Homework 04

Generalized Linear Models

Name

October 5, 2017

Data analysis

Poisson regression:

The folder `risky.behavior` contains data from a randomized trial targeting couples at high risk of HIV infection. The intervention provided counseling sessions regarding practices that could reduce their likelihood of contracting HIV. Couples were randomized either to a control group, a group in which just the woman participated, or a group in which both members of the couple participated. One of the outcomes examined after three months was "number of unprotected sex acts".

- "sex" is a factor variable with labels "woman" and "man". This is the member of the couple that reporting sex acts to the researcher
- The variables "couple" and "women_alone" code the intervention:

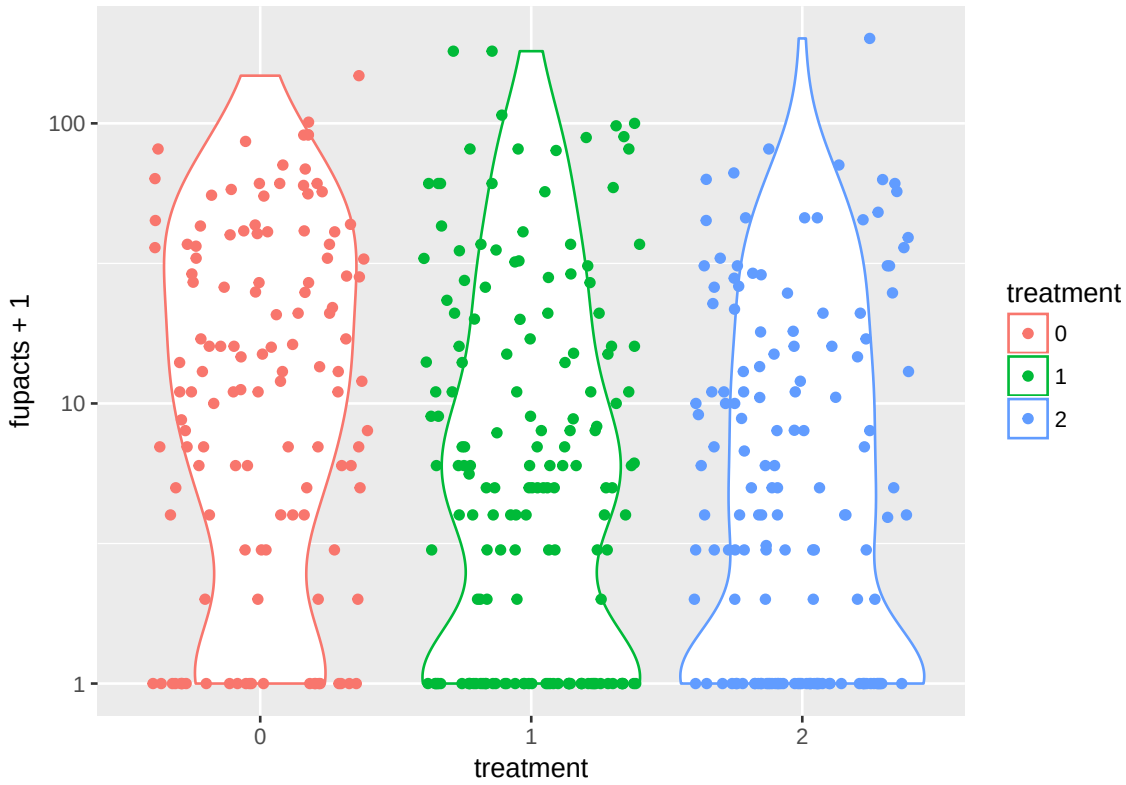
couple	women_alone	
0	0	control - no counseling
1	0	the couple was counseled together
0	1	only the woman was counseled

- "bs_hiv" indicates whether the member reporting sex acts was HIV-positive at "baseline", that is, at the beginning of the study.
- "bupacts" - number of unprotected sex acts reported at "baseline",that is, at the beginning of the study
- "fupacts" - number of unprotected sex acts reported at the end of the study (final report).

1. Model this outcome as a function of treatment assignment using a Poisson regression. Does the model fit well? Is there evidence of overdispersion?

We start with EDA, the outcome of interest is `fupacts`

```
ggplot(risky_behaviors)+geom_violin()+geom_jitter()+
  aes(x=treatment,y=fupacts+1,color=treatment)+scale_y_log10()
```



we add 1 so that we can take logs when displaying.

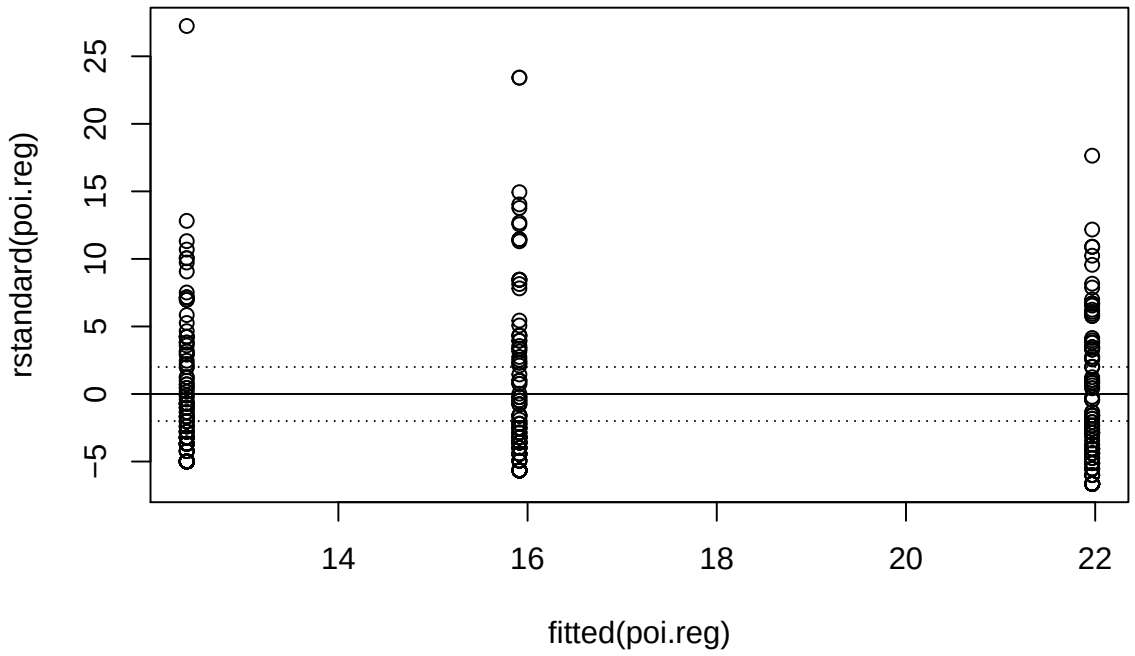
Since there are three treatment conditions the intercept is the expected values for the control group.

```
poi.reg <- glm(round(fupacts) ~ couples + women_alone,
               family=poisson, data=risky_behaviors)
display(poi.reg)
```

```
## glm(formula = round(fupacts) ~ couples + women_alone, family = poisson,
##     data = risky_behaviors)
##             coef.est coef.se
## (Intercept)  3.09     0.02
## couples     -0.32     0.03
## women_alone -0.57     0.03
## ---
##   n = 434, k = 3
##   residual deviance = 12925.5, null deviance = 13298.6 (difference = 373.1)
```

The residual plot shows signs of overdispersion

```
plot(fitted(poi.reg),rstandard(poi.reg));abline(h=0);
abline(h=2,lty=3)
abline(h=-2,lty=3)
```




```
#plot(jitter(fitted(poi.reg)),jitter(log(risky_behaviors$fupacts-fitted(poi.reg))/sqrt(fitted(poi.reg))
```

The estimated dispersion parameter is very large, suggesting overdispersion.

```
qpoi.reg <- glm(round(fupacts) ~ couples+women_alone, family=quasipoisson, data=risky_behaviors)
summary(qpoi.reg)$dispersion
```

```
## [1] 44.13468
```

2. Next extend the model to include pre-treatment measures of the outcome and the additional pre-treatment variables included in the dataset. Does the model fit well? Is there evidence of overdispersion?

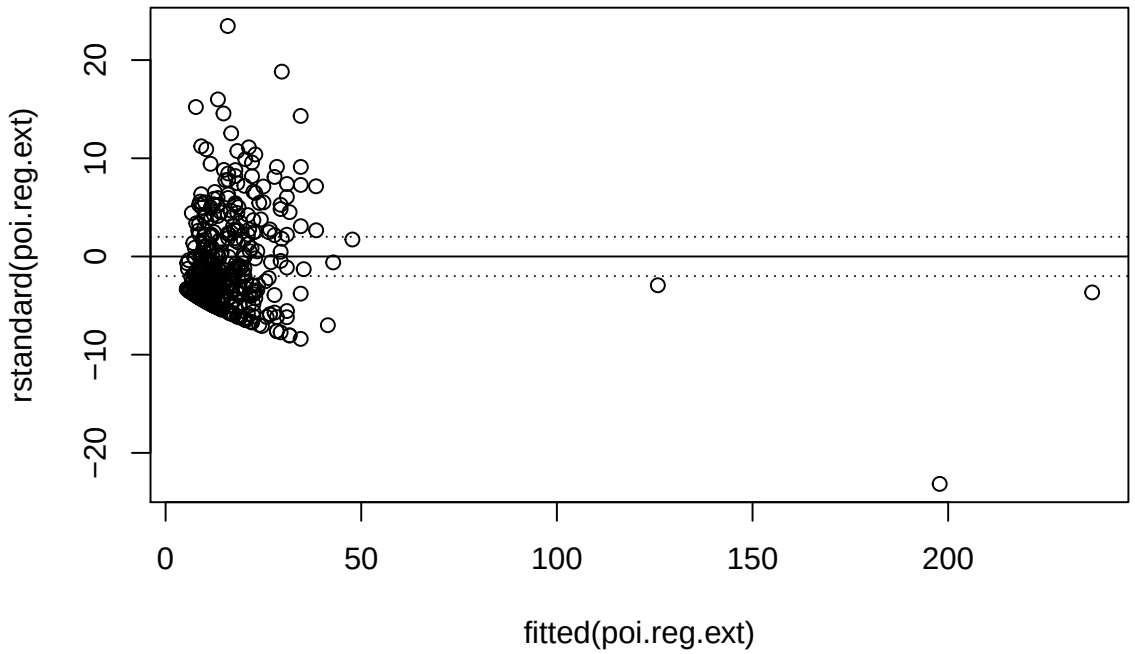
We will start by centering the variable `bupacts`

```
risky_behaviors$bupacts_c <- scale(risky_behaviors$bupacts, center=TRUE)
```

```
poi.reg.ext <- glm(round(fupacts) ~ women_alone + sex + bupacts_c + couples + bs_hiv, family=poisson, da
display(poi.reg.ext)
```

```
## glm(formula = round(fupacts) ~ women_alone + sex + bupacts_c +
##     couples + bs_hiv, family = poisson, data = risky_behaviors)
##                coef.est coef.se
## (Intercept)     3.18     0.02
## women_alone    -0.66     0.03
## sexman         -0.11     0.02
## bupacts_c       0.34     0.01
## couples        -0.41     0.03
## bs_hivpositive -0.44     0.04
## ---
##   n = 434, k = 6
##   residual deviance = 10200.4, null deviance = 13298.6 (difference = 3098.2)
```

```
plot(fitted(poi.reg.ext),rstandard(poi.reg.ext));abline(h=0)
abline(h=2,lty=3)
abline(h=-2,lty=3)
```

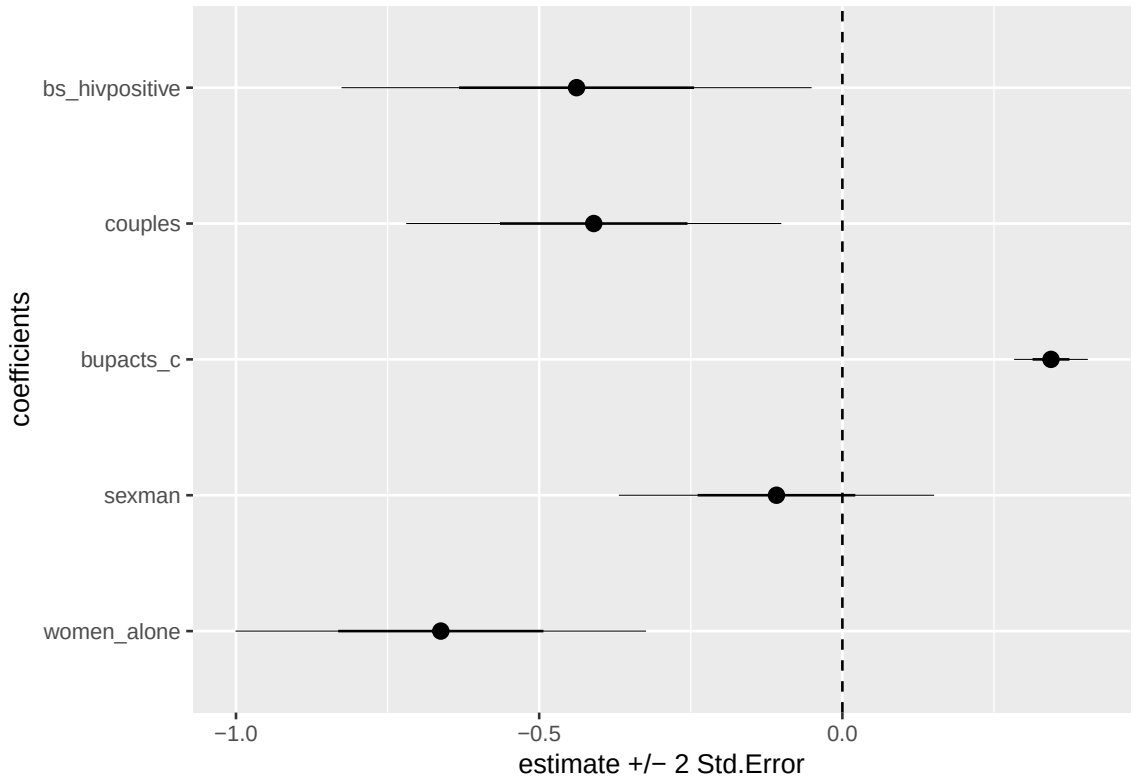


3. Fit an overdispersed Poisson model. What do you conclude regarding effectiveness of the intervention?

```
qpoi.reg.ext <- glm(round(fupacts) ~ women_alone + sex + bupacts_c + couples + bs_hiv, family=quasipois
display(qpoi.reg.ext)
```

