**Task 1 Create Binary Response Variables**

* Dichotomize the mental health variable such that the new binary response variable is coded as one (1 = having poor mental health) if the number of days for poor mental health is greater than zero, otherwize it's coded as zero (0 = having excellent mental health). Label this new variable as mntlhlthc2
* Create Dummy Variables from Binary Variables. Create a dummy variable for sex using male as the reference category (hint: for example the new indicator variable can be called female with female coded as one and male coded as zero). Note that you can create a dummy variable for sex using female as the reference category ( (hint: for example the new indicator variable can be called male with male coded as one and female coded as zero). But when one enters the sex variable in a regression model, both dummy variables cannot be entered simultaneously, and one has to be dropped. Please think about why.

> #TASK 1: Create Binary Response Variables

> # 1 = poor mental health mntlhl > 0

> useddta$mntlhc2 <- ifelse(useddta$mntlhlth > 0, 1, 0)

> #Create dummy variables female (male = 0)

> useddta$female <- as.numeric(useddta$sex == "female")

**Task 2 Create Binary Indicator Variables for Multi-Category Nomial Variables**

* Create a set of dummy variables for the race variable. Note that the race variable has three categories, so one can create three dummy variables for race. Please be careful and clear about 1) how many of the three dummy variables, all measuring race, are usually used in a regression model and 2) how to interpret the results/corresponding coefficients (which group is the reference group?).

#Task 2: Create Binary Indicator Variables for Multi-Category Nomial Variables

> #Create dummy variables for race

> useddta$raceF <- factor(useddta$race,

+ levels = c(1,2,3,NA),

+ labels = c("1white", "2black", "3other"))

> table(useddta$raceF,useddta$race, useNA = 'ifany')

iap white black other

1white 0 0 0 0

2black 0 0 0 0

3other 0 0 0 0

<NA> 0 4644 770 292

> # In document discuss how many can be used in the regression model

> # In document discuss how to interpret the coefficient

>

1. Since there are 3 levels, we can use two of the variables in a regression model.
2. Whichever variable is omitted is the reference group.

**Task 3 Graph Bivariate Scatter Plot**

* Drop missing cases
* Draw pairwise bivariate scatter plots (in one graph) and provide a brief description about the sub-graph plotting mntlhlthc2 against educ.

> #Task 3 Graph Bivariate Scatterplot

> #Drop Missing Cases

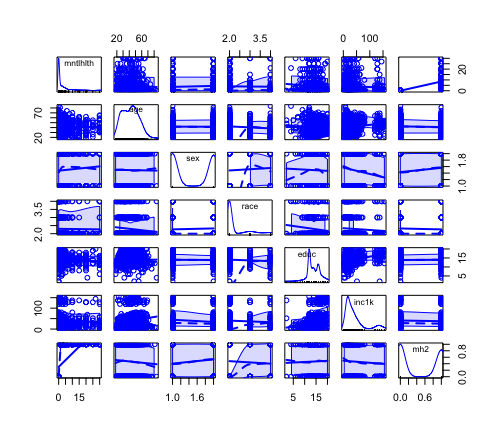
> nmdta <- useddta[complete.cases(useddta),]

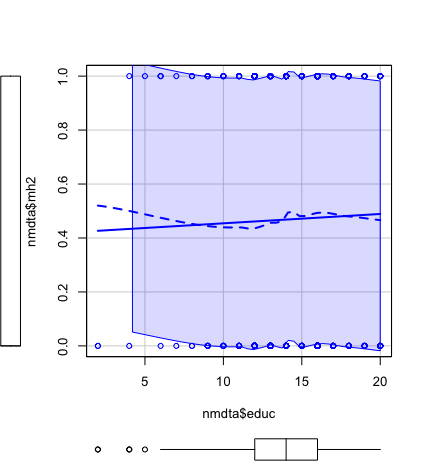
> #Create Scatter plot matrix

> scatterplotMatrix(~ mntlhlth + age + sex + race +

+ educ + inc1k + mntlhc2,

+ smooth = list(span = 0.7), data = useddta)

> scatterplot(nmdta$educ,nmdta$mntlhc2) 



From the scatterplot above, it appears that poor mental health occurs with the same frequency and distribution, regardless of education.

**Task 4 Run Logit**

* Run a logit model of mntlhlthc2 on age (age), sex (sex; male is used as the reference category), race (race; white is used as the reference category), education (educ), and income (inc1k)

> #Task 4 Run Logit

> logit.model <- glm(mntlhc2 ~ age + female + nonwhite + educ + inc1k, family = binomial(link = 'logit'),

+ data = useddta)

> coef(logit.model)

(Intercept) age female nonwhite educ inc1k

-0.211867666 -0.007896290 0.532314849 -0.375736361 0.022785936 -0.002262255

> summary(logit.model)

Call:

glm(formula = mntlhc2 ~ age + female + nonwhite + educ + inc1k,

family = binomial(link = "logit"), data = useddta)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.3750 -1.1019 -0.9097 1.1967 1.6133

