QBS121_ASS4

Parinitha Kompala

2/6/2022

1 Data Analyses 1.1 Modelling Student Absences Analyze the dataset quine which comes with the R library MASS. The dependent variable is number of student absences.

```
library(MASS)
data<-quine
```

1. Put together a table of univariable (1 covariate at at time) results on how each of the covariates relate to student absences.

```
z<-glm(Days~Eth, data=data)
summary(z)
```

```
##
## Call:
## glm(formula = Days ~ Eth, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -21.232  -10.232   -5.182    5.568   59.768
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   21.232      1.886  11.261  < 2e-16 ***
## EthN          -9.050      2.596  -3.486 0.000651 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 245.3038)
##
##     Null deviance: 38304  on 145  degrees of freedom
## Residual deviance: 35324  on 144  degrees of freedom
## AIC: 1221.7
##
## Number of Fisher Scoring iterations: 2
```

```
z$coefficients
```

```
## (Intercept)        EthN
##   21.231884   -9.050066
```

```
x<-glm(Days~Sex, data=data)
summary(x)
```

```
##
## Call:
```

```
## glm(formula = Days ~ Sex, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -17.955  -10.955   -5.090    6.525   65.775
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.225      1.817   8.379 4.37e-14 ***
## SexM           2.730      2.703   1.010    0.314
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 264.1307)
##
##     Null deviance: 38304  on 145  degrees of freedom
## Residual deviance: 38035  on 144  degrees of freedom
## AIC: 1232.5
##
## Number of Fisher Scoring iterations: 2
```

```
x$coefficients
```

```
## (Intercept)        SexM
##   15.225000    2.729545
```

```
c<-glm(Days~Age, data=data)
summary(c)
```

```
##
## Call:
## glm(formula = Days ~ Age, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -21.050   -9.852   -4.951    5.924   59.950
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   14.852      3.054   4.862 3.04e-06 ***
## AgeF1         -3.700      3.848  -0.962    0.338
## AgeF2          6.198      3.953   1.568    0.119
## AgeF3          4.754      4.119   1.154    0.250
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 251.8952)
##
##     Null deviance: 38304  on 145  degrees of freedom
## Residual deviance: 35769  on 142  degrees of freedom
## AIC: 1227.5
##
## Number of Fisher Scoring iterations: 2
```

```
c$coefficients
```

```
## (Intercept)       AgeF1       AgeF2       AgeF3
##   14.851852   -3.699678    6.198148    4.754209
```

```
v<-glm(Days~Lrn, data=data)
summary(v)
```

```
##
## Call:
## glm(formula = Days ~ Lrn, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -17.302  -11.302   -5.060    6.181   63.698
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   15.819      1.788   8.846 2.97e-15 ***
## LrnSL          1.482      2.722   0.544    0.587
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 265.4553)
##
##     Null deviance: 38304  on 145  degrees of freedom
## Residual deviance: 38226  on 144  degrees of freedom
## AIC: 1233.2
##
## Number of Fisher Scoring iterations: 2
```

```
v$coefficients
```

```
## (Intercept)       LrnSL
##    15.81928     1.48231
```

#creating a presentatble table

```
attach(data)
Univ <- matrix(nrow=0, ncol=4)
dimnames(Univ)[[2]] <- c("Odds Ratio", "95%CI Lo", "Up", "P-value")
Columns <- c(1,2,3,4)
for (i in Columns) {
  os <- summary(glm(Days ~ data[,i]))
  #os<-summary(glm(Days~data[,i], data=data))
  Univ <- rbind(Univ, c(exp(os$coef[2,1:2] %*% matrix(nrow=2, ncol=3, c(1,0,1,-2,1,+2))), os$coef[2,4])
}
dimnames(Univ)[[1]] <- names(data)[Columns]
round(Univ, 3)
```

```
##     Odds Ratio 95%CI Lo       Up P-value
## Eth      0.000    0.000    0.021   0.001
## Sex     15.326    0.069 3410.400   0.314
## Age      0.025    0.000   54.374   0.338
## Lrn      4.403    0.019 1019.682   0.587
```

```
detach(data)
```

2. Put together a table of multivariable results, i.e., run a multivariable model using all of the variables (or a subset if you choose).

```
mod1.1.2<-glm(Days~Eth+Sex+Age+Lrn,data=data)
summary(mod1.1.2)
```

```
##
## Call:
## glm(formula = Days ~ Eth + Sex + Age + Lrn, data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -23.038  -10.027   -3.297    7.094   54.799
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   16.233      3.767   4.309 3.08e-05 ***
## EthN          -8.745      2.529  -3.458 0.000721 ***
## SexM           2.530      2.635   0.960 0.338631
## AgeF1         -4.457      3.929  -1.134 0.258547
## AgeF2          4.701      3.906   1.204 0.230778
## AgeF3          6.805      4.107   1.657 0.099771 .
## LrnSL          5.267      3.055   1.724 0.086934 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 231.9183)
##
##     Null deviance: 38304  on 145  degrees of freedom
## Residual deviance: 32237  on 139  degrees of freedom
## AIC: 1218.3
##
## Number of Fisher Scoring iterations: 2
```

3. Do this in two ways,

(i) using Poisson regression in conjunction with sandwich variance to determine standard errors (or by selecting family=quasipoisson in the glm function) and

