```
##Question 1
##1. converted excel file to csv file.

> gardasil <- read.csv("C:/Users/James/Desktop/gardasil.csv")

## Re-coded the variables.
gardasil$AgeGroup <- factor(gardasil$AgeGroup, labels=c("18-26", "11-17"),
levels=1:0)
gardasil$Race <- factor(gardasil$Race, labels=c("white", "black", "hispanic",
"other/unknown"), levels=0:3)
gardasil$Completed <- factor(gardasil$Completed, labels=c("no", "yes"),
levels=0:1)
gardasil$InsuranceType <- factor(gardasil$InsuranceType, labels=c("medical
assistance", "private payer", "hospital based", "military"), levels=0:3)
gardasil$MedAssist <- factor(gardasil$MedAssist, labels=c("yes", "no"),
levels=1:0)
gardasil$Location <- factor(gardasil$Location, labels=c("Odenton", "White
Marsh", "Johns Hopkins Outpatient Center", "Bayview"), levels=1:4)
gardasil$LocationType <- factor(gardasil$LocationType, labels=c("urban",
"suburban"), levels=1:0)
gardasil$PracticeType <- factor(gardasil$PracticeType, labels=c("pediatric",
"family practice", "OB-GYN"), levels=0:2)

##Finding frequencies for first part of Table 2, Chou et al.
## AgeGroup x Completed.

> summary(gardasil$AgeGroup)
18-26 11-17
  712   701

## Creating 2x2 Table.
> sn.tab <- table(gardasil$AgeGroup, gardasil$Completed)
> sn.tab
         no yes
  18-26 490 222
  11-17 454 247
## Selecting Values.
> sn.tab[3:4]
[1] 222 247

## Finding proportions.
> prop.table(sn.tab, 1)

               no       yes
  18-26 0.6882022 0.3117978
  11-17 0.6476462 0.3523538
> proportion<-prop.table(sn.tab, 1)

## Selecting values.
> proportion[3:4]
[1] 0.3117978 0.3523538

## Making a new vector.

> proportionage<-proportion[3:4]

## multiplying by 100 to get percentage.

> univage<-cbind(summary(gardasil$AgeGroup),sn.tab[3:4],proportionage*100)
> univage
      [,1] [,2]     [,3]
18-26  712  222 31.17978
11-17  701  247 35.23538
```

```
##InsuranceType x Completed.
## Repeat previous steps (from AgeGroup x Completed).

## Find frequency (n).
> summary(gardasil$InsuranceType)
medical assistance      private payer     hospital based 
               275                723                 84 
          military 
               331 

## Create 2x2 table
> sn.tabInsuranceType <- table(gardasil$InsuranceType, gardasil$Completed)
> sn.tabInsuranceType

                      no yes
  medical assistance 220  55
  private payer      470 253
  hospital based      45  39
  military           209 122

## Select desired cells.
> sn.tabInsuranceType[5:8]
[1]  55 253  39 122

## Creating percentages.
## First find proportions.
> prop.table(sn.tabInsuranceType, 1)

                            no       yes
  medical assistance 0.8000000 0.2000000
  private payer      0.6500692 0.3499308
  hospital based     0.5357143 0.4642857
  military           0.6314199 0.3685801

> proportioninstype<-prop.table(sn.tabInsuranceType, 1)
> proportioninstype[5:8]
[1] 0.2000000 0.3499308 0.4642857 0.3685801
> proportionins.type<-proportioninstype[5:8]

## Put n, 2x2 table results and percentage together.
> univInsuranceType<-
cbind(summary(gardasil$InsuranceType),sn.tabInsuranceType[5:8],proportionins.