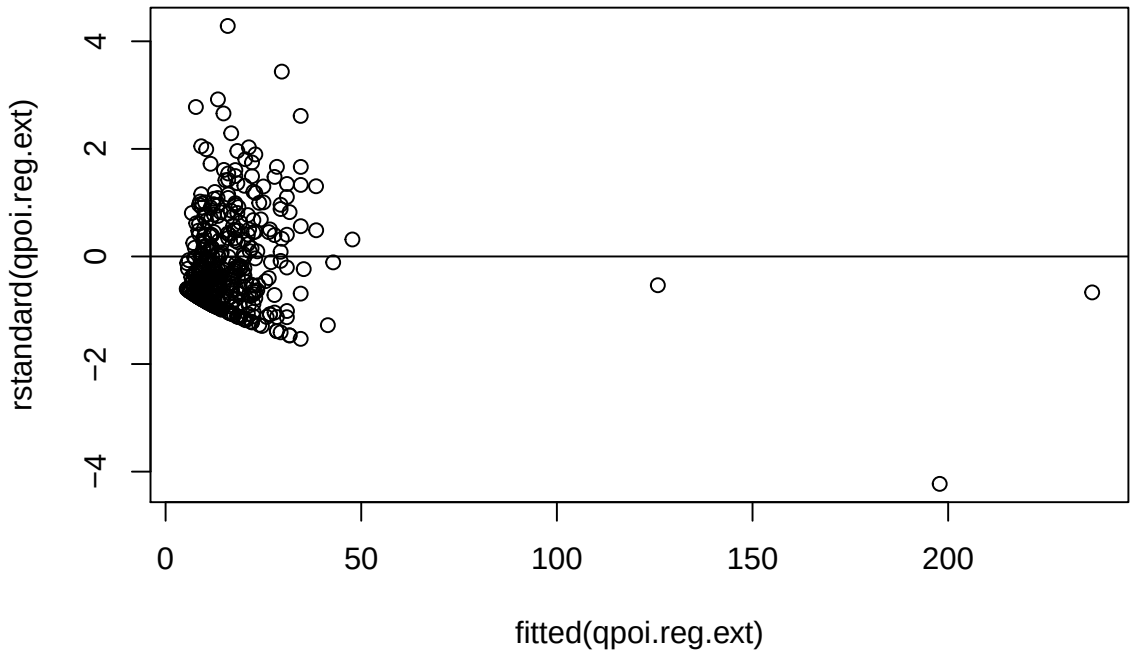
```
## glm(formula = round(fupacts) ~ women_alone + sex + bupacts_c +
##     couples + bs_hiv, family = quasipoisson, data = risky_behaviors)
##                coef.est coef.se
## (Intercept)     3.18     0.12
## women_alone    -0.66     0.17
## sexman         -0.11     0.13
## bupacts_c       0.34     0.03
## couples        -0.41     0.15
## bs_hivpositive -0.44     0.19
## ---
##   n = 434, k = 6
##   residual deviance = 10200.4, null deviance = 13298.6 (difference = 3098.2)
##   overdispersion parameter = 30.0
```

```
coefplot_my(qpoi.reg.ext)
```

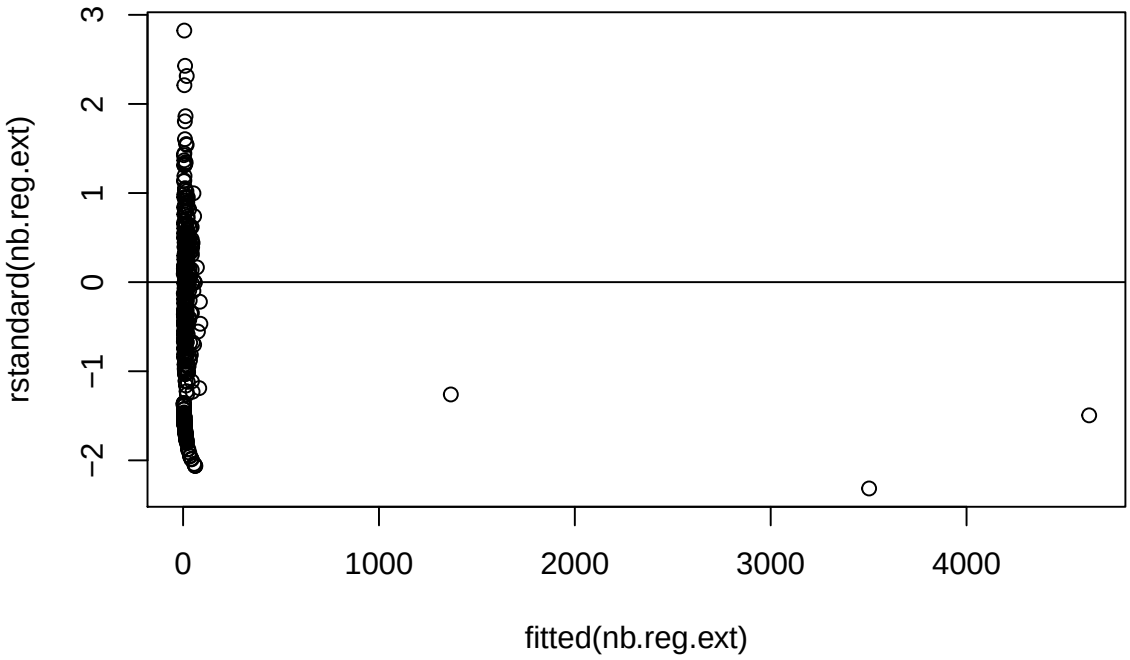


There seems to be significant reduction in the number of unprotected sex for `women_alone` and `couples` treatmented group on average compared to the control group.

```
plot(fitted(qpoi.reg.ext),rstandard(qpoi.reg.ext));abline(h=0)
```

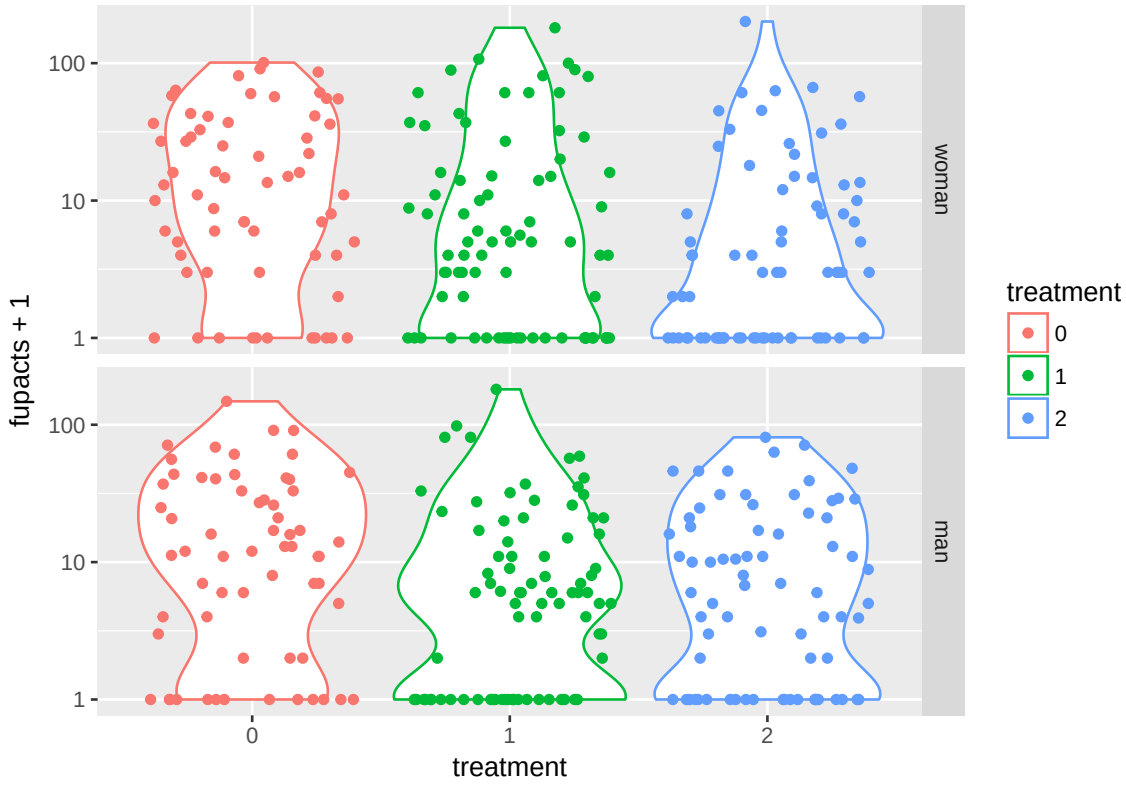



```
nb.reg.ext <- MASS::glm.nb(round(fupacts) ~ women_alone + sex + bupacts_c + couples + bs_hiv,  data=risk
plot(fitted(nb.reg.ext),rstandard(nb.reg.ext));abline(h=0)
```

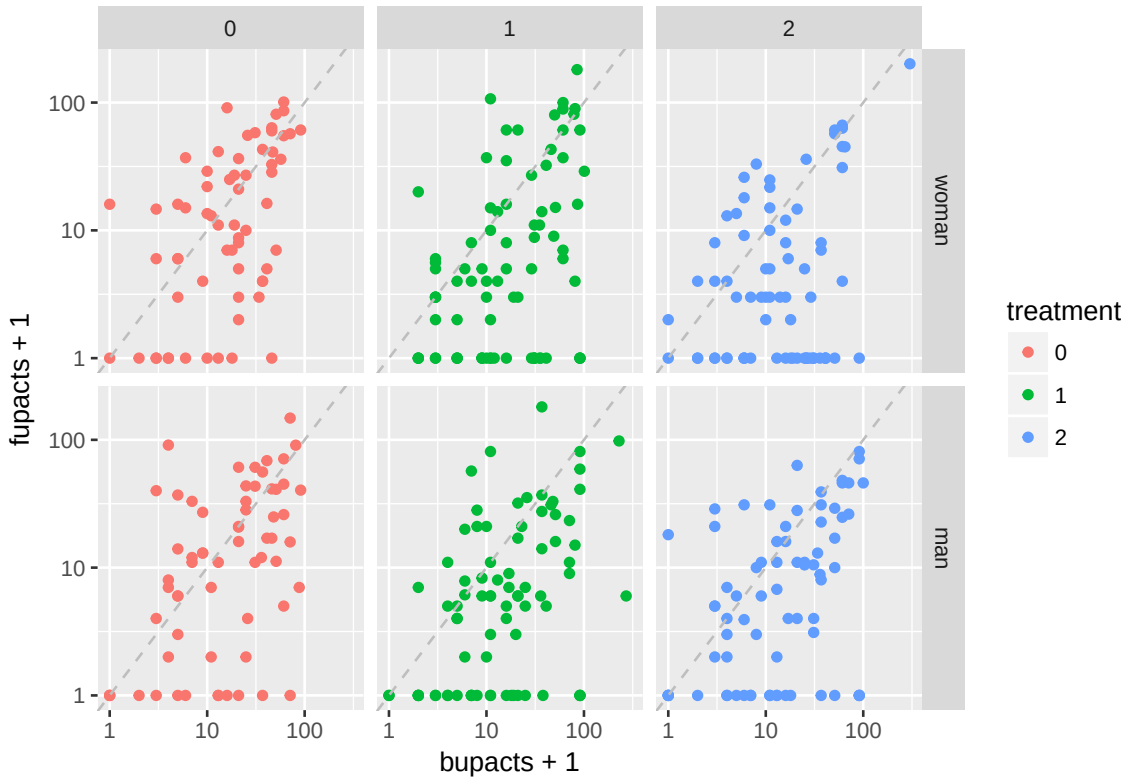


4. These data include responses from both men and women from the participating couples. Does this give you any concern with regard to our modeling assumptions?

```
ggplot(risky_behaviors)+geom_violin()+geom_jitter()+
  aes(x=treatment,y=fupacts+1,color=treatment)+scale_y_log10()+facet_grid(sex~.)
```