(ii) negative binomial regression. Comment on the difference or similarity between the two sets of results.

```
mod1.1.3.1<-glm(Days~Eth+Sex+Age+Lrn,family=quasipoisson,data=data)
summary(mod1.1.3.1)
```

```
##
## Call:
## glm(formula = Days ~ Eth + Sex + Age + Lrn, family = quasipoisson,
##     data = data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -6.808  -3.065  -1.119   1.819   9.909
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.7154     0.2347  11.569  < 2e-16 ***
## EthN         -0.5336     0.1520  -3.511 0.000602 ***
## SexM          0.1616     0.1543   1.047 0.296914    
## AgeF1        -0.3339     0.2543  -1.313 0.191413    
## AgeF2         0.2578     0.2265   1.138 0.256938    
## AgeF3         0.4277     0.2456   1.741 0.083831 .  
## LrnSL         0.3489     0.1888   1.848 0.066760 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for quasipoisson family taken to be 13.16691)
## 
##     Null deviance: 2073.5  on 145  degrees of freedom
## Residual deviance: 1696.7  on 139  degrees of freedom
## AIC: NA
## 
## Number of Fisher Scoring iterations: 5
```

```
#Quasi Poisson indicates that only Ethnicity and intercept are significant.
```

```
mod1.1.3.2<-glm.nb(Days ~ Sex + Age + Eth + Lrn, data=data)
summary(mod1.1.3.2)
```

```
## 
## Call:
## glm.nb(formula = Days ~ Sex + Age + Eth + Lrn, data = data, init.theta = 1.274892646, 
##     link = log)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.7918  -0.8892  -0.2778   0.3797   2.1949  
## 
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  2.89458    0.22842  12.672  < 2e-16 ***
## SexM         0.08232    0.15992   0.515 0.606710    
## AgeF1       -0.44843    0.23975  -1.870 0.061425 .  
## AgeF2        0.08808    0.23619   0.373 0.709211    
## AgeF3        0.35690    0.24832   1.437 0.150651    
## EthN        -0.56937    0.15333  -3.713 0.000205 ***
## LrnSL        0.29211    0.18647   1.566 0.117236    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for Negative Binomial(1.2749) family taken to be 1)
## 
##     Null deviance: 195.29  on 145  degrees of freedom
## Residual deviance: 167.95  on 139  degrees of freedom
## AIC: 1109.2
## 
## Number of Fisher Scoring iterations: 1
## 
## 
##               Theta:  1.275 
##           Std. Err.:  0.161 
```

```
##
##  2 x log-likelihood:  -1093.151
```

```
#We get roughly the same qualitative conclusion as quasi Poisson
```

1.2 Cancer Counts in Danish Cities Access the data eba1977 in the R library ISwR. This is a small dataset on cancer counts by city and age group in Denmark.

```
library(ISwR)
```

1. Which variable makes sense to use as an offset. Answer-pop,this variable that is used to denote the exposure period in the Poisson regression
2.

a. Use Poisson regression to model the association with age group.

```
model1.2.2.a <- glm(cases ~age, offset = log(pop), family = poisson, data = eba1977)
summary(model1.2.2.a)
```

```
##
## Call:
## glm(formula = cases ~ age, family = poisson, data = eba1977,
##     offset = log(pop))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8520  -0.6424  -0.1067   0.7853   1.5468
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.8623     0.1741 -33.676  < 2e-16 ***
## age55-59      1.0823     0.2481   4.363 1.29e-05 ***
## age60-64      1.5017     0.2314   6.489 8.66e-11 ***
## age65-69      1.7503     0.2292   7.637 2.22e-14 ***
## age70-74      1.8472     0.2352   7.855 4.00e-15 ***
## age75+        1.4083     0.2501   5.630 1.80e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 129.908  on 23  degrees of freedom
## Residual deviance:  28.307  on 18  degrees of freedom
## AIC: 136.69
##
## Number of Fisher Scoring iterations: 5
```

b. Test the significance of age using anova(o.glm, test=”Chisq”)

```
anova(model1.2.2.a,test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: cases
##
## Terms added sequentially (first to last)
```

```
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    23    129.908
## age   5    101.6        18     28.307 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

c. Test the association of age as an ordinal variable (hint: create ageOrdinal = as.numeric(age)).

```
#ageOrdinal = as.numeric(data2$age)

model1.2.2.c <- glm(cases ~ as.numeric(age), offset = log(pop), family = poisson, data = eba1977)
summary(model1.2.2.c )
```

```
##
## Call:
## glm(formula = cases ~ as.numeric(age), family = poisson, data = eba1977,
##     offset = log(pop))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.3267  -0.3953   0.2912   1.0869   2.3017
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -5.63185    0.14058 -40.062  < 2e-16 ***
## as.numeric(age)  0.28459    0.03498   8.135 4.11e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 129.908  on 23  degrees of freedom
## Residual deviance:  65.323  on 22  degrees of freedom
## AIC: 165.71
##
## Number of Fisher Scoring iterations: 4
```

3. a. Use Poisson regression to model the association with city.