type*100)
> univInsuranceType
                   [,1] [,2]     [,3]
medical assistance  275   55 20.00000
private payer       723  253 34.99308
hospital based       84   39 46.42857
military            331  122 36.85801


## MedAssitst x Completed.
## Repeat the same steps as AgeGroup x Completed.

## Frequency (n).
> summary(gardasil$MedAssist)
 yes   no 
 275 1138 

## 2x2 table.
> sn.tabMedAssist <- table(gardasil$MedAssist, gardasil$Completed)
> sn.tabMedAssist
```

```
       no yes
  yes 220  55
  no  724 414

## Select desired values.
> sn.tabMedAssist[3:4]
[1]  55 414

## Find percentages.
## Find proportions first.

> prop.table(sn.tabMedAssist, 1)

             no       yes
  yes 0.8000000 0.2000000
  no  0.6362039 0.3637961
> proportionmedassist<-prop.table(sn.tabMedAssist, 1)

> proportionmedassist

             no       yes
  yes 0.8000000 0.2000000
  no  0.6362039 0.3637961

## Select vales.
> proportionmedassist[3:4]
[1] 0.2000000 0.3637961
> proportionmed.assist<-proportionmedassist[3:4]
> proportionmed.assist
[1] 0.2000000 0.3637961

#Find percentage.
> proportionmed.assist*100
[1] 20.00000 36.37961

## Put n, 2x2 results and percentage together.
> univMedAssist<-
cbind(summary(gardasil$MedAssist),sn.tabMedAssist[3:4],proportionmed.assist*1
00)
> univMedAssist
    [,1] [,2]     [,3]
yes  275   55 20.00000
no  1138  414 36.37961


## LocationType x Completed.
## Repeat the same steps as AgeGroup x Completed.
> summary(gardasil$LocationType)
   urban suburban
     450      963
> sn.tablocationtype<-table(gardasil$LocationType, gardasil$Completed)
>
> sn.tablocationtype

            no yes
  urban    336 114
  suburban 608 355
> sn.tablocationtype[3:4]
[1] 114 355
> prop.table(sn.tablocationtype, 1)

                  no       yes
  urban    0.7466667 0.2533333
  suburban 0.6313603 0.3686397
```

```
> proportionlocationtype<-prop.table(sn.tablocationtype, 1)
> proportionlocationtype[3:4]
[1] 0.2533333 0.3686397
> proportion.locationtype<-proportionlocationtype[3:4]
> proportion.locationtype*100
[1] 25.33333 36.86397
> univLocationType<-
cbind(summary(gardasil$LocationType),sn.tablocationtype[3:4],proportion.locat
iontype*100)
> univLocationType
         [,1] [,2]     [,3]
urban     450  114 25.33333
suburban  963  355 36.86397

## PracticeType x Completed.
## Repeat the same steps as AgeGroup x Completed.

> summary(gardasil$PracticeType)
      pediatric family practice          OB-GYN
            515             365             533
> sn.tabpracticetype<-table(gardasil$PracticeType, gardasil$Completed)
> sn.tabpracticetype

                   no yes
  pediatric       353 162
  family practice 259 106
  OB-GYN          332 201
> sn.tabpracticetype[4:6]
[1] 162 106 201
> prop.table(sn.tabpracticetype, 1)

                         no       yes
  pediatric       0.6854369 0.3145631
  family practice 0.7095890 0.2904110
  OB-GYN          0.6228893 0.3771107
> proportionpracticetype<-prop.table(sn.tabpracticetype, 1)
> proportionpracticetype[4:6]
[1] 0.3145631 0.2904110 0.3771107
> proportion.practicetype<-proportionpracticetype[4:6]
> proportion.practicetype*100
[1] 31.45631 29.04110 37.71107
> univPracticeType<-
cbind(summary(gardasil$PracticeType),sn.tabpracticetype[4:6],proportion.pract
icetype*100)
> univPracticeType
                [,1] [,2]     [,3]
pediatric        515  162 31.45631
family practice  365  106 29.04110
OB-GYN           533  201 37.71107

