, null deviance is not computed by polr

prob<-predict(datasetpolr,data.frame(ses='middle',write=54),type ="probs")
prob

##   general  academic  vocation
## 0.1780447 0.5357214 0.2862339

3.solution

ke<-dataset[6:10]
datasetpca =prcomp(ke,scale.=TRUE)
datasetpca

## Standard deviations:
## [1] 1.8387006 0.7465777 0.6378031 0.5967980 0.5466638
##
## Rotation:
##           PC1          PC2          PC3          PC4          PC5
## read    0.4664184 -0.02727868  0.5312736731  0.02057541 -0.7064239
## write   0.4483893  0.20754742 -0.8064237887 -0.05575345 -0.3200677
## math    0.4587755 -0.26090184  0.0005952692  0.78003732  0.3361498
## science 0.4355824 -0.61089329  0.0069539237 -0.58947561  0.2992449
## socst   0.4256688  0.71757896  0.2595770518 -0.20131689  0.4426938

ggbiplot(datasetpca, obs.scale = 1, groups = dataset$prog)
```