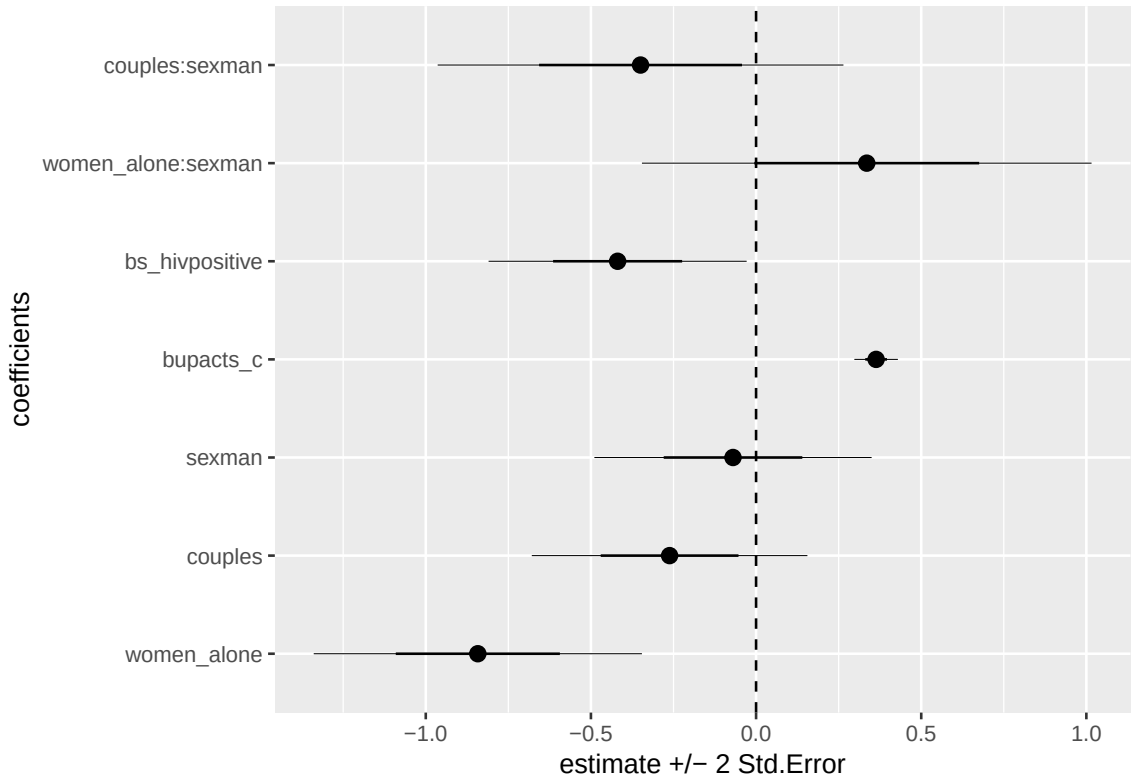

```
ggplot(risky_behaviors)+geom_point()+
  aes(x=bupacts+1,y=fupacts+1,color=treatment)+scale_x_log10()+scale_y_log10()+facet_grid(sex~treatment)
```




```
qpoi.reg.ext2 <- glm(round(fupacts) ~ (women_alone +couples )* sex + bupacts_c  + bs_hiv, family=quasip
display(qpoi.reg.ext2)
```

```
## glm(formula = round(fupacts) ~ (women_alone + couples) * sex +
##     bupacts_c + bs_hiv, family = quasipoisson, data = risky_behaviors)
##                     coef.est coef.se
## (Intercept)          3.15     0.15
## women_alone         -0.84     0.25
## couples             -0.26     0.21
## sexman              -0.07     0.21
## bupacts_c            0.36     0.03
## bs_hivpositive      -0.42     0.20
## women_alone:sexman   0.34     0.34
## couples:sexman      -0.35     0.31
## ---
##   n = 434, k = 8
##   residual deviance = 10086.8, null deviance = 13298.6 (difference = 3211.8)
##   overdispersion parameter = 30.4
```

```
coefplot_my(qpoi.reg.ext2)
```



Comparing logit and probit:

Take one of the data examples from Chapter 5. Fit these data using both logit and probit model. Check that the results are essentially the same (after scaling by factor of 1.6)

Motor Trend Car Road Tests.

The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles

```
?mtcars
model_l <- glm(formula= vs ~ wt+ disp , data=mtcars, family=binomial(link = "logit"))
model_p <- glm(formula= vs ~ wt+ disp , data=mtcars, family=binomial(link = "probit"))

coef(model_l)/1.6
```

```
## (Intercept)          wt        disp
##  1.00537038  1.01647078 -0.02152108
```

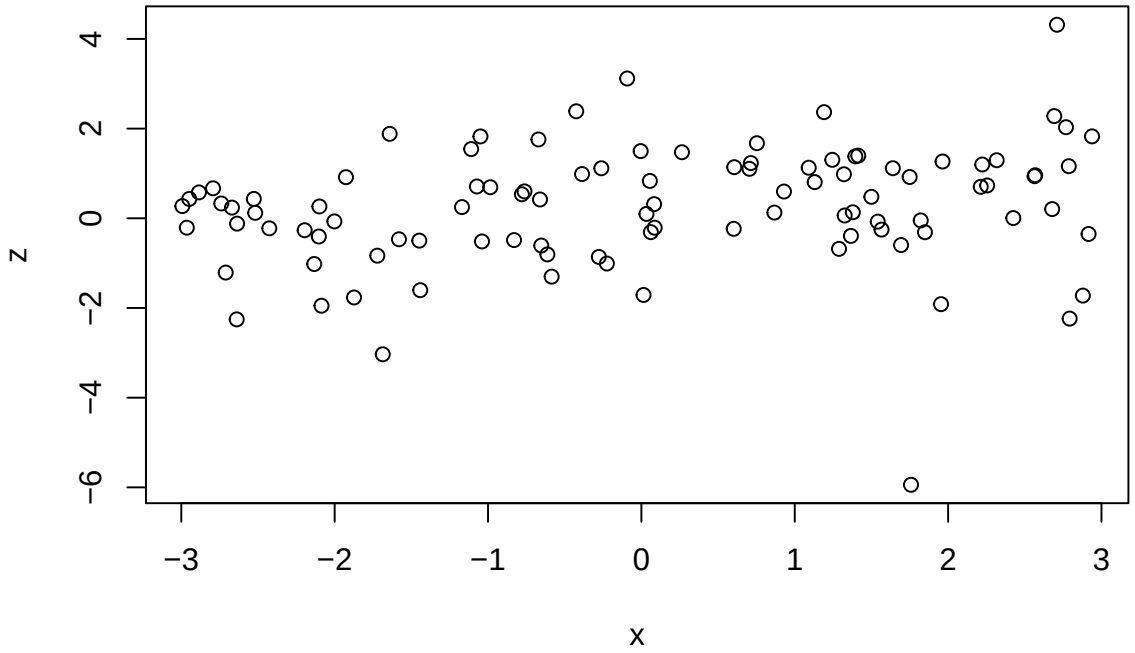
```
coef(model_p)
```

```
## (Intercept)          wt        disp
##  1.08140156  0.90605490 -0.01990869
```

Comparing logit and probit:

construct a dataset where the logit and probit models give different estimates.

```
set.seed(12345)
n<- 100
b0<- 0
b1<- 0.3
x<-runif(n,-3,3)
z<- b0+b1*x+rt(n,df=3)
plot(x,z)
```




```
y<- 1*(z>0)
xr<- c(x,1)
yr<- c(y,-8)
model_l <- glm(formula=y~x , family=binomial(link = "logit"))
model_p <- glm(formula=y~x , family=binomial(link = "probit"))

coef(model_l)/1.6
```

```
## (Intercept)           x
##   0.2817165   0.1317535
```

```
coef(model_p)
```

```
## (Intercept)           x
##   0.2794619   0.1297090
```

Tobit model for mixed discrete/continuous data:

experimental data from the National Supported Work example are available in the folder `lalonde`. Use the treatment indicator and pre-treatment variables to predict post-treatment (1978) earnings using a tobit model. Interpret the model coefficients.

- sample: 1 = NSW; 2 = CPS; 3 = PSID.
- treat: 1 = experimental treatment group (NSW); 0 = comparison group (either from CPS or PSID) - Treatment took place in 1976/1977.
- age = age in years
- educ = years of schooling
- black: 1 if black; 0 otherwise.
- hisp: 1 if Hispanic; 0 otherwise.
- married: 1 if married; 0 otherwise.
- nodegree: 1 if no high school diploma; 0 otherwise.
- re74, re75, re78: real earnings in 1974, 1975 and 1978
- educ_cat = 4 category education variable (1=<hs, 2=hs, 3=sm college, 4=college)

```
# create factor variables
nsw$sample <- factor(nsw$sample, labels=c("NSW", "CPS", "PSID"))
nsw$black  <- factor(nsw$black)
nsw$hisp   <- factor(nsw$hisp)
nsw$nodegree  <- factor(nsw$nodegree)
nsw$married   <- factor(nsw$married)
nsw$treat     <- factor(nsw$treat)
nsw$educ_cat4 <- factor(nsw$educ_cat4, labels=c("less than high school", "high school", "sm college", "

nsw$c_age <- scale(nsw$age, center=TRUE)
nsw$c_educ<- scale(nsw$educ,center=TRUE)
nsw$c_re74<- scale(nsw$re74,center=TRUE)
nsw$c_re75<- scale(nsw$re75,center=TRUE)

# create a dummy variable to represent when re78 is greater than 0
nsw$earn.pos <- ifelse(nsw$re78>0, 1, 0)
ggplot(nsw)+geom_histogram()+aes(x=re78)
ggplot(nsw)+geom_jitter()+geom_violin(alpha=0.3)+aes(x=treat,y=re78-re75)
ggplot(nsw)+geom_jitter(alpha=0.3)+aes(x=c_age,y=re78)
ggplot(nsw)+geom_jitter(alpha=0.3)+aes(x=educ_cat4,y=re78-re75)+facet_grid(treat~.)
ggplot(nsw)+geom_jitter(alpha=0.3)+aes(x=c_age,y=re78-re75)+facet_grid(treat~.)+geom_smooth(method="lm"

 fit_tobit <-vglm(re78-re75 ~ treat+c_age+  educ_cat4+ c_re75+black+hisp+married,
                  family= tobit(Lower = 0, Upper=25564 ), data=nsw)
summary( fit_tobit)

 fit_tobit0 <-vglm(re78 ~ c_age+  educ_cat4+ c_re75+black+hisp+married,,
                  family= tobit(Lower = 0, Upper=25564 ), data=nsw)

nsw$yhat <- fitted(fit_tobit)[,1]
nsw$rr <- resid(fit_tobit, type = "response")
ggplot(nsw)+geom_point()+aes(y=rr,x=yhat)+geom_hline(yintercept=0)


(p <- pchisq(2 * (logLik(fit_tobit) - logLik(fit_tobit0)), df = 1, lower.tail = FALSE))

 fit_lm<-lm(re78 ~ treat+c_age+  educ_cat4+ c_re75, data=nsw)
 summary( fit_lm)
 plot(fitted(fit_lm),resid(fit_lm))
round( summary( fit_lm)$coefficient,2)
round( summary( fit_tobit)@coef3,2)
```

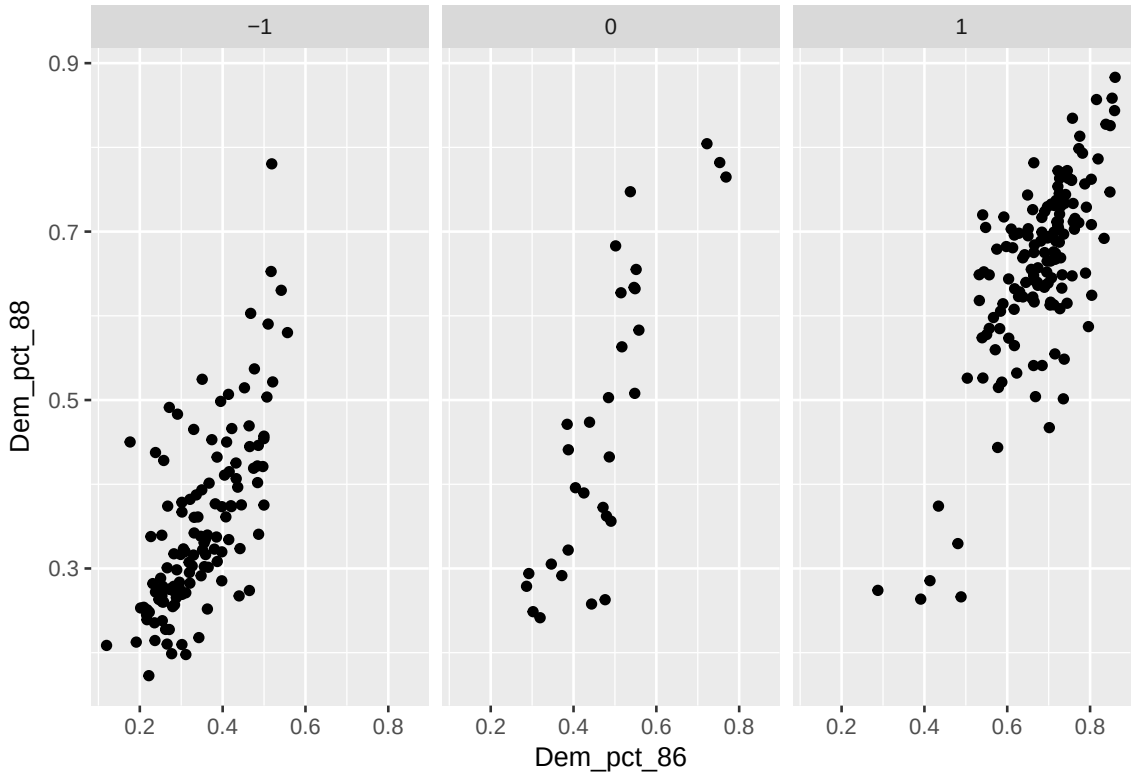
```
nsw$re78_cens <- nsw$re78
nsw$re78_cens [nsw$re78>=25564]<-25564

library( censReg )
estResult <- censReg( re78_cens ~ treat+c_age+black+hisp+married+educ_cat4+ c_re75,  data = nsw ,right =
summary( estResult)
nsw$yhat <- predict(estResult)[,1]
nsw$rr <- resid(fit_tobit, type = "response")
ggplot(nsw)+geom_point()+aes(y=rr,x=yhat)+geom_hline(yintercept=0)

fit_lm<-lm(re78-re75 ~ treat+c_age+  educ_cat4+ c_re74+black+hisp+married, data=nsw)
 summary( fit_lm)
```