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -0.211868 0.473886 -0.447 0.654813

age -0.007896 0.005747 -1.374 0.169414

female 0.532315 0.150556 3.536 0.000407 \*\*\*

nonwhite -0.375736 0.185659 -2.024 0.042991 \*

educ 0.022786 0.029540 0.771 0.440492

inc1k -0.002262 0.002280 -0.992 0.321134

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1036.6 on 749 degrees of freedom

Residual deviance: 1016.6 on 744 degrees of freedom

(4956 observations deleted due to missingness)

AIC: 1028.6

Number of Fisher Scoring iterations: 4

**Tasks 5 Produce and Interpret Odds Ratio (Coefficients)**

* Produce odds ratio coefficients and provide substantive interpretation of the odds ratio coefficients and their precision estimates (i.e., confidence intervals) for sex, education, and race.

> #Task 5 Produce and Interpret Odds Ratio

> exp(logit.model$coefficients)

(Intercept) age female nonwhite educ inc1k

0.8090718 0.9921348 1.7028696 0.6867834 1.0230475 0.9977403

> #Create confidence intervals ... for fun

> exp(confint(logit.model))

2.5 % 97.5 %

(Intercept) 0.3183974 2.046213

age 0.9809774 1.003349

female 1.2688314 2.290031

nonwhite 0.4757934 0.986059

educ 0.9656190 1.084432

inc1k 0.9932471 1.002187

**Odds Ratio Interpretations:**

Being nonwhite versus white decreases the odds of being mentally well by a factor of 0.6868, on average.

Being female versus males increases the odds of being mentally well by a factor of 1.7029, on average.

Holding all variables constant, for each additional year of education, we would expect the odds of being mentally well to increase by a factor or 1.02304.

**Confidence Intervals**

We are 95% confident that the odds ratio coefficient for education in the population to be between 0.9656 and 1.0844.

We are 95% confident that the odds ratio coefficient for nonwhite in the population to be between 0.47579 and 0.9861.

We are 95% confident that the odds ratio coefficient for female in the population to be between 1.269 and 2.29.

**R Script**

# source("/Users/burrisfaculty/Desktop/DSCode/SOC686/Shepherd\_Lab03\_SOC686.R", echo = TRUE, max.deparse.length = 1000)

#Open Log and Read in Data

sink("assign\_03\_shepherd.log")

rm(list=ls(all = TRUE))

setwd("/Users/burrisfaculty/Desktop/DSCode/SOC686")

library(foreign)

library(carData)

library(car)

mygss <- read.dta("gsscum7212teach.dta")

#Select, Tabulate and Summarize Variables for Analysis

usevar <- c("mntlhlth",'age','sex','race','educ','inc1k')

useddta <- mygss[usevar]

table(useddta$mntlhlth, useNA = c("ifany"))

summary(useddta$mntlhlth)

table(useddta$age, useNA = c("ifany"))

summary(useddta$age)

table(useddta$sex, useNA = c("ifany"))

summary(useddta$sex)

table(useddta$race, useNA = c("ifany"))

summary(useddta$race)

table(useddta$educ, useNA = c("ifany"))

summary(useddta$educ)

table(useddta$inc1k, useNA = c("ifany"))

summary(useddta$inc1k)

#TASK 1: Create Binary Response Variables

# 1 = poor mental health mntlhl > 0

useddta$mntlhc2 <- ifelse(useddta$mntlhlth > 0, 1, 0)

#Create dummy variables female (male = 0)

useddta$female <- as.numeric(useddta$sex == "female")

#Task 2: Create Binary Indicator Variables for Multi-Category Nomial Variables

#Create dummy variables for race

useddta$raceF <- factor(useddta$race,

levels = c(1,2,3,NA),

labels = c("1white", "2black", "3other"))

table(useddta$raceF,useddta$race, useNA = 'ifany')

useddta$nonwhite <- as.numeric(useddta$race != 'white')

# In document discuss how many can be used in the regression model

# In document discuss how to interpret the coefficient

#Task 3 Graph Bivariate Scatterplot

#Drop Missing Cases

nmdta <- useddta[complete.cases(useddta),]

#Create Scatter plot matrix

scatterplotMatrix(~ mntlhlth + age + sex + race +

educ + inc1k + mntlhc2,

smooth = list(span = 0.7), data = useddta)

scatterplot(useddta$educ,useddta$mntlhc2)

#Describe scatterplot

#Task 4 Run Logit

logit.model <- glm(mntlhc2 ~ age + female + nonwhite + educ + inc1k, family = binomial(link = 'logit'),

data = useddta)

coef(logit.model)

summary(logit.model)

#Task 5 Produce and Interpret Odds Ratio

exp(logit.model$coefficients)

#Create confidence intervals ... for fun

exp(confint(logit.model))

#Close Out

save(useddta, file = "Assignment\_03.rdata")

sink()

**Log**

> rm(list=ls(all = TRUE))

> setwd("/Users/burrisfaculty/Desktop/DSCode/SOC686")

> library(foreign)

> library(carData)

> library(car)