## Race x Completed.
## Repeat the same steps as AgeGroup x Completed.

> summary(gardasil$Race)
        white         black      hispanic other/unknown
          732           443            52           186
> sn.tabrace<-table(gardasil$Race, gardasil$Completed)
> sn.tabrace

                 no yes
  white         452 280
  black         338 105
  hispanic       35  17
```

```
  other/unknown 119  67
> sn.tabrace[5:7]
[1] 280 105  17
> prop.table(sn.tabrace, 1)
               
                       no       yes
  white         0.6174863 0.3825137
  black         0.7629797 0.2370203
  hispanic      0.6730769 0.3269231
  other/unknown 0.6397849 0.3602151
> proportionrace<-prop.table(sn.tabrace, 1)
> proportionrace[5:8]
[1] 0.3825137 0.2370203 0.3269231 0.3602151
> proportion.race<-proportionrace[5:8]
> proportion.race*100
[1] 38.25137 23.70203 32.69231 36.02151
> univrace<-cbind(summary(gardasil$Race),sn.tabrace[5:8],proportion.race*100)
> univrace
              [,1] [,2]     [,3]
white          732  280 38.25137
black          443  105 23.70203
hispanic        52   17 32.69231
other/unknown  186   67 36.02151


##Creating a new vector called “gardasil$Race2” that combines “black” and “Hispanic”.
> gardasil$Race2<-gardasil$Race
## Verifying the levels for Race2.
> levels(gardasil$Race2)
[1] "white"         "black"         "hispanic"      "other/unknown"
## Renaming the levels for Race2.
> levels(gardasil$Race2) <-c("white","minority","minority","other/unknown")
## Checking to see that the changes are made.
> levels(gardasil$Race2)
[1] "white"         "minority"      "other/unknown"

## Race2 x Completed.
## Repeat the same steps as AgeGroup x Completed.

> summary(gardasil$Race2)
        white      minority other/unknown 
          732           495           186 
> sn.tabRace2<-table(gardasil$Race2, gardasil$Completed)
> sn.tabRace2
               
                 no yes
  white         452 280
  minority      373 122
  other/unknown 119  67

> sn.tabRace2[4:6]
[1] 280 122
> prop.table(sn.tabRace2, 1)
               
                       no       yes
  white         0.6174863 0.3825137
  minority      0.7535354 0.2464646
  other/unknown 0.6397849 0.3602151
> proportionRace2<-prop.table(sn.tabRace2, 1)

> proportion.Race2<-proportionRace2[4:6]
> proportion.Race2*100
[1] 38.25137 24.64646 36.02151
```

```
> univRace2<-
cbind(summary(gardasil$Race2),sn.tabRace2[4:6],proportion.Race2*100)
> univRace2
              [,1] [,2]     [,3]
white          732  280 38.25137
minority       495  122 24.64646
other/unknown  186   67 36.02151

## Binding all frequencies together to make a table.

> totalncomplpercent <- rbind(univage, univInsuranceType, univLocationType,
univPracticeType, univrace, univRace2)
> totalncomplpercent
                    [,1] [,2]     [,3]
18-26                712  222 31.17978
11-17                701  247 35.23538
medical assistance   275   55 20.00000
private payer        723  253 34.99308
hospital based        84   39 46.42857
military             331  122 36.85801
urban                450  114 25.33333
suburban             963  355 36.86397
pediatric            515  162 31.45631
family practice      365  106 29.04110
OB-GYN               533  201 37.71107
white                732  280 38.25137
black                443  105 23.70203
hispanic              52   17 32.69231
other/unknown        186   67 36.02151
white                732  280 38.25137
minority             495  122 24.64646
other/unknown        186   67 36.02151

##change column names

>colnames(totalncomplpercent) <- c("n","Completed 3 Vaccinations in 12
Mo","(%)")

##change row names

> rownames(totalncomplpercent) <- c("18-26","11-
17","Public","Private","Private payer","Hospital-
based","Military","Urban","Suburban","Pediatrics","Family
practice","Gynecology","White", "Minority","African American","Hispanic")

##convert to csv file to make table look nicer.

> write.csv(totalncomplpercent, "C:/Users/James/Desktop/GardasilTable1.csv")


##Part 2 of Table 1 (Odds Ratios, 95%CI, P):

##Set up race as the exposure and completed as outcome in a new vector.