```
model1.2.3.a <- glm(cases ~city, offset = log(pop), family = poisson, data = eba1977)
summary(model1.2.3.a)
```

```
##
## Call:
## glm(formula = cases ~ city, family = poisson, data = eba1977,
##     offset = log(pop))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.8908  -0.3705   1.0893   2.2012   3.1090
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.5837     0.1250 -36.670   <2e-16 ***
## cityHorsens  -0.2286     0.1813  -1.261   0.2073
## cityKolding  -0.3357     0.1877  -1.789   0.0737 .
## cityVejle    -0.1883     0.1877  -1.003   0.3157
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 129.91  on 23  degrees of freedom
## Residual deviance: 126.52  on 20  degrees of freedom
## AIC: 230.9
##
## Number of Fisher Scoring iterations: 5
```

b. Test the significance of city.

```
ageOrdinal <- as.numeric(eba1977$age)
model1.2.3.b<- glm(eba1977$cases~ eba1977$city + ageOrdinal, family = "poisson")
summary(model1.2.3.b)
```

```
##
## Call:
## glm(formula = eba1977$cases ~ eba1977$city + ageOrdinal, family = "poisson")
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.08323  -0.48712   0.06495   0.55558   1.70502
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)           2.32404    0.18645  12.464   <2e-16 ***
## eba1977$cityHorsens -0.09844    0.18129  -0.543    0.587
## eba1977$cityKolding -0.22706    0.18770  -1.210    0.226
## eba1977$cityVejle   -0.22706    0.18770  -1.210    0.226
## ageOrdinal           0.01225    0.03913   0.313    0.754
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 27.704  on 23  degrees of freedom
## Residual deviance: 25.523  on 19  degrees of freedom
## AIC: 131.91
##
## Number of Fisher Scoring iterations: 4
```

4. Run the multivariable model with city and age.

```
model1.2.4 <- glm(cases ~ city + age, offset = log(pop), family = poisson(link = "log"), data = eba1977
summary(model1.2.4)
```

```
##
## Call:
## glm(formula = cases ~ city + age, family = poisson(link = "log"),
##     data = eba1977, offset = log(pop))
##
```

```
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.63573  -0.67296  -0.03436   0.37258   1.85267
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.6321     0.2003 -28.125  < 2e-16 ***
## cityHorsens  -0.3301     0.1815  -1.818   0.0690 .
## cityKolding  -0.3715     0.1878  -1.978   0.0479 *
## cityVejle    -0.2723     0.1879  -1.450   0.1472
## age55-59      1.1010     0.2483   4.434 9.23e-06 ***
## age60-64      1.5186     0.2316   6.556 5.53e-11 ***
## age65-69      1.7677     0.2294   7.704 1.31e-14 ***
## age70-74      1.8569     0.2353   7.891 3.00e-15 ***
## age75+        1.4197     0.2503   5.672 1.41e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 129.908  on 23  degrees of freedom
## Residual deviance:  23.447  on 15  degrees of freedom
## AIC: 137.84
##
## Number of Fisher Scoring iterations: 5
```

5. For interest, instead of using an offset include log(population) as a covariate. Is the coefficient significantly different from 1.0 ?

```
model1.2.5 <- glm(cases ~ city + age+log(pop) , family = poisson(link = "log"), data = eba1977)
summary(model1.2.5)
```

```
##
## Call:
## glm(formula = cases ~ city + age + log(pop), family = poisson(link = "log"),
##     data = eba1977)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.44001  -0.64195  -0.04286   0.50052   1.51893
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  11.7496     8.8151   1.333    0.183
## cityHorsens   0.1833     0.3193   0.574    0.566
## cityKolding  -0.0483     0.2520  -0.192    0.848
## cityVejle    -0.1679     0.1965  -0.855    0.393
## age55-59     -1.3842     1.2729  -1.087    0.277
## age60-64     -1.2367     1.4049  -0.880    0.379
## age65-69     -1.4378     1.6310  -0.882    0.378
## age70-74     -1.8049     1.8608  -0.970    0.332
## age75+       -1.8383     1.6588  -1.108    0.268
## log(pop)     -1.2096     1.1227  -1.077    0.281
##
## (Dispersion parameter for poisson family taken to be 1)
```

```
##
##     Null deviance: 27.704  on 23  degrees of freedom
## Residual deviance: 19.498  on 14  degrees of freedom
## AIC: 135.89
##
## Number of Fisher Scoring iterations: 4
```

2 Simulate and Analyze 2.1 Large Counts: Linear Regression vs Poisson If the dependent variable is a count that takes large values (e.g. counts that are zero with very low frequency) it may be preferable to use linear regression. 1. Choose a sample size, e.g. n=500