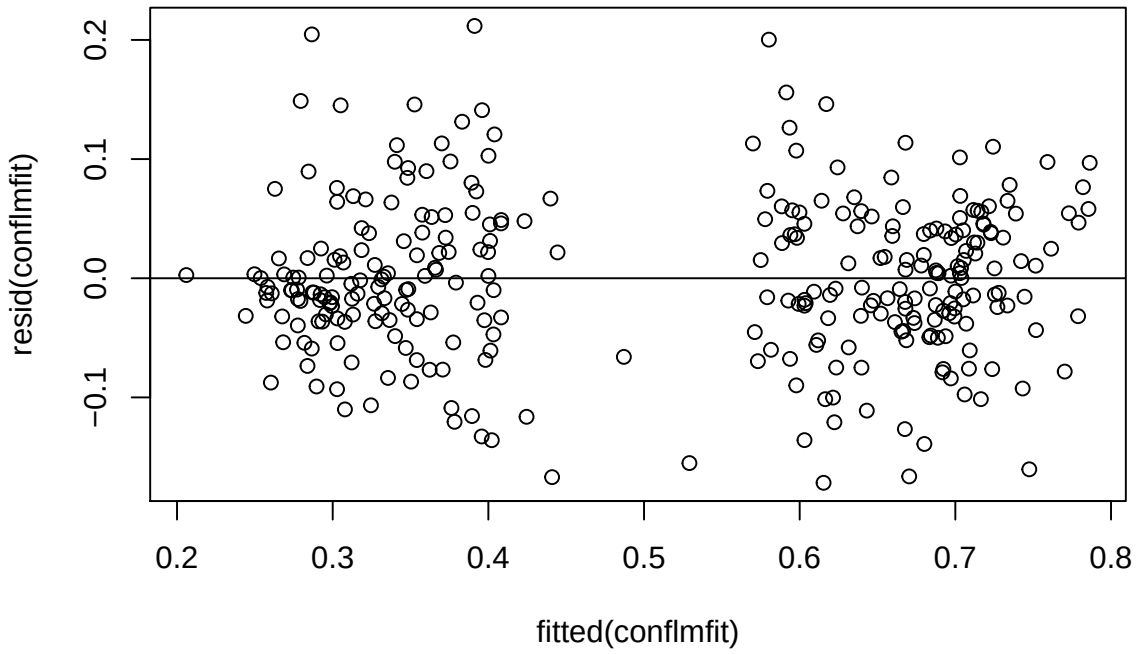
Robust linear regression using the t model:

The csv file `congress` has the votes for the Democratic and Republican candidates in each U.S. congressional district in between 1896 and 1992, along with the parties' vote proportions and an indicator for whether the incumbent was running for reelection. For your analysis, just use the elections in 1986 and 1988 that were contested by both parties in both years.



1. Fit a linear regression (with the usual normal-distribution model for the errors) predicting 1988 Democratic vote share from the other variables and assess model fit.

```
conflmfit<-lm(Dem_pct_88~Dem_pct_86*incumbent_88,data=cong_dt)
plot(fitted(conflmfit),resid(conflmfit));abline(h=0)
```



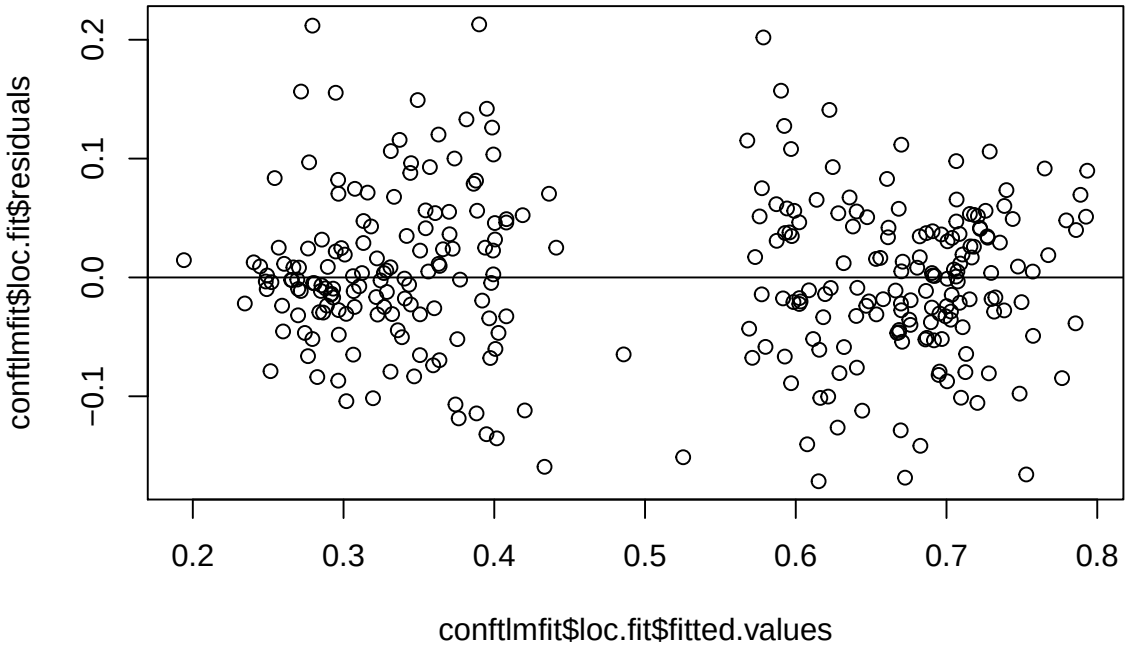
2. Fit a t-regression model predicting 1988 Democratic vote share from the other variables and assess model fit; to fit this model in R you can use the `vglm()` function in the VGLM package or `tlm()` function in the hett package.

```
conftlmfit<-hett::tlm(Dem_pct_88~Dem_pct_86*incumbent_88,data=cong_dt)
summary(conftlmfit)
```

```
## Location model :
##
## Call:
## hett::tlm(lform = Dem_pct_88 ~ Dem_pct_86 * incumbent_88, data = cong_dt)
##
## Residuals:
##      Min        1Q    Median        3Q       Max
## -0.17148  -0.03369  -0.00179   0.04096   0.21291
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               0.18963    0.01710  11.087  < 2e-16 ***
## Dem_pct_86                0.59571    0.03362  17.716  < 2e-16 ***
## incumbent_88              0.06293    0.01827   3.445 0.000652 ***
## Dem_pct_86:incumbent_88   0.03281    0.03570   0.919 0.358838
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Scale parameter(s) as estimated below)
##
##
## Scale Model :
##
## Call:
## hett::tlm(lform = Dem_pct_88 ~ Dem_pct_86 * incumbent_88, data = cong_dt)
##
## Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -1.9994  -1.6516  -0.6672   1.2777   4.9112
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -6.040      0.114  -53.01   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Scale parameter taken to be  2 )
##
##
## Est. degrees of freedom parameter:  3
## Standard error for d.o.f:  NA
## No. of iterations of model : 10 in 0.006
## Heteroscedastic t Likelihood : 400.0771
```

```
plot(conftlmfit$loc.fit$fitted.values,conftlmfit$loc.fit$residuals);abline(h=0)
```



3. Which model do you prefer?

Robust regression for binary data using the robit model:

Use the same data as the previous example with the goal instead of predicting for each district whether it was won by the Democratic or Republican candidate.

1. Fit a standard logistic or probit regression and assess model fit.

```
cong_dt$democrat_won <- 1*(cong_dt$Dem_pct_88>0.5)
logit_fit <- glm(democrat_won~Dem_pct_86*incumbent_86,data=cong_dt,family=binomial(link="logit"))
display(logit_fit)
```

```
## glm(formula = democrat_won ~ Dem_pct_86 * incumbent_86, family = binomial(link = "logit"),
##     data = cong_dt)
```

```
##                          coef.est coef.se
## (Intercept)              -16.00     2.92
## Dem_pct_86                32.33     5.77
## incumbent_86               0.59     3.13
## Dem_pct_86:incumbent_86   -0.99     6.16
## ---
##   n = 308, k = 4
##   residual deviance = 85.7, null deviance = 425.4 (difference = 339.7)
```

```
probit_fit <- glm(democrat_won~Dem_pct_86*incumbent_86,data=cong_dt,family=binomial(link="probit"))
display(probit_fit)
```

```
## glm(formula = democrat_won ~ Dem_pct_86 * incumbent_86, family = binomial(link = "probit"),
##     data = cong_dt)
##                          coef.est coef.se
## (Intercept)              -7.47     1.24
## Dem_pct_86               15.18     2.48
## incumbent_86              0.85     1.32
## Dem_pct_86:incumbent_86 -1.50     2.60
## ---
##   n = 308, k = 4
##   residual deviance = 91.0, null deviance = 425.4 (difference = 334.4)
```

2. Fit a robit regression and assess model fit.

The robit link function can be found in `glmx` package. We will use `gosset(2)` link function.

```
library(glmx)
```

```
##
## Attaching package: 'glmx'
```

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

```
t2_fit <- glm(democrat_won~Dem_pct_86*incumbent_86,data=cong_dt,family=binomial(link=gosset(2)))
display(t2_fit)
```

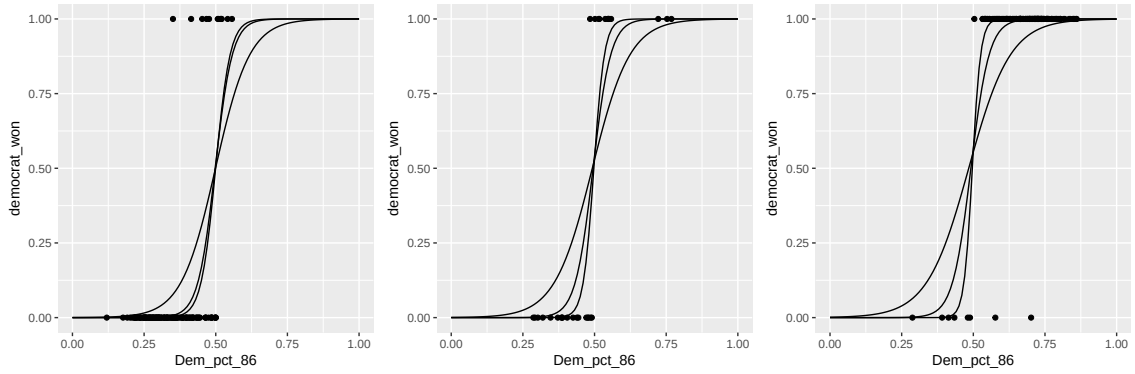
```
## glm(formula = democrat_won ~ Dem_pct_86 * incumbent_86, family = binomial(link = gosset(2)),
##     data = cong_dt)
##                          coef.est coef.se
## (Intercept)              -26.92     7.96
## Dem_pct_86                54.11    15.97
## incumbent_86              -7.23     8.63
## Dem_pct_86:incumbent_86   14.67    17.32
## ---
##   n = 308, k = 4
##   residual deviance = 76.5, null deviance = 425.4 (difference = 348.9)
```

3. Which model do you prefer?

```
fun1<- function(x,incumbent=0){invlogit(coef(logit_fit)[1]+coef(logit_fit)[2]*x+coef(logit_fit)[3]*incu
fun2<- function(x,incumbent=0){invlogit(coef(probit_fit)[1]+coef(probit_fit)[2]*x+coef(probit_fit)[3]*i
fun3<- function(x,incumbent=0){invlogit(coef(t2_fit)[1]+coef(t2_fit)[2]*x+coef(t2_fit)[3]*incumbent+coe

library(gridExtra)
grid.arrange(
ggplot(cong_dt[cong_dt$incumbent_86==-1,])+
```

```
  geom_point()+aes(x=Dem_pct_86,y=democrat_won)+
  stat_function(fun=fun1,args=list(incumbent=-1))+
  stat_function(fun=fun2,args=list(incumbent=-1))+
  stat_function(fun=fun3,args=list(incumbent=-1))+xlim(0,1)
,
ggplot(cong_dt[cong_dt$incumbent_86==0,])+
  geom_point()+aes(x=Dem_pct_86,y=democrat_won)+
  stat_function(fun=fun1,args=list(incumbent=0))+
  stat_function(fun=fun2,args=list(incumbent=0))+
  stat_function(fun=fun3,args=list(incumbent=0))+xlim(0,1)
,
ggplot(cong_dt[cong_dt$incumbent_86==1,])+
  geom_point()+aes(x=Dem_pct_86,y=democrat_won)+
  stat_function(fun=fun1,args=list(incumbent=1))+
  stat_function(fun=fun2,args=list(incumbent=1))+
  stat_function(fun=fun3,args=list(incumbent=1))+xlim(0,1)
,ncol=3)
```