> logitRace <- glm(Completed ~ Race, data = gardasil, family = "binomial")
## P-values for the association between race and completed.
> summary(logitRace)

Call:
glm(formula = Completed ~ Race, family = "binomial", data = gardasil)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.9819  -0.9819  -0.7356   1.3864   1.6968
```

```
Coefficients:
                  Estimate Std. Error z value Pr(>|z|)    
(Intercept)       -0.47889    0.07605  -6.297 3.04e-10 ***
Raceblack         -0.69019    0.13515  -5.107 3.28e-07 ***
Racehispanic      -0.24324    0.30525  -0.797    0.426    
Raceother/unknown -0.09554    0.17062  -0.560    0.576    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1796  on 1412  degrees of freedom
Residual deviance: 1768  on 1409  degrees of freedom
AIC: 1776

Number of Fisher Scoring iterations: 4


##Finding ORs, 95% CIs and extracting p=values from summary(logitRace).

> cbind(OR=exp(coef(logitRace)), exp(confint(logitRace)),
p=summary(logitRace)$coef[, 4])
Waiting for profiling to be done...
                         OR     2.5 %    97.5 %            p
(Intercept)       0.6194690 0.5331816 0.7184667 3.035432e-10
Raceblack         0.5014793 0.3837039 0.6519972 3.277065e-07
Racehispanic      0.7840816 0.4215722 1.4058708 4.255341e-01
Raceother/unknown 0.9088836 0.6481777 1.2663418 5.755240e-01

## Giving odds ratios, 95%CI and p-values a new vector name.

> univraceor95cip<-cbind(OR=exp(coef(logitRace)), exp(confint(logitRace)),
p=summary(logitRace)$coef[, 4])
Waiting for profiling to be done...



##Set up age group as the exposure and completed as outcome in a new vector.

> logitAgeGroup <- glm(Completed ~ AgeGroup, data = gardasil, family =
"binomial")

## P-values for the association between race and completed.
> summary(logitAgeGroup)

Call:
glm(formula = Completed ~ AgeGroup, family = "binomial", data = gardasil)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-0.9321  -0.9321  -0.8645   1.4444   1.5267  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)    -0.7917     0.0809  -9.786   <2e-16 ***
AgeGroup11-17   0.1830     0.1131   1.618    0.106    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1796.0  on 1412  degrees of freedom
Residual deviance: 1793.4  on 1411  degrees of freedom
```

```
AIC: 1797.4

Number of Fisher Scoring iterations: 4


## Finding OR, 95% confidence interval, and extracting p-value from summary.
> cbind(OR=exp(coef(logitAgeGroup)), exp(confint(logitAgeGroup)),
p=summary(logitAgeGroup)$coef[, 4])
Waiting for profiling to be done...
                     OR     2.5 %    97.5 %            p
(Intercept)   0.4530612 0.3859824 0.5301187 1.291527e-22
AgeGroup11-17 1.2008374 0.9621912 1.4993632 1.056853e-01

## Giving the OR, the 95% CI and p-value a new vector name.

> univageor95cip<-cbind(OR=exp(coef(logitAgeGroup)),
exp(confint(logitAgeGroup)), p=summary(logitAgeGroup)$coef[, 4])
Waiting for profiling to be done...


##Set up insurance type as the exposure and completed as outcome in a new
vector.
> logitInsuranceType <- glm(Completed ~ InsuranceType, data = gardasil,
family = "binomial")

##Finding the OR, 95% CI and extracting p-value from
summary(logitInsuranceType).

> cbind(OR=exp(coef(logitInsuranceType)), exp(confint(logitInsuranceType)),
p=summary(logitInsuranceType)$coef[, 4])
Waiting for profiling to be done...
                                  OR     2.5 %    97.5 %
(Intercept)                 0.250000 0.1843109 0.3332127
InsuranceTypeprivate payer  2.153191 1.5533828 3.0247079
InsuranceTypehospital based 3.466667 2.0598141 5.8495818
InsuranceTypemilitary       2.334928 1.6191330 3.3995684
                                       p
(Intercept)                 3.727732e-20
InsuranceTypeprivate payer  6.221592e-06
InsuranceTypehospital based 2.880877e-06
InsuranceTypemilitary       7.208165e-06

## Giving the OR, 95% CI and p-value a new vector name.

> univinsurtypeor95cip<-cbind(OR=exp(coef(logitInsuranceType)),
exp(confint(logitInsuranceType)), p=summary(logitInsuranceType)$coef[, 4])
Waiting for profiling to be done...