```
n=500
```

2. Generate a couple continuous variables, Z1=rnorm(n) and Z2=rnorm(n)

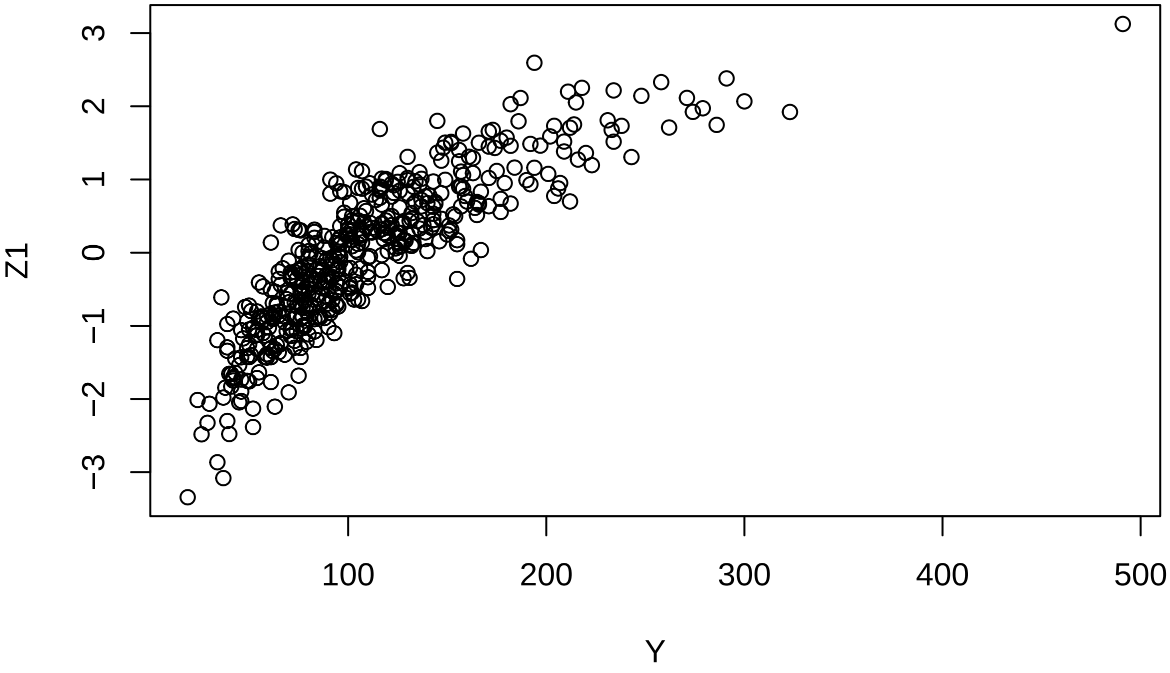
```
Z1=rnorm(n)
Z2=rnorm(n)
```

3. Generate a large count Y=rpois(n, lambda=100*1.5**Z1/1.2**Z2)

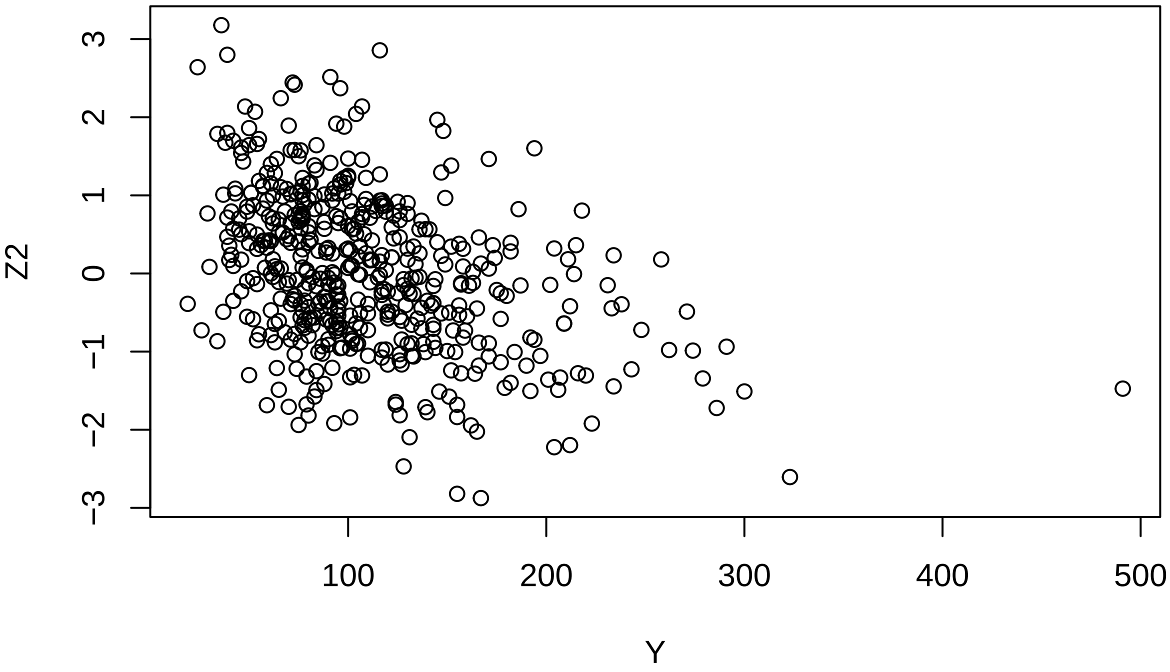
```
Y=rpois(n, lambda=100*1.5**Z1/1.2**Z2)
```

4. Plot this count vs Z1, and then versus Z2

```
plot(Y,Z1)
```



```
plot(Y,Z2)
```



5. Use multivariable Poisson regression to model Y vs Z1 and Z2.