> mygss <- read.dta("gsscum7212teach.dta")

> #Select, Tabulate and Summarize Variables for Analysis

> usevar <- c("mntlhlth",'age','sex','race','educ','inc1k')

> useddta <- mygss[usevar]

> table(useddta$mntlhlth, useNA = c("ifany"))

0 1 2 3 4 5 6 7 8 10 12 14 15 16 18 20 21 25 27 30 <NA>

401 34 62 37 29 39 6 19 2 35 3 4 22 1 2 21 2 9 1 23 4954

> summary(useddta$mntlhlth)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

0.00 0.00 0.00 3.98 5.00 30.00 4954

> table(useddta$age, useNA = c("ifany"))

18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39

27 92 91 93 80 119 111 123 114 125 151 124 118 117 143 126 136 114 131 107 121 91

40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61

102 111 95 123 115 83 112 89 99 117 99 91 98 76 80 80 73 72 74 69 81 70

62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83

77 78 71 74 58 72 67 71 62 49 58 49 54 37 37 43 46 25 21 26 9 23

84 85 86 87 88 89 <NA>

22 21 16 14 10 35 18

> summary(useddta$age)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

18.00 31.00 43.00 45.57 59.00 89.00 18

> table(useddta$sex, useNA = c("ifany"))

male female

2480 3226

> summary(useddta$sex)

male female

2480 3226

> table(useddta$race, useNA = c("ifany"))

iap white black other

0 4644 770 292

> summary(useddta$race)

iap white black other

0 4644 770 292

> table(useddta$educ, useNA = c("ifany"))

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 <NA>

20 7 15 25 33 30 85 90 251 213 216 350 1817 479 580 249 679 167 189 91 102 18

> summary(useddta$educ)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

0.0 12.0 12.0 12.7 15.0 20.0 18

> table(useddta$inc1k, useNA = c("ifany"))

0.245000049471855 0.25900000333786 0.267749965190887 0.284249991178513 0.301900029182434 0.312849968671799

6 3 2 7 6 1

0.345000028610229 0.363000065088272 0.382000058889389 0.444000065326691 0.482999950647354 0.510000050067902

1 4 1 1 2 1

0.550000071525574 0.602999866008759 0.904999792575836 0.962999880313873 0.980000197887421 1.03600001335144

2 5 3 2 1 2

1.07099986076355 1.1120001077652 1.13700008392334 1.20760011672974 1.23399996757507 1.25139987468719

1 2 4 4 4 2

1.31000018119812 1.32999980449677 1.37799978256226 1.45000004768372 1.52800023555756 1.57200014591217

4 3 4 3 3 3

1.67099976539612 1.7150000333786 1.81299960613251 1.92999982833862 1.98974978923798 2.00000023841858

3 2 3 2 3 2

2.11100053787231 2.11329984664917 2.18995046615601 2.20100021362305 2.20500040054321 2.27200055122375

5 3 2 4 1 4

2.32699966430664 2.40974974632263 2.41100025177002 2.53800058364868 2.55825018882751 2.67400002479553

3 3 9 4 5 1

2.69500041007996 2.70700025558472 2.7171003818512 2.75099968910217 2.81564974784851 2.84899997711182

1 13 3 5 1 3

2.92499923706055 2.9452497959137 2.99200034141541 3.02099895477295 3.09999847412109 3.10499882698059

3 2 1 4 1 4

3.12675023078918 3.24699878692627 3.26300096511841 3.31584334373474 3.32883048057556 3.36700057983398

8 2 3 1 1 4

3.37264037132263 3.37800002098083 3.43799901008606 3.44135165214539 3.48074817657471 3.50000143051147

1 2 5 4 3 3

3.5369987487793 3.56700110435486 3.61899828910828 3.65699911117554 3.67500066757202 3.69500041007996

2 5 9 3 2 5

3.69525074958801 3.74912452697754 3.75999879837036 3.78900098800659 3.85199952125549 3.88499999046326

8 1 5 3 13 3

3.92470073699951 3.96213483810425 3.9760000705719 3.98800015449524 4.01625156402588 4.06704807281494

4 1 2 4 3 3

4.0740008354187 4.11047840118408 4.17499876022339 4.20200109481812 4.21999979019165 4.2637505531311

10 1 4 8 12 6

4.32199907302856 4.32300090789795 4.34300088882446 4.34620380401611 4.41000080108643 4.44700145721436

5 2 3 1 4 14

4.47800064086914 4.50000047683716 4.5285005569458 4.57300615310669 4.58700037002563 4.59599924087524

3 9 1 1 6 3

4.66200017929077 4.69275188446045 4.71300172805786 4.74999809265137 4.81950187683105 4.87900114059448

4 4 6 6 5 3

4.93499755859375 4.95199823379517 4.96600151062012 4.9870023727417 5.10199928283691 5.10900163650513

15 7 1 8 4 2

5.11199855804443 5.11650037765503 5.16700172424316 5.24200248718262 5.2870020866394 5.30799865722656