##Set up medical assistance as the exposure and completed as outcome in a new
vector.
> logitMedAssist <- glm(Completed ~ MedAssist, data = gardasil, family =
"binomial")

## Finding the odds ratio, 95% CI and extracting p-value from
summary(logitMedAssist).

> cbind(OR=exp(coef(logitMedAssist)), exp(confint(logitMedAssist)),
p=summary(logitMedAssist)$coef[, 4])
Waiting for profiling to be done...
                  OR     2.5 %    97.5 %            p
(Intercept) 0.250000 0.1843109 0.3332127 3.727732e-20
MedAssistno 2.287293 1.6736972 3.1725619 3.770511e-07

## Giving the odds ratio, 95% CI, and p-value a new vector name.
```

```
> univmedassistor95cip<-cbind(OR=exp(coef(logitMedAssist)),
exp(confint(logitMedAssist)), p=summary(logitMedAssist)$coef[, 4])
Waiting for profiling to be done...

##Set up location type as the exposure and completed as outcome in a new
vector.
> logitLocationType <- glm(Completed ~ LocationType, data = gardasil, family
= "binomial")

## Finding odds ratio, 95% CI and extracting p-value from
summary(logitLocationType).

> cbind(OR=exp(coef(logitLocationType)), exp(confint(logitLocationType)),
p=summary(logitLocationType)$coef[, 4])
Waiting for profiling to be done...
                            OR    2.5 %    97.5 %            p
(Intercept)          0.3392857 0.273272 0.4181235 2.009737e-23
LocationTypesuburban 1.7209141 1.343982 2.2145694 2.009836e-05

##Giving the odds ratio, 95% CI and p-value a new vector name.

> univlocattypeor95cip<-cbind(OR=exp(coef(logitLocationType)),
exp(confint(logitLocationType)), p=summary(logitLocationType)$coef[, 4])
Waiting for profiling to be done...

##Set up practice type as the exposure and completed as outcome in a new
vector.
> logitPracticeType<-glm(Completed ~ PracticeType, data = gardasil, family =
"binomial")

## Finding the odds ratios, 95% CIs, and extracting p-values from
summary(logitPracticeType).

> cbind(OR=exp(coef(logitPracticeType)), exp(confint(logitPracticeType)),
p=summary(logitPracticeType)$coef[, 4])
Waiting for profiling to be done...
                                   OR     2.5 %    97.5 %
(Intercept)                 0.4589235 0.3801637 0.5516303
PracticeTypefamily practice 0.8917966 0.6645241 1.1937630
PracticeTypeOB-GYN          1.3192213 1.0222206 1.7043677
                                       p
(Intercept)                 2.259682e-16
PracticeTypefamily practice 4.431689e-01
PracticeTypeOB-GYN          3.356505e-02

## Giving the odds ratios, 95% CIs, p-values a new vector name.

> univpracticetypeor95cip<-cbind(OR=exp(coef(logitPracticeType)),
exp(confint(logitPracticeType)), p=summary(logitPracticeType)$coef[, 4])
Waiting for profiling to be done...

##Set up Race as the exposure and completed as outcome in a new vector.
> logitRace<-glm(Completed ~ Race, data = gardasil, family = "binomial")

##Finding odds ratios, 95% CIs, and extracting p-values from
summary(logitRace).

> cbind(OR=exp(coef(logitRace)), exp(confint(logitRace)),
p=summary(logitRace)$coef[, 4])
Waiting for profiling to be done...
                         OR     2.5 %    97.5 %            p
(Intercept)       0.6194690 0.5331816 0.7184667 3.035432e-10
Raceblack         0.5014793 0.3837039 0.6519972 3.277065e-07
```

```
Racehispanic      0.7840816 0.4215722 1.4058708 4.255341e-01
Raceother/unknown 0.9088836 0.6481777 1.2663418 5.755240e-01


##Giving the odds ratios, 95% CIs and p-values a new vector name.