```
mod2.5<-glm(Y~Z1+Z2,family=poisson)
summary(mod2.5)
```

```
##
## Call:
## glm(formula = Y ~ Z1 + Z2, family = poisson)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.5242  -0.7381   0.0014   0.7603   2.7776
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.605494   0.004642  992.06   <2e-16 ***
## Z1           0.399815   0.004203   95.11   <2e-16 ***
## Z2          -0.182644   0.004285  -42.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 11730.37  on 499  degrees of freedom
## Residual deviance:   530.69  on 497  degrees of freedom
## AIC: 3749.8
##
## Number of Fisher Scoring iterations: 4
```

6. Use multivariable linear regression to model Y vz Z1 and Z2

```
mod2.6<-lm(Y~Z1+Z2)
summary(mod2.6)
```

```
##
## Call:
## lm(formula = Y ~ Z1 + Z2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -38.800 -12.594  -2.859   7.761 213.354
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 110.7813     0.9002  123.06   <2e-16 ***
## Z1           43.7390     0.8794   49.74   <2e-16 ***
## Z2          -20.4976     0.9010  -22.75   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.12 on 497 degrees of freedom
## Multiple R-squared:  0.8593, Adjusted R-squared:  0.8587
## F-statistic:  1517 on 2 and 497 DF,  p-value: < 2.2e-16
```

7. Use multivariable linear regression to model log(Y) vz Z1 and Z2

```r
mod2.7<-lm(log(Y)~Z1+Z2)
summary(mod2.7)
```

```
##
## Call:
## lm(formula = log(Y) ~ Z1 + Z2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.46892 -0.06973  0.00507  0.07602  0.32123
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  4.599516   0.004927  933.54   <2e-16 ***
## Z1           0.404459   0.004813   84.04   <2e-16 ***
## Z2          -0.181185   0.004931  -36.74   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1101 on 497 degrees of freedom
## Multiple R-squared:  0.9449, Adjusted R-squared:  0.9447
## F-statistic:  4265 on 2 and 497 DF,  p-value: < 2.2e-16
```

8. Assess similarities and differences of the estimates, standard errors and Z-values from these three models. Answer- Similarities: In all models, the pvalues of Z1 and Z2 are significant. The estimates, standard errors, and z-values from the poisson regression and log linear regression models are identical (model of 2.1.7). The log transformed linear regression is similar to poisson regression when the counts value is big. Differences- The estimates are different for linear regression with the other two, the t value is different for everyone but the poisson regression and log linear somw what has simialr.

3 Simulation

3.1 AUROC as Measure of Difference of Two Distributions

The AUROC of a score that predicts an event equals the probability that a subject with the event will have a higher score than a person without the event. If the distribution of the scores in subjects with the event is normal with mean m1 and s1 and the distribution of scores in subjects without the event is normal with mean m0 and s0, then the following R line of code estimates the concordancy.

a. Create a table of Concordancy vs the following choices, m0=0,sd=1, m1 = 0.0, 0.25,0.5,0.75,1,1.5,2,3 and s1=1.

```
m0<- 0
s0<-1
m1<-c(0,0.25,0.5,0.75,1.0,1.5,2,3)
s1<-1
n<-1000


m0.values<-rep(0,8)
s0.values<-rep(1,8)
s1.values<-rep(1,8)
auc<-rep(0,8)

for (i in 1:8){
  auc[i]=mean( rnorm(n=n<-10^6, mean=m0, sd=s0) < rnorm(n=n, mean=m1[i], sd=s1) )
}

Tab<-as.data.frame(cbind(m0.values,s0.values,m1,s1.values,auc))
Tab
```

```
##   m0.values s0.values   m1 s1.values      auc
## 1         0         1 0.00         1 0.499492
## 2         0         1 0.25         1 0.570035
## 3         0         1 0.50         1 0.638689
## 4         0         1 0.75         1 0.702743
## 5         0         1 1.00         1 0.760735
## 6         0         1 1.50         1 0.855389
## 7         0         1 2.00         1 0.921424
## 8         0         1 3.00         1 0.982955
```

b. Suppose a score for the risk of an event is such that its distribution in those who will have the event is normal with mean m1 and standard deviation of s1, and its distribution in those who will not have the event is mean 0 and standard deviation 1. Simulate the score of 1000 events (cases) and 1000 controls and plot the corresponding ROC curve for the following 4 scenarios (m1=0.5, s1=1), (m1=0.5, s1=2), (m1=2.0, s1=1), (m1=2.0, s1=2).

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
m1=0.5
s1=1
Score1 <- rnorm(n=n, mean=m1,sd=s1)
Score0 <- rnorm(n=n, mean=0,sd=1)
Event <- rep(c(1,0), each=n)
```