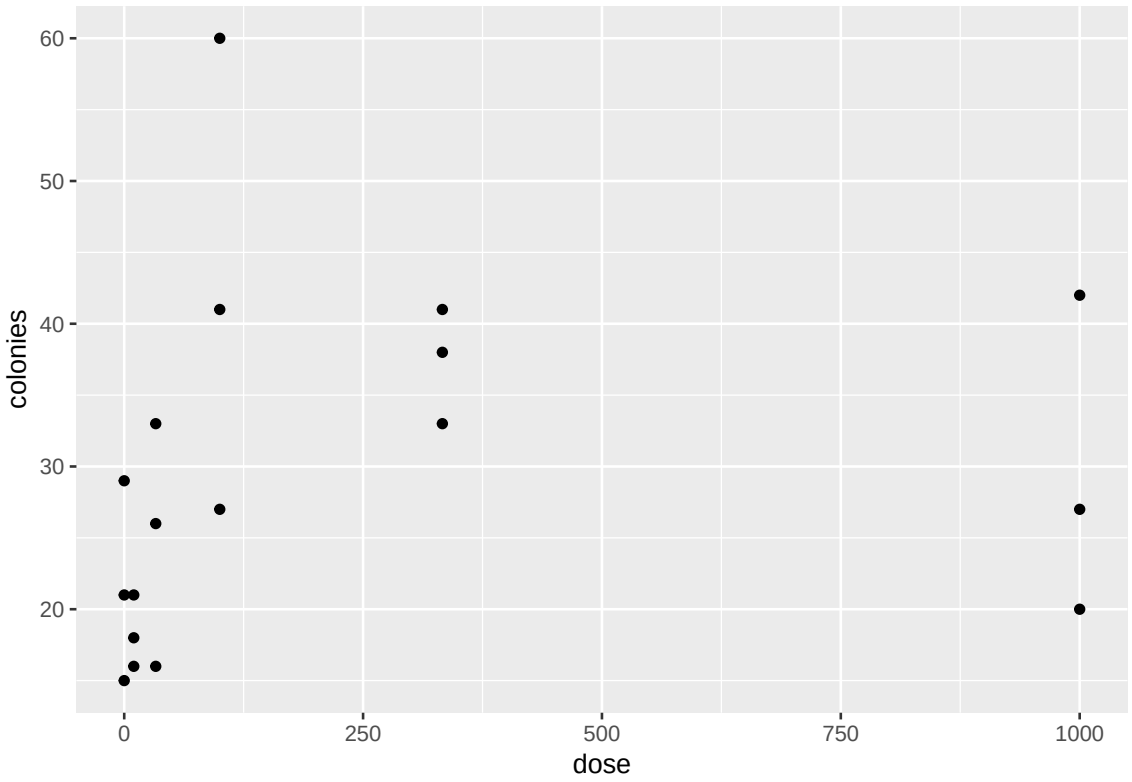
Salmonellla

The `salmonella` data was collected in a salmonella reverse mutagenicity assay. The predictor is the dose level of quinoline and the response is the numbers of revertant colonies of TA98 salmonella observed on each of three replicate plates. Show that a Poisson GLM is inadequate and that some overdispersion must be allowed for. Do not forget to check out other reasons for a high deviance.

```
data(salmonella)
?salmonella
```

When you plot the data you see that the number of colonies as a function of dose is not monotonic especially around the dose of 1000.

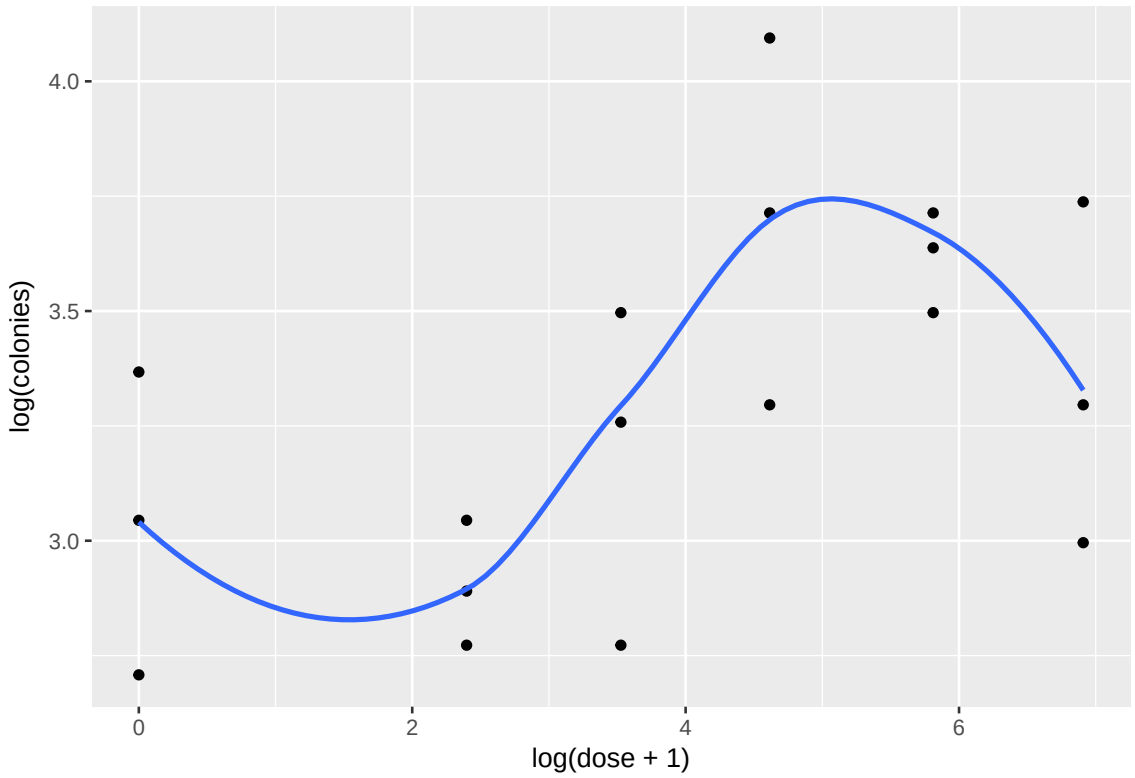
```
ggplot(salmonella)+geom_point()+aes(x=dose,y=colonies)
```



Since we are fitting log linear model we should look at the data on log scale. Also becase the dose is not equally spaced on the raw scale it may be better to plot it on the log scale as well.

```
ggplot(salmonella)+geom_point()+
  aes(x=log(dose+1),y=log(colonies))+
  geom_smooth(se=FALSE)
```

```
## `geom_smooth()` using method = 'loess'
```



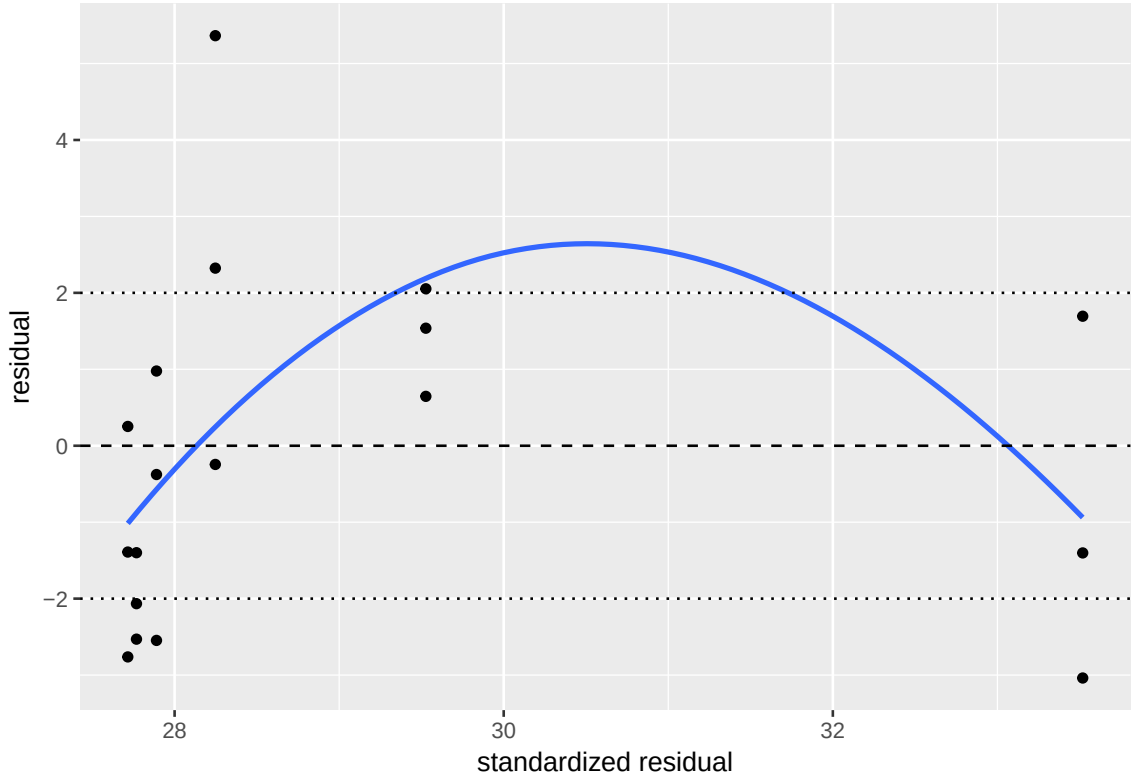
This shows that the trend is not monotonic. Hence when you fit the model and look at the residual you will see a trend.

```
salmonella.pois = glm(colonies~dose, family=poisson, data=salmonella)
summary(salmonella.pois)
```

```
##
## Call:
## glm(formula = colonies ~ dose, family = poisson, data = salmonella)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6482  -1.8225  -0.2993   1.2917   5.1861
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 3.3219950  0.0540292  61.485   <2e-16 ***
## dose        0.0001901  0.0001172   1.622    0.105
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 75.806  on 16  degrees of freedom
## AIC: 172.34
##
## Number of Fisher Scoring iterations: 4
```

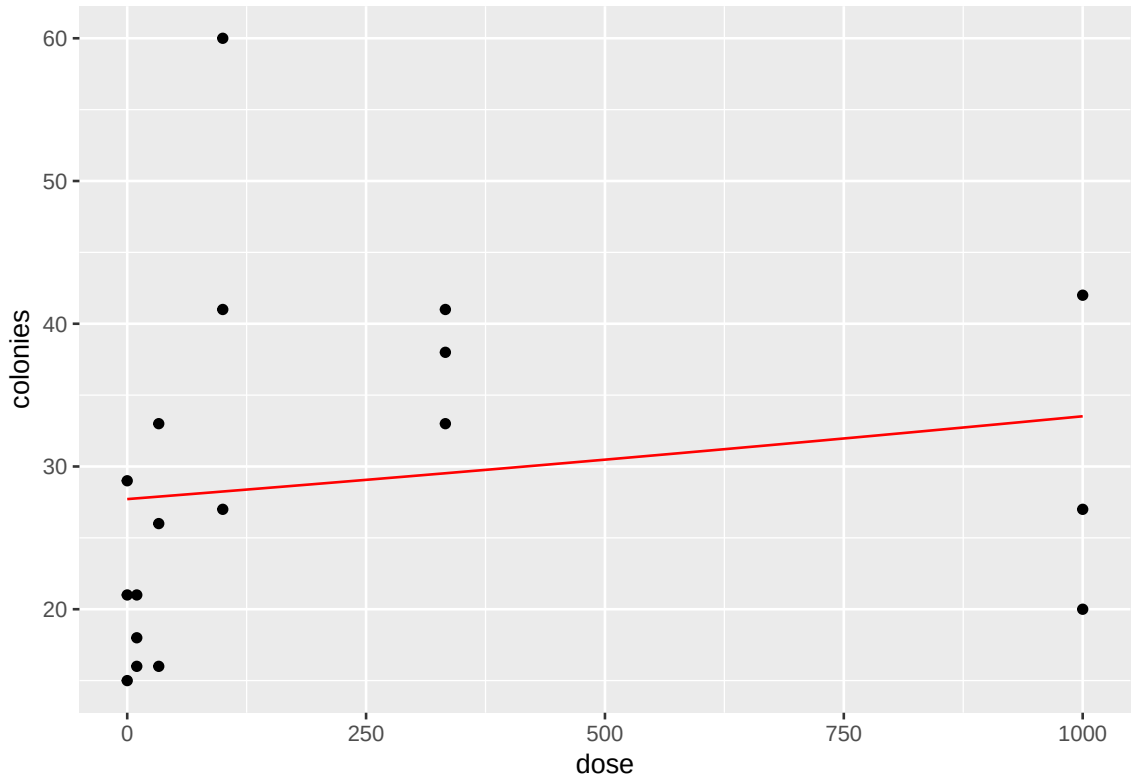
```
salmonella$fitted<-fitted(salmonella.pois)
salmonella$residual<-rstandard(salmonella.pois)
ggplot(salmonella)+geom_point()+aes(x=fitted,y=residual)+
  geom_smooth(se=FALSE,span=2)+
  geom_hline(yintercept=0,linetype="dashed")+
  xlab("standardized residual")+
  geom_hline(yintercept=2,linetype="dotted")+
  geom_hline(yintercept=-2,linetype="dotted")
```

```
## `geom_smooth()` using method = 'loess'
```



The lack of fit is also evident if we plot the fitted line onto the data.

```
test <- function(x) {predict(salmonella.pois,newdata=list(dose=x),type="response")}
ggplot(salmonella)+geom_point()+aes(x=dose,y=colonies)+
  stat_function(fun = test, colour = "red")
```



How do we adress this problem? The serious problem to address is the nonlinear trend of dose ranther than the overdispersion since the line is missing the points. Let's add a bendy line with 3rd order polynomial.