6 7 5 6 5 6

5.42499923706055 5.43199872970581 5.43420076370239 5.43800067901611 5.49999809265137 5.51250123977661

10 2 6 3 5 7

5.60599994659424 5.63130235671997 5.7300009727478 5.76599931716919 5.80599880218506 5.8274998664856

2 8 4 1 3 7

5.89500093460083 5.89644050598145 5.98500204086304 6.02437734603882 6.02999925613403 6.0529990196228

4 1 9 10 4 5

6.05927133560181 6.15190982818604 6.20099973678589 6.23559617996216 6.24800157546997 6.26700258255005

1 1 7 1 3 3

6.27299976348877 6.30417394638062 6.33300161361694 6.39562606811523 6.45313119888306 6.49999856948853

3 1 6 19 1 4

6.52499723434448 6.62500190734863 6.63100051879883 6.65299940109253 6.73749876022339 6.74100160598755

9 1 9 6 8 10

6.74680233001709 6.7927508354187 6.79427337646484 6.7979998588562 6.86100101470947 6.87600088119507

1 3 1 4 3 6

6.9580020904541 7.03912782669067 7.03949069976807 7.04162549972534 7.0740008354187 7.11407232284546

2 6 1 1 3 1

7.11829328536987 7.12249708175659 7.13000011444092 7.15299940109253 7.23799991607666 7.36312437057495

1 6 10 3 4 6

7.38400220870972 7.48100280761719 7.49999761581421 7.52099800109863 7.64399862289429 7.65081071853638

3 14 8 2 2 1

7.71156692504883 7.75099802017212 7.78200244903564 7.81687259674072 7.83600234985352 7.83913421630859

1 8 11 17 7 1

7.86015462875366 7.88833808898926 7.91699934005737 7.96249914169312 7.9840030670166 8.12199974060059

1 1 3 8 7 12

8.14200115203857 8.15599727630615 8.25400257110596 8.2664966583252 8.29364585876465 8.30224704742432

7 13 8 1 1 9

8.30799674987793 8.31664657592773 8.34899711608887 8.41749668121338 8.43902206420898 8.5200023651123

4 1 5 7 1 3

8.59500217437744 8.60337543487549 8.63600063323975 8.66699886322021 8.68500423431396 8.69449234008789

15 8 7 4 5 1

8.70187473297119 8.70726299285889 8.84299945831299 8.87659358978271 8.99999713897705 9.04199695587158

5 1 8 1 6 7

9.06238746643066 9.07034301757812 9.14223098754883 9.14299869537354 9.16699981689453 9.17300033569336

1 1 1 8 3 17

9.18749809265137 9.23812294006348 9.24072170257568 9.3040189743042 9.40099716186523 9.47300434112549

6 11 1 1 10 12

9.5 9.5033073425293 9.62625789642334 9.64777278900146 9.71249580383301 9.75203418731689

4 1 1 1 3 1

9.75462055206299 9.81174850463867 9.81900215148926 9.85030937194824 9.86153221130371 9.90500164031982

1 6 7 1 1 5

9.95200347900391 9.96899795532227 9.97226810455322 9.98000431060791 10.0050001144409 10.0112991333008

3 10 1 7 3 1

10.040623664856 10.0774421691895 10.1676263809204 10.1768712997437 10.2203073501587 10.2229976654053

5 1 8 1 1 8

10.2312297821045 10.2455148696899 10.3233404159546 10.387354850769 10.3965711593628 10.4124975204468

1 1 1 1 1 11

10.4359979629517 10.4838199615479 10.504997253418 10.5187711715698 10.5940046310425 10.6593713760376

6 1 9 1 6 13

10.7324876785278 10.7467136383057 10.8060026168823 10.8069925308228 10.8080015182495 10.816065788269

1 1 15 1 6 1

10.8183240890503 10.8472929000854 10.8500032424927 10.8570003509521 10.86243724823 10.892219543457

1 1 7 11 1 1

10.9157829284668 10.9222602844238 10.9857225418091 11.0075044631958 11.0141201019287 11.0360431671143

1 1 1 6 1 1

11.0463190078735 11.0499439239502 11.0514621734619 11.1030035018921 11.111011505127 11.1368961334229

1 1 1 1 1 1

11.1959991455078 11.2050037384033 11.2282056808472 11.2499961853027 11.3173589706421 11.3212461471558

4 7 1 12 1 7

11.3290014266968 11.3793725967407 11.3844528198242 11.4659976959229 11.4900035858154 11.5382747650146

4 9 1 9 9 1

11.6375017166138 11.6599760055542 11.6940622329712 11.7184782028198 11.7318754196167 11.744647026062

6 1 1 1 10 1

11.7609996795654 11.7724018096924 11.7810049057007 11.793999671936 11.8189172744751 11.8489255905151

9 1 6 4 1 1

11.875997543335 11.9353685379028 12.063362121582 12.0806198120117 12.1227216720581 12.1687984466553

8 1 1 14 1 1

12.174464225769 12.1969966888428 12.22900390625 12.3024988174438 12.3177843093872 12.3274793624878