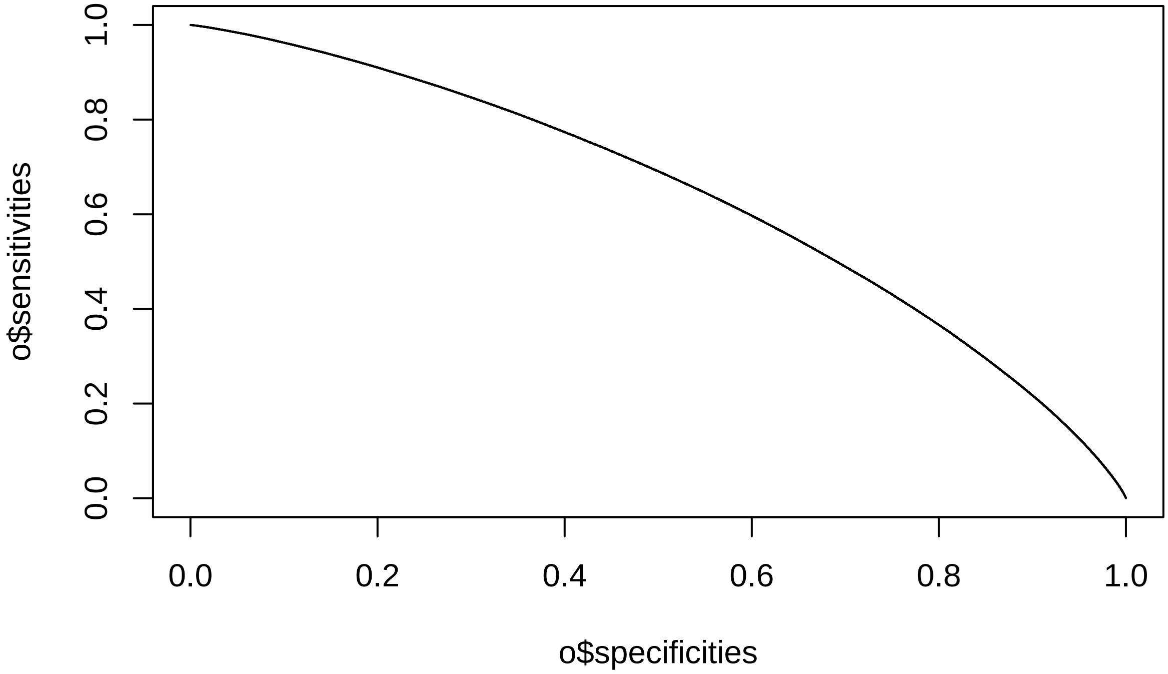
```
Score <- c(Score1, Score0)
o <- roc(Event, Score)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(o$specificities, o$sensitivities, type="line")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'line' will be truncated to first
## character
```



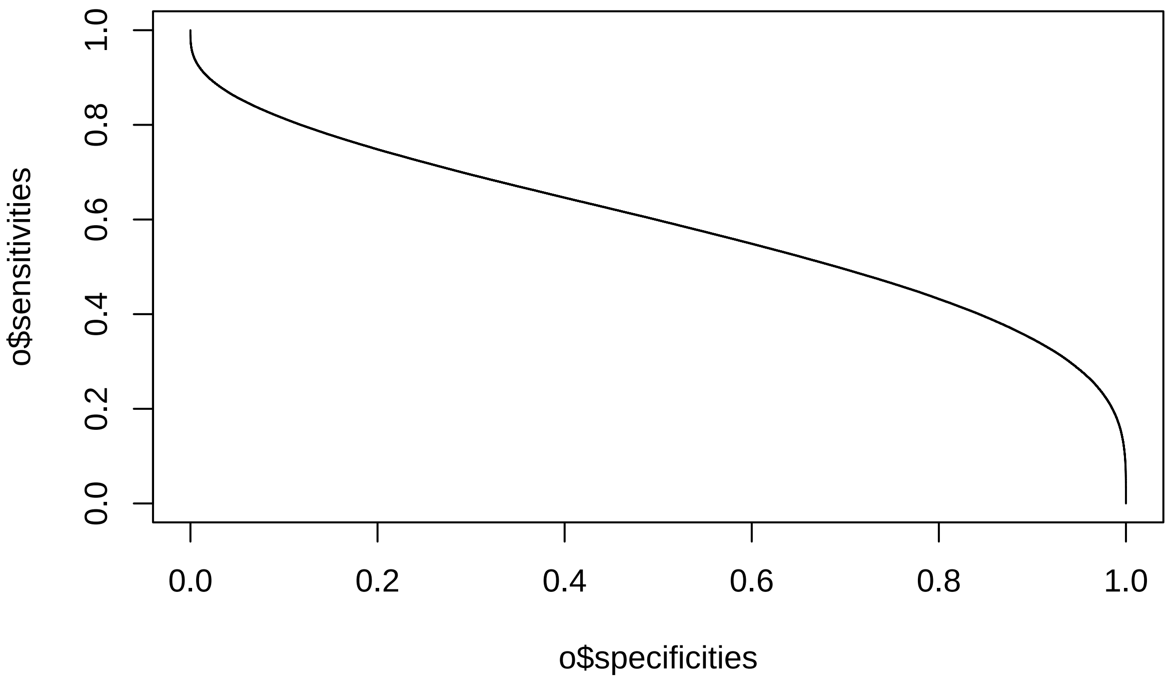
```
m1=0.5
s1=2
Score1 <- rnorm(n=n, mean=m1,sd=s1)
Score0 <- rnorm(n=n, mean=0,sd=1)
Event <- rep(c(1,0), each=n)
Score <- c(Score1, Score0)
o <- roc(Event, Score)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(o$specificities, o$sensitivities, type="line")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'line' will be truncated to first
## character
```