> univraceor95cip<-cbind(OR=exp(coef(logitRace)), exp(confint(logitRace)),
p=summary(logitRace)$coef[, 4])
Waiting for profiling to be done...

##Set up Race 2 (combined black and hispanic) as the exposure and completed
as outcome in a new vector.

> logitRace2<-glm(Completed ~ Race2, data = gardasil, family = "binomial")

##Finding the odds ratios, 95% CIs, and extracting p-values from
summary(logitRace2).

> cbind(OR=exp(coef(logitRace2)), exp(confint(logitRace2)),
p=summary(logitRace2)$coef[, 4])
Waiting for profiling to be done...
                          OR     2.5 %    97.5 %            p
(Intercept)        0.6194690 0.5331816 0.7184667 3.035432e-10
Race2minority      0.5279969 0.4090908 0.6787296 7.503948e-07
Race2other/unknown 0.9088836 0.6481777 1.2663418 5.755240e-01

##Giving the odds ratios, 95% Cis and extracting p-values a new vector name.

> univrace2or95cip<-cbind(OR=exp(coef(logitRace2)), exp(confint(logitRace2)),
p=summary(logitRace2)$coef[, 4])
Waiting for profiling to be done...

## Combining all the newly created vectors that contain ORs, 95% CIs and p-
values all into one table.
> total <- rbind(univageor95cip, univinsurtypeor95cip, univmedassistor95cip,
univlocattypeor95cip, univpracticetypeor95cip, univraceor95cip,
univrace2or95cip)

##Creating table.
> total
                                   OR     2.5 %    97.5 %
(Intercept)                 0.4530612 0.3859824 0.5301187
AgeGroup11-17               1.2008374 0.9621912 1.4993632
(Intercept)                 0.2500000 0.1843109 0.3332127
InsuranceTypeprivate payer  2.1531915 1.5533828 3.0247079
InsuranceTypehospital based 3.4666667 2.0598141 5.8495818
InsuranceTypemilitary       2.3349282 1.6191330 3.3995684
(Intercept)                 0.2500000 0.1843109 0.3332127
MedAssistno                 2.2872928 1.6736972 3.1725619
(Intercept)                 0.3392857 0.2732720 0.4181235
LocationTypesuburban        1.7209141 1.3439819 2.2145694
(Intercept)                 0.4589235 0.3801637 0.5516303
PracticeTypefamily practice 0.8917966 0.6645241 1.1937630
PracticeTypeOB-GYN          1.3192213 1.0222206 1.7043677
(Intercept)                 0.6194690 0.5331816 0.7184667
Raceblack                   0.5014793 0.3837039 0.6519972
Racehispanic                0.7840816 0.4215722 1.4058708
Raceother/unknown           0.9088836 0.6481777 1.2663418
(Intercept)                 0.6194690 0.5331816 0.7184667
Race2minority               0.5279969 0.4090908 0.6787296
Race2other/unknown          0.9088836 0.6481777 1.2663418
                                       p
(Intercept)                 1.291527e-22
AgeGroup11-17               1.056853e-01
```

```
(Intercept)                  3.727732e-20
InsuranceTypeprivate payer   6.221592e-06
InsuranceTypehospital based 2.880877e-06
InsuranceTypemilitary        7.208165e-06
(Intercept)                  3.727732e-20
MedAssistno                  3.770511e-07
(Intercept)                  2.009737e-23
LocationTypesuburban         2.009836e-05
(Intercept)                  2.259682e-16
PracticeTypefamily practice 4.431689e-01
PracticeTypeOB-GYN           3.356505e-02
(Intercept)                  3.035432e-10
Raceblack                    3.277065e-07
Racehispanic                 4.255341e-01
Raceother/unknown            5.755240e-01
(Intercept)                  3.035432e-10
Race2minority                7.503948e-07
Race2other/unknown           5.755240e-01
```

```
## Change the table into a CSV file so that it can be tidied up into a
publishable-looking table using Excel.