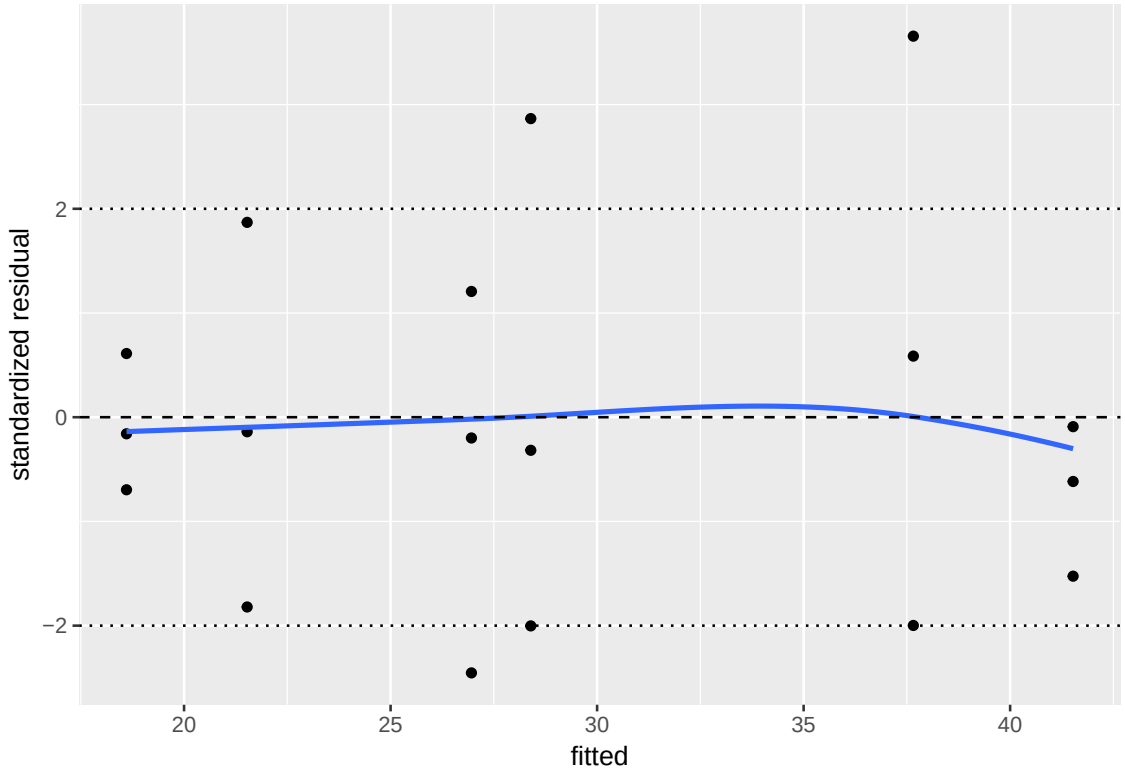
```
salmonella.pois.2 = glm(colonies~poly(log(dose+1),3), family=poisson, data=salmonella)
summary(salmonella.pois.2)
```

```
##
## Call:
## glm(formula = colonies ~ poly(log(dose + 1), 3), family = poisson,
##     data = salmonella)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2851  -1.1842  -0.1635   0.5420   3.3487
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)               3.33135    0.04543  73.321  < 2e-16 ***
## poly(log(dose + 1), 3)1   0.83000    0.20022   4.146 3.39e-05 ***
## poly(log(dose + 1), 3)2 -0.11514    0.19903  -0.578    0.563
## poly(log(dose + 1), 3)3 -0.86047    0.19230  -4.475 7.65e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 37.347  on 14  degrees of freedom
## AIC: 137.88
##
```

```
## Number of Fisher Scoring iterations: 4
```

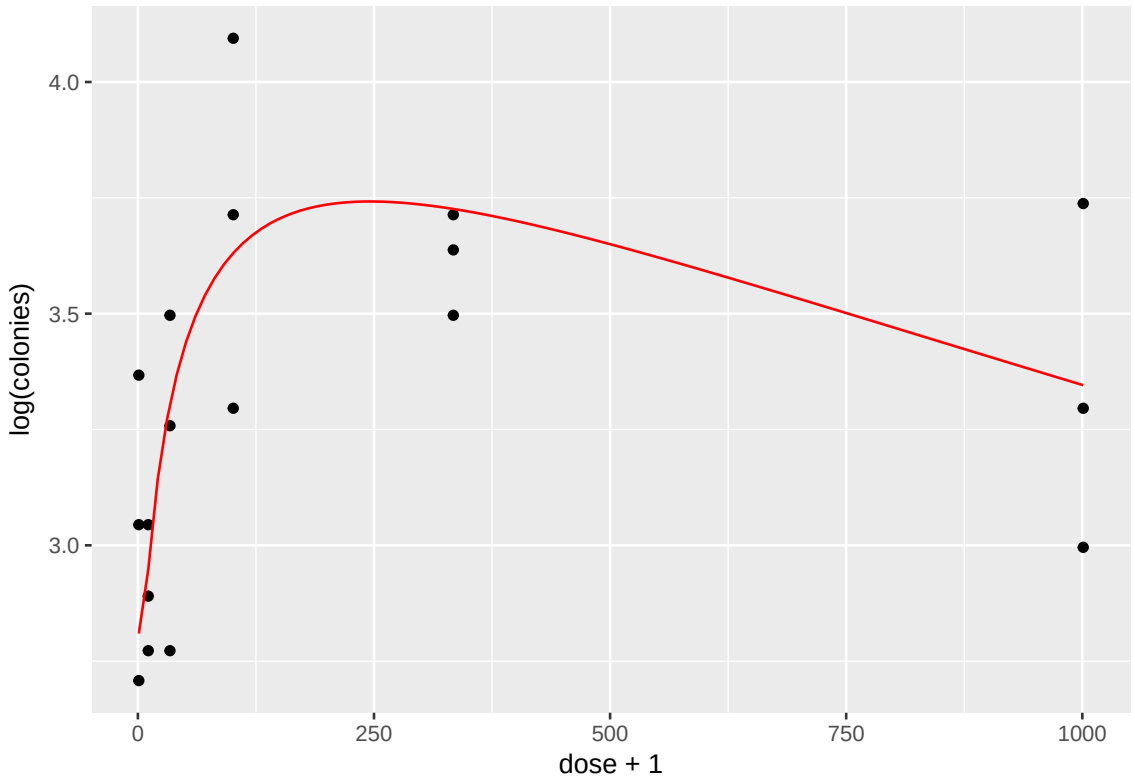
```
salmonella$fitted<-fitted(salmonella.pois.2)
salmonella$residual<- rstandard(salmonella.pois.2)
ggplot(salmonella)+geom_point()+aes(x=fitted,y=residual)+
  geom_smooth(se=FALSE,span=2)+
  geom_hline(yintercept=0,linetype="dashed")+
  xlab("fitted")+
  ylab("standardized residual")+
  geom_hline(yintercept=2,linetype="dotted")+
  geom_hline(yintercept=-2,linetype="dotted")
```

```
## `geom_smooth()` using method = 'loess'
```



The resulting residual looks nice and if you plot it on the raw data. Whether the trend makes real contextual sense will need to be validated but for the given data it looks feasible.

```
test <- function(x) {predict(salmonella.pois.2,newdata=list(dose=x),type="link")}
ggplot(salmonella)+geom_point()+
  aes(x=dose+1,y=log(colonies))+
  stat_function(fun = test, colour = "red")
```



Dispite the fit, the overdispersion still exists so we'd be better off using the quasi Poisson model.

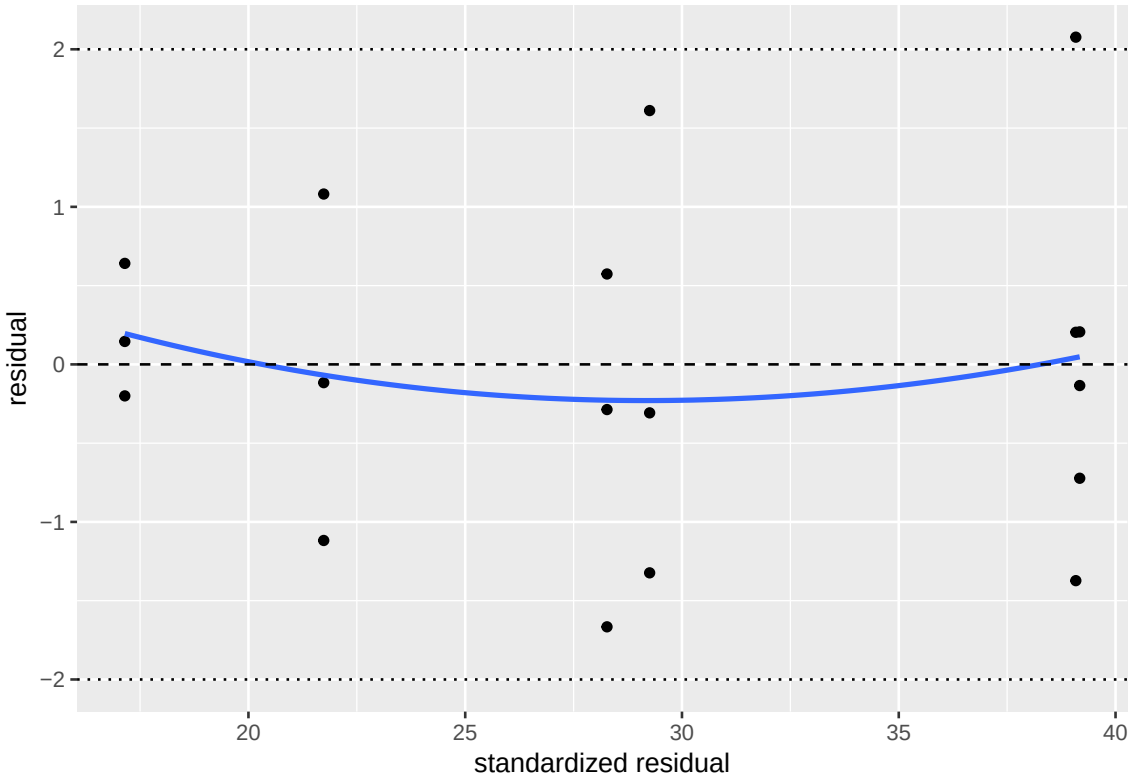
```
salmonella.pois.3 = glm(colonies~poly(log(dose+1),4), family=quasipoisson, data=salmonella)
summary(salmonella.pois.3)
```

```
##
## Call:
## glm(formula = colonies ~ poly(log(dose + 1), 4), family = quasipoisson,
##     data = salmonella)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5150  -0.8689  -0.1737   0.7258   3.0990
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              3.32874    0.07656  43.477 1.83e-15 ***
## poly(log(dose + 1), 4)1  0.85239    0.33735   2.527   0.0253 *
## poly(log(dose + 1), 4)2 -0.08819    0.33479  -0.263   0.7963
## poly(log(dose + 1), 4)3 -0.90342    0.33439  -2.702   0.0181 *
## poly(log(dose + 1), 4)4  0.20216    0.31233   0.647   0.5287
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 2.814972)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 36.156  on 13  degrees of freedom
## AIC: NA
##
```

```
## Number of Fisher Scoring iterations: 4
```

```
salmonella$fitted<-fitted(salmonella.pois.3)
salmonella$residual<-rstandard(salmonella.pois.3)
ggplot(salmonella)+geom_point()+aes(x=fitted,y=residual)+geom_smooth(se=FALSE,span=2)+geom_hline(yinter
```

```
## `geom_smooth()` using method = 'loess'
```



We could also check if we need to include more higher up terms by looking at AIC.

```
AIC(glm(colonies~poly(log(dose+1),1), family=poisson, data=salmonella))
```

```
## [1] 156.1657
```

```
AIC(glm(colonies~poly(log(dose+1),2), family=poisson, data=salmonella))
```

```
## [1] 156.6184
```

```
AIC(glm(colonies~poly(log(dose+1),3), family=poisson, data=salmonella))
```

```
## [1] 137.8836
```

```
AIC(glm(colonies~poly(log(dose+1),4), family=poisson, data=salmonella))
```

```
## [1] 138.6918
```

```
AIC(glm(colonies~poly(log(dose+1),5), family=poisson, data=salmonella))
```

```
## [1] 138.0322
```

The result shows we don't have strong evidence for higher order terms.