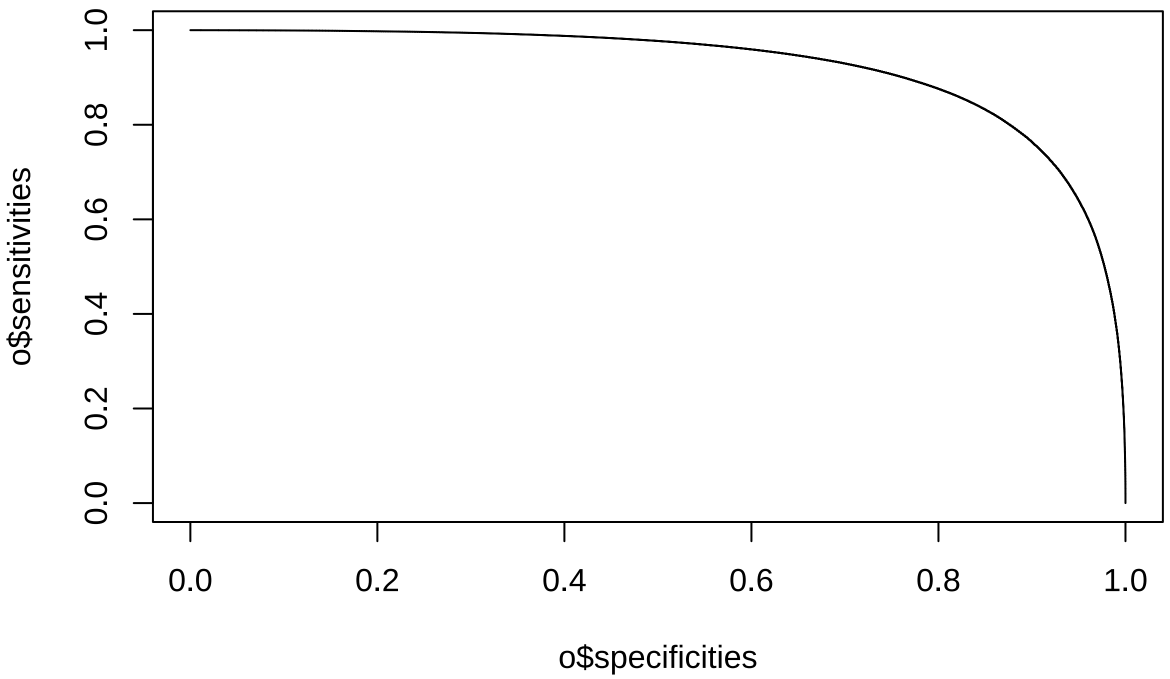
```
m1=2.0
s1=1
Score1 <- rnorm(n=n, mean=m1,sd=s1)
Score0 <- rnorm(n=n, mean=0,sd=1)
Event <- rep(c(1,0), each=n)
Score <- c(Score1, Score0)
o <- roc(Event, Score)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(o$specificities, o$sensitivities, type="line")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'line' will be truncated to first
## character
```



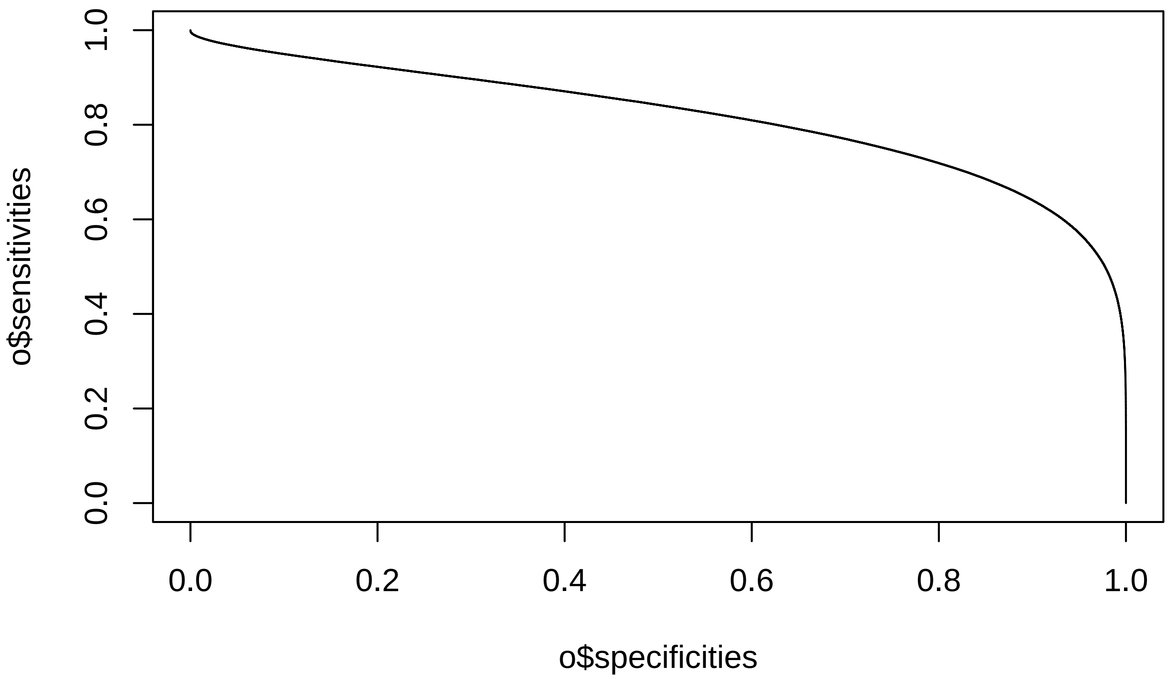
```
m1=2.0
s1=2
Score1 <- rnorm(n=n, mean=m1,sd=s1)
Score0 <- rnorm(n=n, mean=0,sd=1)
Event <- rep(c(1,0), each=n)
Score <- c(Score1, Score0)
o <- roc(Event, Score)
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