1 4 5 9 1 1

12.3418779373169 12.3565406799316 12.3810052871704 12.4090557098389 12.4149980545044 12.467999458313

1 1 5 1 13 9

12.5199966430664 12.5951814651489 12.71812915802 12.718165397644 12.7560033798218 12.7729969024658

5 1 4 1 6 9

12.7790040969849 12.8147125244141 12.8245306015015 12.8307447433472 12.8385782241821 12.8903274536133

10 1 1 12 1 1

12.9051609039307 12.9180040359497 12.9335851669312 13.008113861084 13.0627012252808 13.123610496521

1 7 1 1 1 1

13.1279163360596 13.2236642837524 13.2319650650024 13.242000579834 13.2690029144287 13.296124458313

1 1 1 6 8 10

13.3172149658203 13.3764915466309 13.4021701812744 13.475004196167 13.4937143325806 13.5018749237061

1 1 1 11 1 13

13.5369958877563 13.5569696426392 13.5629959106445 13.5699949264526 13.5790061950684 13.5939970016479

19 1 13 14 12 12

13.5949954986572 13.6877126693726 13.7353763580322 13.7500028610229 13.7579507827759 13.7665882110596

9 1 1 7 1 1

13.8759098052979 13.8913879394531 13.9312152862549 13.9367027282715 13.942193031311 13.9433364868164

1 1 1 1 1 1

13.9802465438843 14.0149936676025 14.1064586639404 14.1081266403198 14.1310052871704 14.1678438186646

1 12 1 1 9 1

14.2201480865479 14.2450008392334 14.3226051330566 14.3249959945679 14.3402500152588 14.4150056838989

1 8 1 6 7 11

14.4459981918335 14.4520053863525 14.4928455352783 14.5150051116943 14.5333576202393 14.5497217178345

4 8 1 11 1 1

14.5613956451416 14.6274385452271 14.6361169815063 14.6410036087036 14.7262554168701 14.7325210571289

1 1 1 9 10 1

14.7380018234253 14.7919321060181 14.8256988525391 14.842604637146 14.8603801727295 14.8783044815063

3 1 2 2 8 1

14.8802194595337 14.9133644104004 14.9317789077759 14.9434328079224 14.9535102844238 14.9568204879761

1 1 1 1 1 1

14.9662227630615 14.9670658111572 14.9963836669922 15.0072135925293 15.0435676574707 15.0733232498169

1 1 1 1 1 1

15.0750045776367 15.1320009231567 15.1443433761597 15.1513795852661 15.1577243804932 15.2433109283447

9 9 1 1 1 1

15.2789974212646 15.3102397918701 15.371994972229 15.3933115005493 15.3972463607788 15.4001235961914

6 1 2 1 1 1

15.4060001373291 15.4105110168457 15.4288196563721 15.4327783584595 15.4431867599487 15.5047388076782

11 1 1 1 1 1

15.5148258209229 15.5412015914917 15.562557220459 15.5770502090454 15.6028003692627 15.6165409088135

1 1 1 1 1 1

15.6189994812012 15.6337518692017 15.6464157104492 15.6680068969727 15.681999206543 15.6861429214478

8 18 1 1 2 1

15.6934299468994 15.7363815307617 15.7455244064331 15.7537403106689 15.788649559021 15.7930011749268

1 1 1 1 1 12

15.7959833145142 15.8189430236816 15.8254156112671 15.8873558044434 15.8884925842285 15.9195852279663

1 1 1 1 1 1

15.9250059127808 15.9518337249756 15.9700231552124 15.9774570465088 16.0170631408691 16.0348987579346

10 1 1 1 1 1

16.0380020141602 16.0444889068604 16.0513916015625 16.0901050567627 16.1358375549316 16.16943359375

8 1 1 1 1 1

16.2212677001953 16.2288188934326 16.2349948883057 16.2500057220459 16.2839946746826 16.3238620758057

2 1 13 2 10 1

16.3466529846191 16.3629989624023 16.3729095458984 16.5113830566406 16.5267601013184 16.5409660339355

1 5 1 1 1 1

16.5630016326904 16.5729560852051 16.5770034790039 16.5806713104248 16.6045017242432 16.6329975128174

5 1 13 1 11 6

16.6653881072998 16.6734981536865 16.6759948730469 16.6890239715576 16.6940364837646 16.6967926025391

1 1 10 1 1 1

16.7022552490234 16.7030048370361 16.7045650482178 16.7278881072998 16.7489585876465 16.7590560913086

1 4 1 1 1 1

16.7656421661377 16.7711448669434 16.8059043884277 16.8350028991699 16.8415222167969 16.8420677185059

1 1 1 12 1 1

16.8458442687988 16.8572044372559 16.8624782562256 16.8728866577148 16.9031181335449 16.9279270172119

2 1 1 1 1 1

16.9358081817627 16.9552974700928 16.9650993347168 16.9738864898682 16.991231918335 16.9940032958984