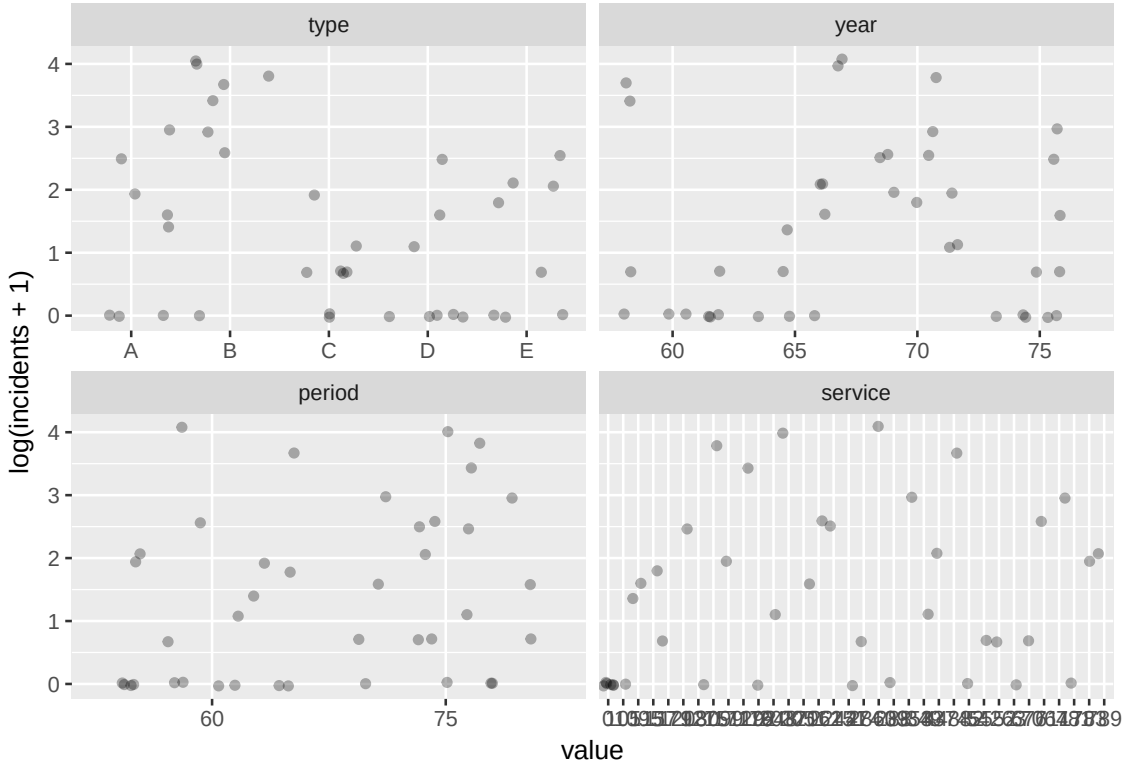
Ships

The `ships` dataset found in the MASS package gives the number of damage incidents and aggregate months of service for different types of ships broken down by year of construction and period of operation.

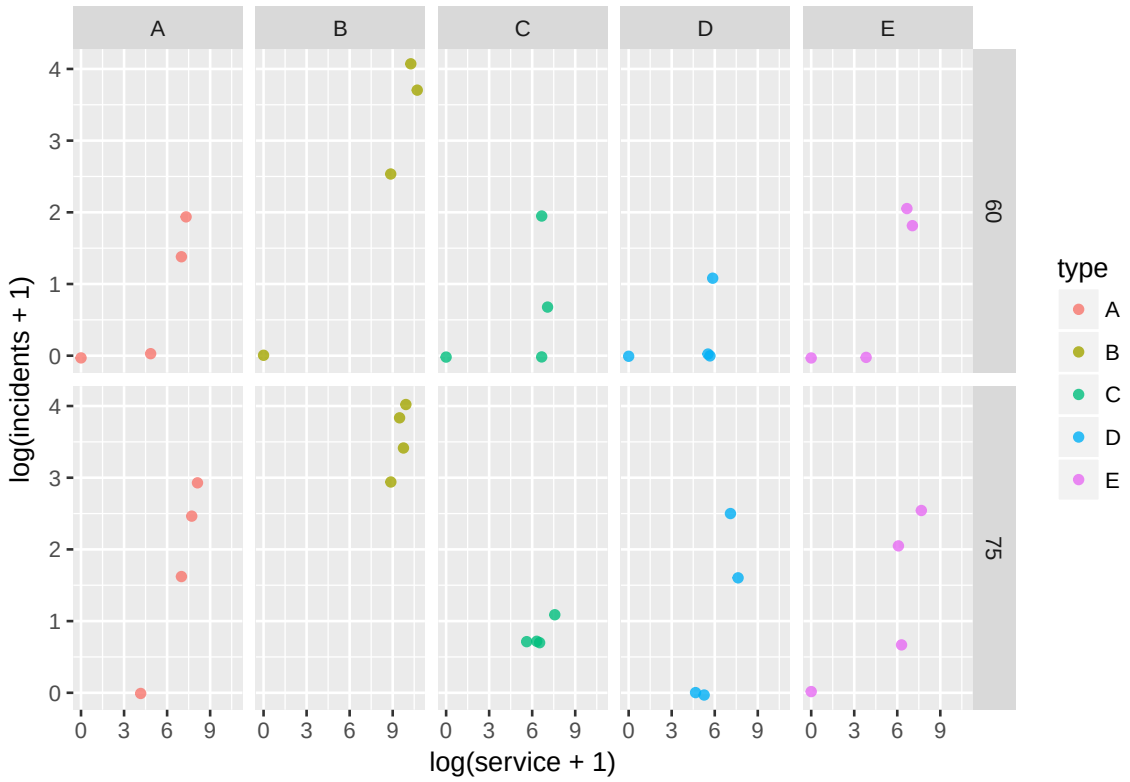
```
data(ships)
?ships
```

Develop a model for the rate of incidents, describing the effect of the important predictors.

```
ggplot(melt(ships,id.vars=c("incidents")))+
  geom_jitter(alpha=0.3)+
  aes(x=value,y=log(incidents+1))+
  facet_wrap(~variable,scale="free_x")
```



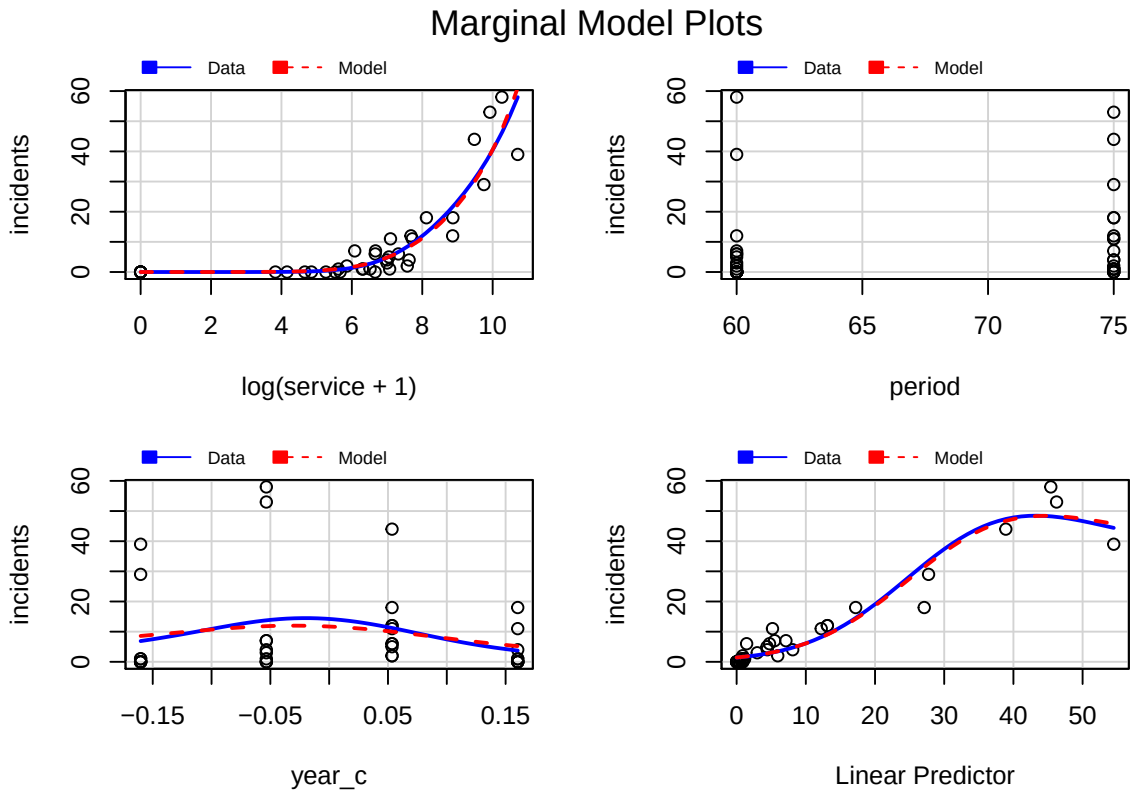

```
ggplot(ships)+
  geom_jitter(alpha=0.8)+
  aes(x=log(service+1),y=log(incidents+1),color=type)+facet_grid(period~type)
```



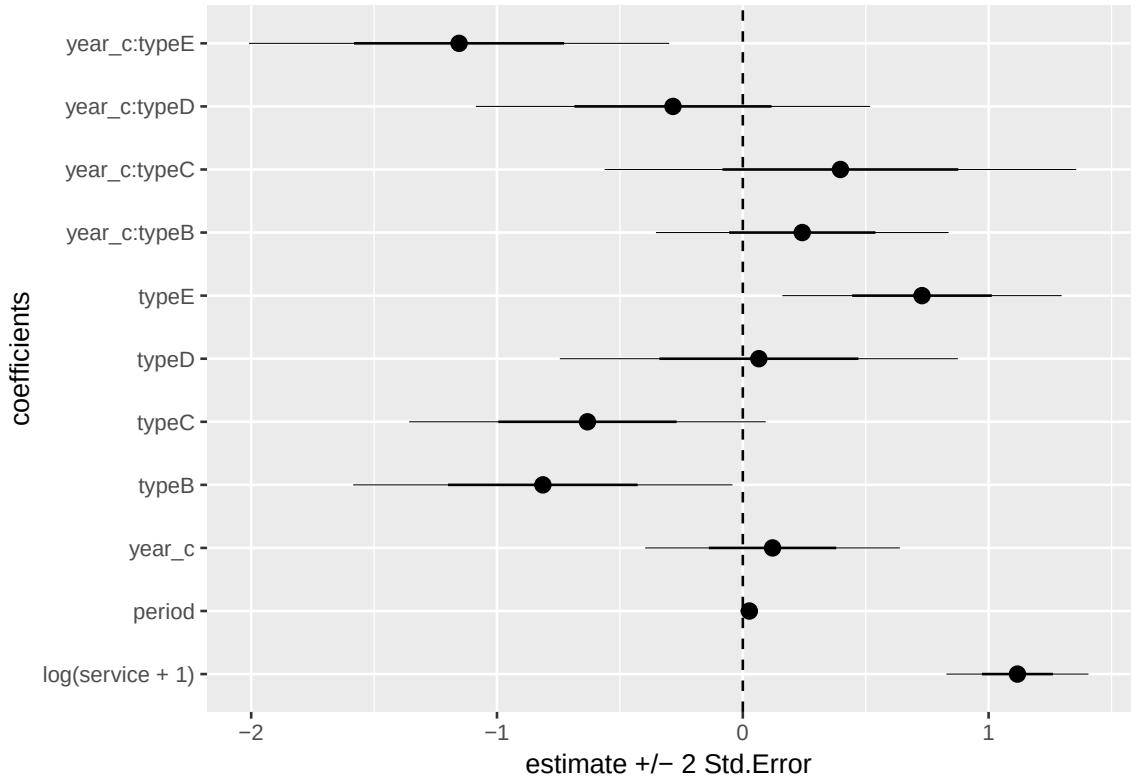

```
ships$year_c<-scale( ships$year,center=TRUE)
modelship<-glm(incidents~log(service+1)+period+year_c*type,family=poisson,data=ships)
marginalModelPlots(modelship)
```

```
## Warning in mmps(...): Splines and/or polynomials replaced by a fitted
## linear combination
```

```
## Warning in mmps(...): Interactions and/or factors skipped
```