```
plot(o$specificities, o$sensitivities, type="line")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'line' will be truncated to first
## character
```

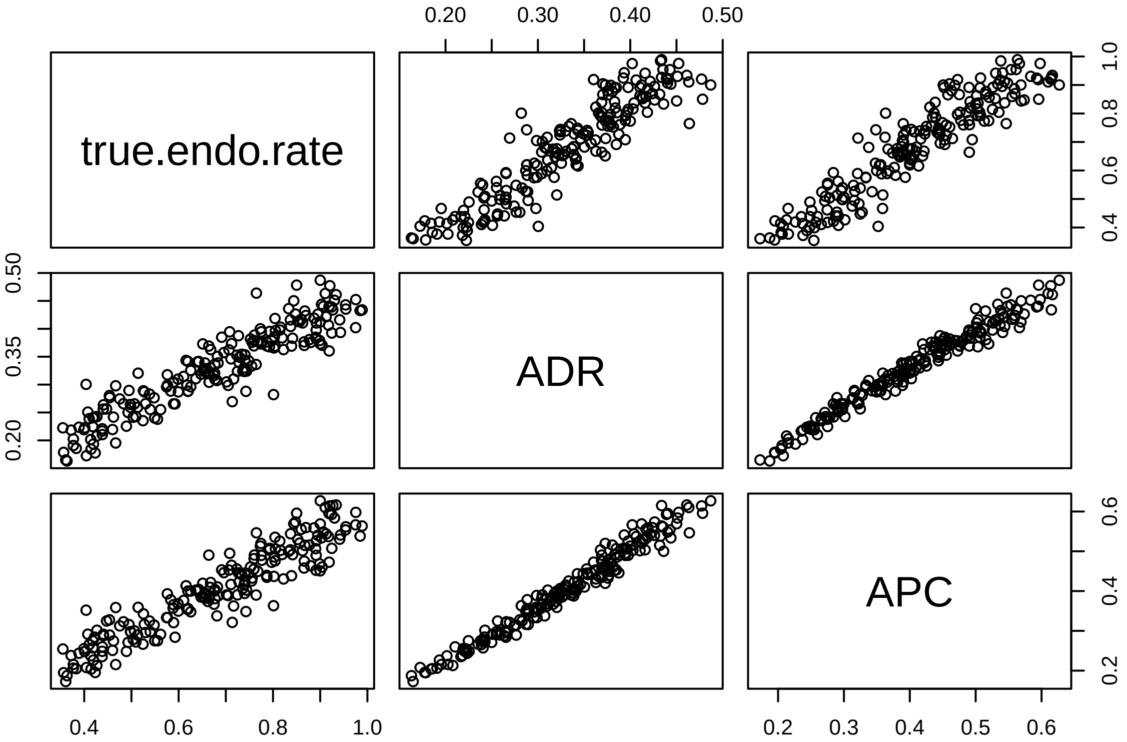


3.2 ADR vs APC The most recommended modality for colorectal cancer screening in the USA is colonoscopy.During a colonoscopy a clinician uses a camera at the end of a tube (colonoscope) to examine the colon. The colonoscope is also equipped with features to remove pre-cancerous lesions (polyps, adenomas). Colonoscopists vary in their ability to detect polyps. One measure of detection ability is he Adenoma Detection Rate (ADR). It is defined as the proportion of colonoscopies in which at least one adenoma is detected; like the proportion of games in which an athlete gets at least one point. An alternative metric is the APC (adenomas per colonoscopies); like the average number of points per game. Explain what the following simulation is doing and interpret the results.

Amswer-Endoscopists and the quantity of polyps in a simulated community were first produced. The ADR and APC were determined. The adenoma detection rate (ADR) is the percentage of colonoscopies that reveal at least one adenoma. The APC is the number of adenomas that each trail can detect on average. The ADR and APC are then compared to endoscopists' genuine rate, with a plot and calculation of the correlation between true rate and each technique. The APC is a better measure of detection ability than the ADR

```
R <- 1000
cor.ADR.true <- cor.APC.true <- R
n.endoscopists <- 200 # number of endoscopists in the cohort
for (r in 1:R) {
  # number of patients each endoscopists scopes in a year
  n.pt.endoscopist <- ceiling(rgamma(n=n.endoscopists, shape=10, scale=30))
  N <- sum(n.pt.endoscopist)
  ID.Endo <- rep(1:n.endoscopists, times=n.pt.endoscopist)
  true.endo.rate <- runif(n.endoscopists, min=0.35,max=0.99) # given a uniform distribution
  long.true.endo.rate <- rep(true.endo.rate, times=n.pt.endoscopist)
  n.polyps <- rpois(n=N, lambda=0.6) # lambda is the average actual adenomas
  n.polyps.detected <- rbinom(n=N, size=n.polyps, prob=long.true.endo.rate)
  at.least.one <- n.polyps.detected>0
  ADR <- tapply(at.least.one, ID.Endo, mean)
  APC <- tapply(n.polyps.detected, ID.Endo, mean)
  cor.ADR.true[r] <- cor(ADR, true.endo.rate)
  cor.APC.true[r] <- cor(APC, true.endo.rate)
}
```

```
pairs(cbind(true.endo.rate, ADR, APC))
```



```
summary(cbind(cor.ADR.true, cor.APC.true))
```

```
##   cor.ADR.true     cor.APC.true
##  Min.   :0.9053   Min.   :0.9213
##  1st Qu.:0.9290   1st Qu.:0.9400
##  Median :0.9344   Median :0.9451
##  Mean   :0.9344   Mean   :0.9445
##  3rd Qu.:0.9403   3rd Qu.:0.9496
##  Max.   :0.9589   Max.   :0.9652
```