1 1 1 1 1 8

17.022876739502 17.0359973907471 17.0379981994629 17.0463676452637 17.0941314697266 17.098518371582

1 9 11 1 1 1

17.1064758300781 17.1100482940674 17.1155815124512 17.1539993286133 17.1830291748047 17.2067584991455

1 1 1 8 1 13

17.2102546691895 17.2744312286377 17.3249340057373 17.334997177124 17.345516204834 17.3792285919189

1 2 1 16 1 1

17.3800563812256 17.3940010070801 17.3962249755859 17.4037418365479 17.4903964996338 17.4913806915283

1 7 1 13 1 1

17.5031795501709 17.5193099975586 17.5435199737549 17.5569438934326 17.5692863464355 17.5719528198242

1 1 1 1 1 1

17.5846424102783 17.6128883361816 17.7022228240967 17.7065296173096 17.7372379302979 17.7570056915283

1 1 1 1 1 7

17.764289855957 17.8056564331055 17.8290901184082 17.8494205474854 17.8696022033691 17.8839912414551

1 1 1 1 1 9

17.8909854888916 17.8934593200684 17.898868560791 17.9243221282959 17.9540042877197 18.0366535186768

1 1 1 1 15 1

18.0735893249512 18.0808982849121 18.0843296051025 18.0940074920654 18.1110496520996 18.170129776001

1 1 1 4 1 1

18.1758117675781 18.1794357299805 18.1984996795654 18.2695350646973 18.2870025634766 18.3176174163818

1 1 1 1 15 1

18.3214435577393 18.3350067138672 18.3517475128174 18.3606414794922 18.371955871582 18.3750038146973

1 10 1 1 1 11

18.416145324707 18.4482765197754 18.4589939117432 18.471004486084 18.4762535095215 18.4860553741455

1 1 14 8 19 1

18.4974632263184 18.504997253418 18.5451011657715 18.5601940155029 18.5874366760254 18.5899906158447

1 8 1 1 1 1

18.5986423492432 18.6467380523682 18.6473770141602 18.6654376983643 18.6657409667969 18.6704044342041

1 1 1 1 1 1

18.6958293914795 18.6988620758057 18.7081718444824 18.721076965332 18.7217178344727 18.7288970947266

1 1 1 1 1 1

18.7500038146973 18.759859085083 18.7699337005615 18.7825946807861 18.799259185791 18.826530456543

6 1 1 1 1 1

18.8509998321533 18.8752136230469 18.9088344573975 18.9326515197754 18.9469928741455 18.9510040283203

6 2 1 1 21 10

18.9881820678711 19.0156002044678 19.0461444854736 19.0488700866699 19.1110095977783 19.1276187896729

1 1 1 1 11 1

19.1623458862305 19.1666049957275 19.1779594421387 19.1947383880615 19.2011280059814 19.2078876495361

1 1 1 1 1 1

19.2405395507812 19.2615776062012 19.2746257781982 19.3010196685791 19.3048496246338 19.313024520874

1 1 1 1 1 1

19.3374462127686 19.3604011535645 19.4052257537842 19.4235572814941 19.4249992370605 19.4325981140137

1 1 1 1 7 1

19.4656314849854 19.4692344665527 19.508264541626 19.5323162078857 19.5475959777832 19.5869140625

1 1 1 1 1 1

19.5910053253174 19.6072673797607 19.6235046386719 19.6569938659668 19.6599578857422 19.6777782440186

13 1 23 9 1 1

19.6904468536377 19.7129936218262 19.7131080627441 19.746826171875 19.7629699707031 19.7801475524902

1 5 1 1 1 1

19.7810726165771 19.7929992675781 19.8027038574219 19.8089942932129 19.8459987640381 19.8551425933838

1 3 1 1 6 1

19.8573760986328 19.8640823364258 19.869176864624 19.8758678436279 19.9179916381836 19.9380054473877

1 1 1 1 1 13

19.9494915008545 19.9900016784668 20.0086154937744 20.0110015869141 20.0221118927002 20.0458030700684

1 1 1 12 1 1

20.0623645782471 20.0726623535156 20.0812568664551 20.0863914489746 20.0925807952881 20.1206321716309

1 1 8 1 1 1

20.1423473358154 20.1862678527832 20.2226295471191 20.2424068450928 20.3110332489014 20.335241317749

1 1 1 1 1 20

20.345308303833 20.3549938201904 20.3640403747559 20.3736763000488 20.378963470459 20.3900241851807

1 10 1 1 1 1

20.4072208404541 20.4086799621582 20.411678314209 20.4129428863525 20.4193305969238 20.4847869873047

1 1 1 1 1 1

20.4877948760986 20.502592086792 20.5069923400879 20.5407752990723 20.6013946533203 20.609058380127

1 1 5 1 1 1

20.6349983215332 20.6762866973877 20.6899375915527 20.7425098419189 20.7700042724609 20.7741451263428

8 1 1 1 8 1

20.8232555389404 20.8870410919189 20.9181365966797 20.9325866699219 20.9417304992676 20.9775505065918