> write.csv(total, "C:/Users/James/Desktop/Table2.csv")


## Table 3:Multivariable Logistic Regression

## Create a vector for multivariate logistic regression.

> multivlogit <- glm(gardasil$Completed ~ gardasil$AgeGroup +
gardasil$MedAssist + gardasil$LocationType + gardasil$PracticeType +
gardasil$Race2, data = gardasil, family = "binomial")

##Finding the odds ratios, the 95% CIs, and extracting p-values from summary
(multivlogit).

> cbind(OR=exp(coef(multivlogit)), exp(confint(multivlogit)),
p=summary(multivlogit)$coef[, 4])
Waiting for profiling to be done...
```

```
                                       OR     2.5 %    97.5 %
(Intercept)                     0.1888946 0.1182674 0.2982195
gardasil$AgeGroup11-17          1.7140308 1.2605387 2.3340832
gardasil$MedAssistno            1.8687278 1.2685351 2.7765079
gardasil$LocationTypesuburban   1.4351719 1.0430511 1.9814658
gardasil$PracticeTypefamily practice 0.7807244 0.5341112 1.1388942
gardasil$PracticeTypeOB-GYN     1.4331787 0.9956191 2.0645077
gardasil$Race2minority          0.6398811 0.4902610 0.8327328
gardasil$Race2other/unknown     0.9375965 0.6605545 1.3233903
                                           p
(Intercept)                     1.556556e-12
gardasil$AgeGroup11-17          6.008125e-04
gardasil$MedAssistno            1.729123e-03
gardasil$LocationTypesuburban   2.718218e-02
gardasil$PracticeTypefamily practice 1.997364e-01
gardasil$PracticeTypeOB-GYN     5.289223e-02
gardasil$Race2minority          9.474385e-04
gardasil$Race2other/unknown     7.158899e-01
```

```
## Giving the odds ratios, 95% CIs and p-values a new vector name.

> multivlogreg<- cbind(OR=exp(coef(multivlogit)), exp(confint(multivlogit)),
p=summary(multivlogit)$coef[, 4])

## Changing the table into a CSV file to edit in excel.
```

```
> write.csv(multivlogreg, "C:/Users/James/Desktop/Table3.csv")

##Syntax for Confounders:

## Examining association between medical assistance(exp) and location
type(potential confounder).
## Set 2x2 table.

> sn.tab15 <-table(gardasil$MedAssist, gardasil$LocationType)
> sn.tab15

      urban suburban
  yes   237       38
  no    213      925




##Changing table and row percentages into CSV for editing in Excel.

> write.csv(sn.tab15, "C:/Users/James/Desktop/snTable15.csv")


## Calculating the magnitude of third variable on main associations.

## Creating a new vector for MedAssist (Public/Private insurance) as
exposure, Completed as outcome and LocationType as confounder.

> confounder1<- glm(formula = gardasil$Completed ~ gardasil$MedAssist +
gardasil$LocationType, family = "binomial", data = gardasil)
##Finding adjusted odds ratios, 95% CIs, extracting p-values from
summary(confounder1)

> cbind(OR=exp(coef(confounder1)), exp(confint(confounder1)),
p=summary(confounder1)$coef[, 4])
Waiting for profiling to be done...
                                    OR     2.5 %    97.5 %
(Intercept)                  0.2412032 0.1771742 0.3227103
gardasil$MedAssistno         1.9419141 1.3396406 2.8428871
gardasil$LocationTypesuburban 1.2761368 0.9513010 1.7198805
                                     p
(Intercept)                  1.218342e-20
gardasil$MedAssistno         5.348087e-04
gardasil$LocationTypesuburban 1.061805e-01

##Giving the odds ratios, 95% CIs, p-values a new name.

> confoundertest1<-cbind(OR=exp(coef(confounder1)),
exp(confint(confounder1)), p=summary(confounder1)$coef[, 4])
Waiting for profiling to be done...

## Changing the table into a CSV file for editing in Excel.

> write.csv(confoundertest1, "C:/Users/James/Desktop/confounderTable1.csv")


## Creating a new vector for MedAssist (Public/Private insurance) as
exposure, Completed as outcome and PracticeType as confounder.

> confounder2<- glm(formula = gardasil$Completed ~ gardasil$MedAssist +
gardasil$PracticeType, family = "binomial", data = gardasil)

##Finding odds ratios, 95% CIs, and extracting p-values from
```

```
summary(confounder2).