1 1 1 1 1 1

20.9807510375977 20.9912986755371 21.0100040435791 21.0206069946289 21.0679893493652 21.0906867980957

1 1 20 1 9 1

21.0908889770508 21.1080303192139 21.1693572998047 21.2163276672363 21.2500019073486 21.298999786377

1 1 1 1 12 4

21.3061504364014 21.3187522888184 21.3319721221924 21.4025592803955 21.4383697509766 21.4469184875488

1 23 1 1 1 1

21.4923725128174 21.6119937896729 21.6126136779785 21.6150035858154 21.6427898406982 21.6589984893799

1 13 1 16 1 5

21.6689968109131 21.6776580810547 21.6932926177979 21.7069702148438 21.7339191436768 21.7655124664307

13 1 1 1 2 1

21.7931346893311 21.8101863861084 21.8915176391602 21.9368877410889 21.9657573699951 22.011157989502

1 1 1 1 1 1

22.0304164886475 22.0319900512695 22.0394725799561 22.0476722717285 22.0500049591064 22.0742645263672

1 4 1 1 17 1

22.1800479888916 22.1951160430908 22.1958332061768 22.205997467041 22.2578792572021 22.2925891876221

1 1 1 15 1 1

22.3486385345459 22.3920097351074 22.4174137115479 22.431999206543 22.4395523071289 22.5441856384277

1 24 1 9 1 1

22.5933647155762 22.6050033569336 22.6249103546143 22.6264209747314 22.6425018310547 22.7713718414307

1 8 1 1 13 1

22.777214050293 22.8018836975098 22.9189987182617 22.9658203125 22.9799957275391 23.035924911499

1 1 7 1 7 1

23.0743370056152 23.0836486816406 23.3099994659424 23.3379821777344 23.3859958648682 23.4032211303711

1 1 10 1 11 1

23.4637603759766 23.563009262085 23.5880107879639 23.6295051574707 23.6875820159912 23.7377853393555

13 26 12 1 1 1

23.7499904632568 23.8359203338623 23.9472007751465 24.001501083374 24.0383148193359 24.0975093841553

11 1 1 1 1 20

24.1210765838623 24.1390037536621 24.207010269165 24.2512836456299 24.3062725067139 24.3659896850586

1 10 6 1 1 18

24.3773555755615 24.3907032012939 24.3950061798096 24.4603748321533 24.5429992675781 24.5470027923584

1 1 8 1 10 5

24.5626449584961 24.5973148345947 24.6775550842285 24.6803550720215 24.6838855743408 24.719762802124

1 1 1 1 1 1

24.7441749572754 24.7658004760742 24.7895259857178 24.8300075531006 24.8619499206543 24.8790016174316

1 1 1 20 1 13

24.9012680053711 24.9026679992676 24.9076557159424 24.9192523956299 24.9370098114014 24.9987525939941

1 1 1 1 18 1

25.1022186279297 25.1756286621094 25.2174873352051 25.3855247497559 25.4090423583984 25.4429664611816

1 1 1 1 1 1

25.4617042541504 25.4650077819824 25.5109958648682 25.514087677002 25.5450077056885 25.5631866455078

1 37 21 1 9 1

25.5825042724609 25.5869922637939 25.6107940673828 25.6173667907715 25.6190032958984 25.6412220001221

34 7 1 1 9 1

25.6426639556885 25.6489753723145 25.6531581878662 25.7967758178711 25.8370056152344 25.8568477630615

1 1 1 1 22 1

25.8781127929688 26.0223693847656 26.0379333496094 26.1136817932129 26.1331634521484 26.1382732391357

1 1 1 1 1 1

26.1417388916016 26.202615737915 26.2186870574951 26.2270164489746 26.3845119476318 26.4109954833984

1 1 1 1 1 11

26.4839897155762 26.5379943847656 26.726526260376 26.7638416290283 26.8955631256104 26.9499950408936

11 12 1 1 1 8

27.0180358886719 27.0659008026123 27.1469917297363 27.1579971313477 27.1710033416748 27.1880073547363

1 1 1 11 23 23

27.3516654968262 27.4999923706055 27.5047912597656 27.5781517028809 27.6257171630859 27.7155456542969

1 13 1 1 1 1

27.7910308837891 27.7929916381836 27.8078117370605 27.9018249511719 28.0290107727051 28.0300025939941

1 36 1 1 9 12

28.1565113067627 28.3015365600586 28.3230018615723 28.4899883270264 28.6500053405762 28.7432460784912

23 1 13 17 18 1

28.8299980163574 28.9983959197998 29.0328750610352 29.2208156585693 29.3102951049805 29.377233505249

9 1 1 1 1 1

29.4020118713379 29.4524974822998 29.4750061035156 29.6023635864258 29.6143360137939 29.7988510131836

14 22 7 1 1 1

29.8067512512207 29.925012588501 29.9440536499023 30.149995803833 30.1614971160889 30.2814235687256

1 22 1 14 1 1

30.3802051544189 30.4580097198486 30.6559371948242 30.6709403991699 30.6760005950928 30.8410015106201

1 19 1 1 1 32

30.8632469177246 30.9091663360596 31.004997253418 31.0110607147217 31.1343631744385 31.1781902313232

1 1 29 1 1 1

31.2674903869629 31.2989940643311 31.3360004425049 31.364013671875 31.4041194915771 31.6605682373047

35 12 9 14 1 1

31.6679973602295 31.7129077911377 31.7424392700195 31.9319610595703 31.9431991577148 31.9510612487793

21 1 1 1 1 1

31.9796257019043 32.0175857543945 32.0702095031738 32.1000137329102 32.1555938720703 32.3958358764648

1 1 1 13 1 1

32.4999923706055 32.506160736084 32.5317802429199 32.5364303588867 32.6249847412109 32.6574211120605

11 1 1 1 32 1

32.7101936340332 32.760986328125 32.9048614501953 33.0160102844238 33.0244522094727 33.075008392334

1 32 1 21 1 9

33.1250114440918 33.1252632141113 33.2089881896973 33.2660102844238 33.4096870422363 33.4123954772949

13 1 18 7 1 1

33.7737693786621 33.8190498352051 33.8344421386719 33.9260063171387 34.0171089172363 34.0779914855957

1 1 1 11 1 21

34.2394065856934 34.3800086975098 34.413501739502 34.5543823242188 34.7036552429199 34.7879867553711

1 35 27 1 1 8

34.9650001525879 35.1210021972656 35.3273620605469 35.3700065612793 35.6509895324707 36.1140174865723

12 10 1 11 16 10

36.1462669372559 36.1669883728027 36.1889991760254 36.2548408508301

25 22 8 1

[ reached getOption("max.print") -- omitted 135 entries ]

> summary(useddta$inc1k)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.245 12.481 22.605 30.279 37.226 162.607

> #TASK 1: Create Binary Response Variables

> # 1 = poor mental health mntlhl > 0

> useddta$mntlhc2 <- ifelse(useddta$mntlhlth > 0, 1, 0)

> #Create dummy variables female (male = 0)

> useddta$female <- as.numeric(useddta$sex == "female")

> #Task 2: Create Binary Indicator Variables for Multi-Category Nomial Variables

> #Create dummy variables for race

> useddta$raceF <- factor(useddta$race,

+ levels = c(1,2,3,NA),

+ labels = c("1white", "2black", "3other"))

> table(useddta$raceF,useddta$race, useNA = 'ifany')

iap white black other

1white 0 0 0 0

2black 0 0 0 0

3other 0 0 0 0

<NA> 0 4644 770 292

> useddta$nonwhite <- as.numeric(useddta$race != 'white')

> # In document discuss how many can be used in the regression model

> # In document discuss how to interpret the coefficient

>

> #Task 3 Graph Bivariate Scatterplot

> #Drop Missing Cases

> nmdta <- useddta[complete.cases(useddta),]

> #Create Scatter plot matrix

> scatterplotMatrix(~ mntlhlth + age + sex + race +

+ educ + inc1k + mntlhc2,

+ smooth = list(span = 0.7), data = useddta)

> scatterplot(useddta$educ,useddta$mntlhc2)

> #Describe scatterplot

>

> #Task 4 Run Logit

> logit.model <- glm(mntlhc2 ~ age + female + nonwhite + educ + inc1k, family = binomial(link = 'logit'),

+ data = useddta)

> coef(logit.model)

(Intercept) age female nonwhite educ inc1k

-0.211867666 -0.007896290 0.532314849 -0.375736361 0.022785936 -0.002262255

> summary(logit.model)

Call:

glm(formula = mntlhc2 ~ age + female + nonwhite + educ + inc1k,

family = binomial(link = "logit"), data = useddta)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.3750 -1.1019 -0.9097 1.1967 1.6133

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -0.211868 0.473886 -0.447 0.654813

age -0.007896 0.005747 -1.374 0.169414

female 0.532315 0.150556 3.536 0.000407 \*\*\*

nonwhite -0.375736 0.185659 -2.024 0.042991 \*

educ 0.022786 0.029540 0.771 0.440492

inc1k -0.002262 0.002280 -0.992 0.321134

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1036.6 on 749 degrees of freedom

Residual deviance: 1016.6 on 744 degrees of freedom

(4956 observations deleted due to missingness)

AIC: 1028.6

Number of Fisher Scoring iterations: 4

> #Task 5 Produce and Interpret Odds Ratio

> exp(logit.model$coefficients)

(Intercept) age female nonwhite educ inc1k

0.8090718 0.9921348 1.7028696 0.6867834 1.0230475 0.9977403

> #Create confidence intervals ... for fun

> exp(confint(logit.model))

2.5 % 97.5 %

(Intercept) 0.3183974 2.046213

age 0.9809774 1.003349

female 1.2688314 2.290031

nonwhite 0.4757934 0.986059

educ 0.9656190 1.084432

inc1k 0.9932471 1.002187

> #Close Out

> save(useddta, file = "Assignment\_03.rdata")

> sink()