> cbind(OR=exp(coef(confounder2)), exp(confint(confounder2)), 
p=summary(confounder2)$coef[, 4])
Waiting for profiling to be done...
                                            OR     2.5 %    97.5 %
(Intercept)                          0.2518927 0.1843916 0.3382898
gardasil$MedAssistno                 2.5266303 1.8082751 3.5781552
gardasil$PracticeTypefamily practice 0.6588756 0.4819030 0.8984027
gardasil$PracticeTypeOB-GYN          1.0397348 0.7931127 1.3629816
                                                p
(Intercept)                          4.526843e-19
gardasil$MedAssistno                 9.728845e-08
gardasil$PracticeTypefamily practice 8.605737e-03
gardasil$PracticeTypeOB-GYN          7.777690e-01

##Giving the odds ratios, 95% CIs and p-values a new vector name.

> confoundertest2<- cbind(OR=exp(coef(confounder2)), 
exp(confint(confounder2)), p=summary(confounder2)$coef[, 4])
Waiting for profiling to be done...

## Changing table into a CSV file for editing in Excel.

> write.csv(confoundertest2, "C:/Users/James/Desktop/confounderTable2.csv")



## Creating a new vector for MedAssist (Public/Private insurance) as 
exposure, Completed as outcome and AgeGroup as confounder.

> confounder3<- glm(formula = gardasil$Completed ~ gardasil$MedAssist + 
gardasil$AgeGroup, family = "binomial", data = gardasil)

##Finding the odds ratios, 95% CIs, and extracting p-values from 
summary(confounder3)

> cbind(OR=exp(coef(confounder3)), exp(confint(confounder3)), 
p=summary(confounder3)$coef[, 4])
Waiting for profiling to be done...
                              OR     2.5 %    97.5 %            p
(Intercept)            0.1865147 0.1303525 0.2629534 5.905593e-21
gardasil$MedAssistno   2.6027686 1.8831295 3.6495183 1.380059e-08
gardasil$AgeGroup11-17 1.4486608 1.1495054 1.8275674 1.720812e-03

##Creating a new vector name for odds ratios, 95% CIs, and p-values.

> confoundertest3<- cbind(OR=exp(coef(confounder3)), 
exp(confint(confounder3)), p=summary(confounder3)$coef[, 4])
Waiting for profiling to be done...

##Changing table into a CSV file for editing in Excel.

> write.csv(confoundertest3, "C:/Users/James/Desktop/confounderTable3.csv")

## Creating a new vector for MedAssist (Public/Private insurance) as 
exposure, Completed as outcome and Race2 as confounder.

> confounder4<- glm(formula = gardasil$Completed ~ gardasil$MedAssist + 
gardasil$Race2, family = "binomial", data = gardasil)

##Finding the odds ratios, 95% CIs, and extracting p-values from 
summary(confounder4).
```

```
> cbind(OR=exp(coef(confounder4)), exp(confint(confounder4)),
p=summary(confounder4)$coef[, 4])
Waiting for profiling to be done...
                                         OR     2.5 %    97.5 %
(Intercept)                       0.3380696 0.2408696 0.4678230
gardasil$MedAssistno              1.9817027 1.4350471 2.7736771
gardasil$RaceSummaryminority      0.5994171 0.4612538 0.7763356
gardasil$RaceSummaryother/unknown 0.8828510 0.6286996 1.2318718
                                              p
(Intercept)                       1.407813e-10
gardasil$MedAssistno              4.591740e-05
gardasil$RaceSummaryminority      1.153635e-04
gardasil$RaceSummaryother/unknown 4.671794e-01

##Giving the odds ratios, 95% CIs and p-values a new vector name.

> confoundertest4<- cbind(OR=exp(coef(confounder4)),
exp(confint(confounder4)), p=summary(confounder4)$coef[, 4])
Waiting for profiling to be done...

##Changing the table into a CSV file to edit in Excel.

> write.csv(confoundertest4, "C:/Users/James/Desktop/confounderTable4.csv")
```