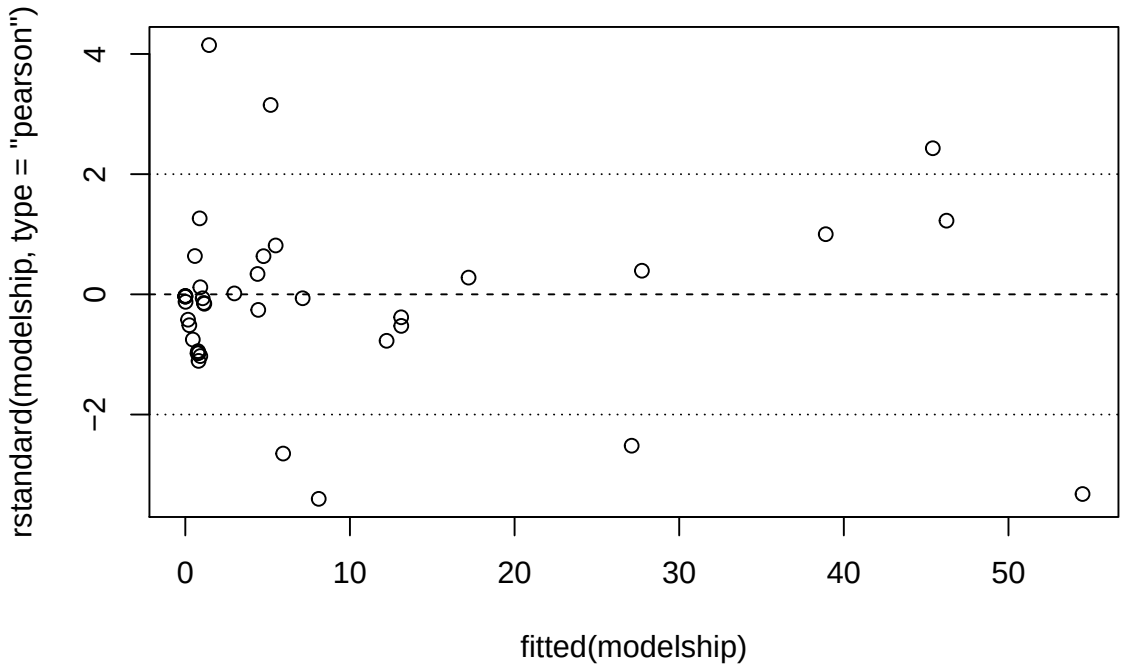



```
coefplot_my(modelship)
```

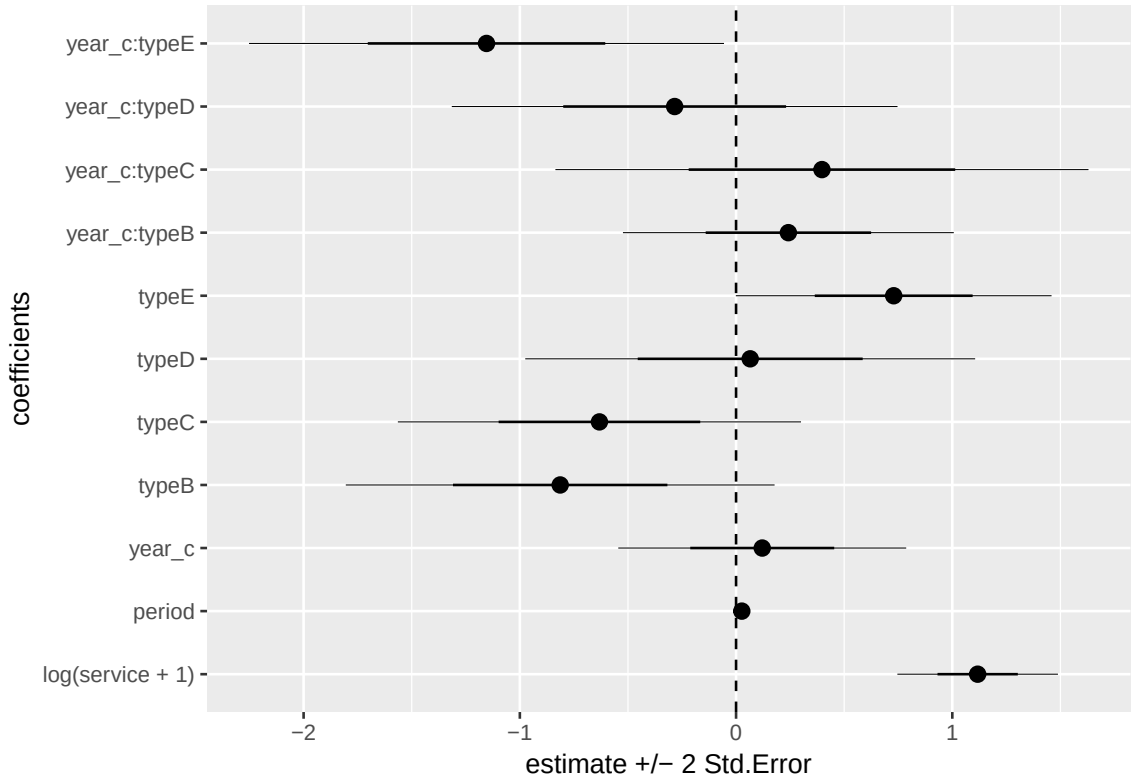


When you look at the residual you see slight signs of over dispersion.

```
plot(fitted(modelship),rstandard(modelship,type="pearson"));abline(h=0,lty=2)
abline(h=-2,lty=3);abline(h=2,lty=3)
```




```
ships$year_c<-scale( ships$year,center=TRUE)
modelship2<-glm(incidents~log(service+1)+period+year_c*type,
                family=quasipoisson,data=ships)
coefplot_my(modelship2)
```



Australian Health Survey

The `dvisits` data comes from the Australian Health Survey of 1977-78 and consist of 5190 single adults where young and old have been oversampled.

```
data(dvisits)
?dvisits
```

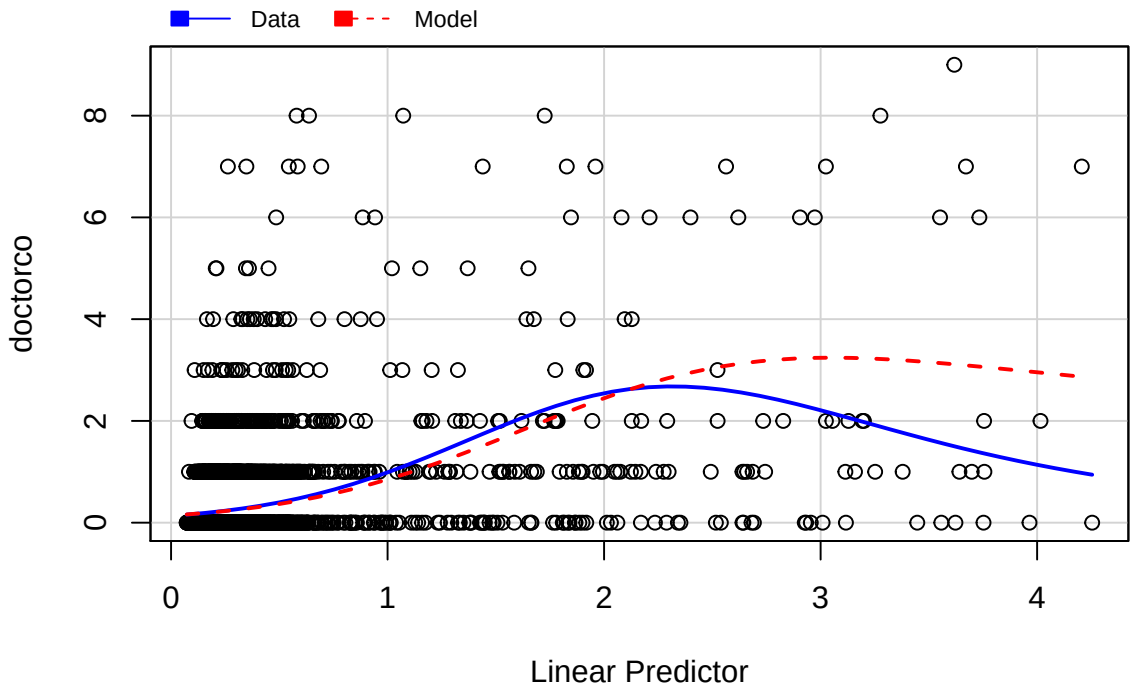
1. Build a Poisson regression model with `doctorco` as the response and `sex`, `age`, `agesq`, `income`, `levyplus`, `freepoor`, `freerepa`, `illness`, `actdays`, `hscore`, `chcond1` and `chcond2` as possible predictor variables. Considering the deviance of this model, does this model fit the data?

Initial model does not seem to capture the high end accurately.

```
moddoc <- glm(doctorco ~ sex +  levyplus + freepoor + freerepa +
                (age+agesq)+illness +income+actdays + hscore +
                chcond1+chcond2,
                family=poisson(),
                data=dvisits)
AIC(moddoc)
```

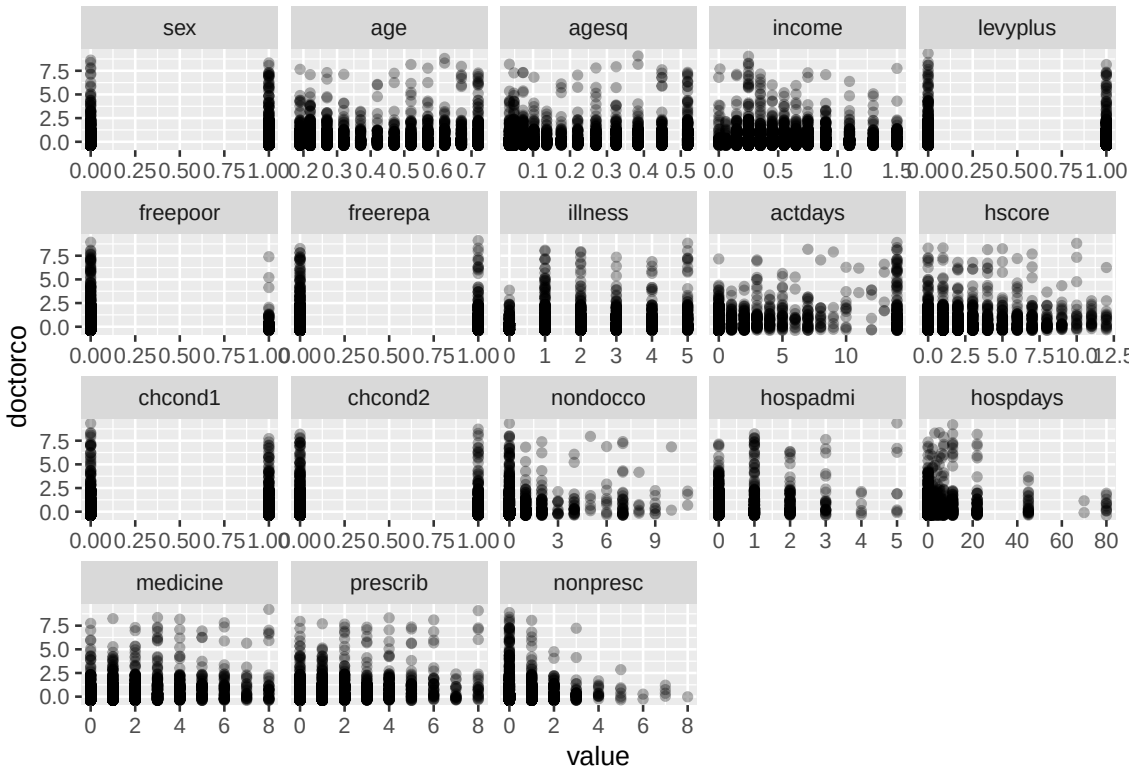
```
## [1] 6737.083
```

```
marginalModelPlot(moddoc)
```



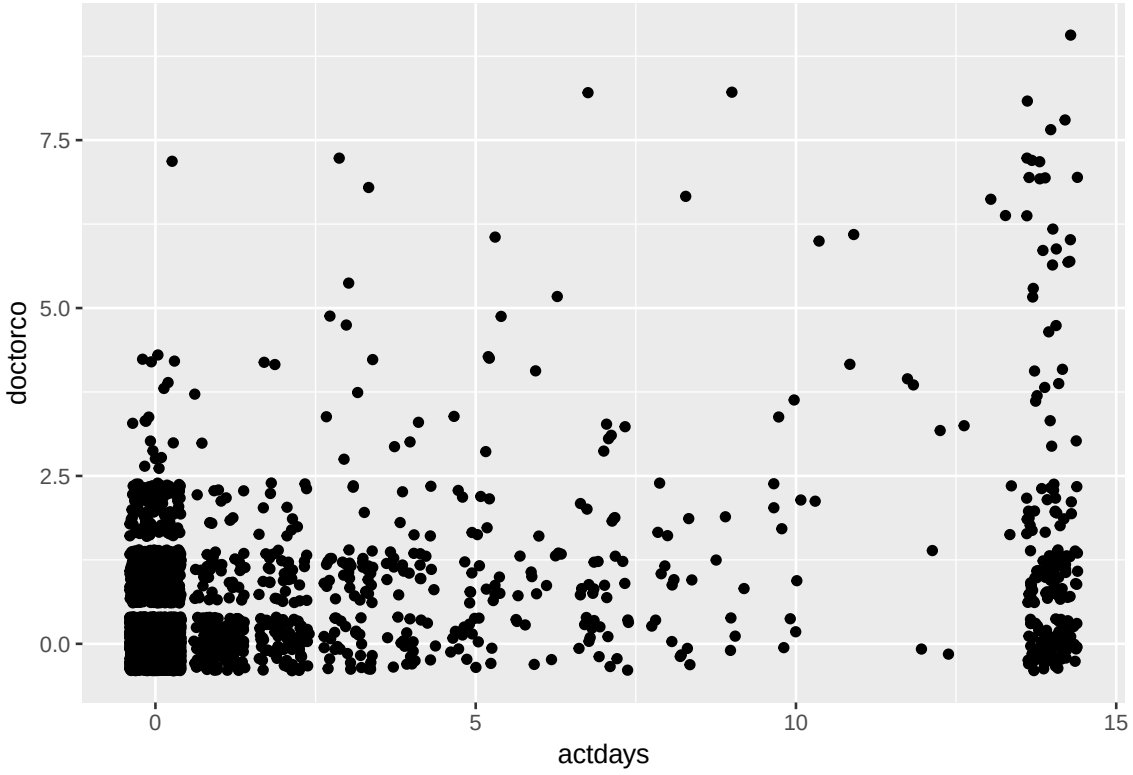
We look at the variables more carefully

```
ggplot(melt(dvisits,id.vars=c("doctorco")))+
  geom_jitter(alpha=0.3)+
  aes(x=value,y=doctorco)+
  facet_wrap(~variable,scale="free_x")
```



When you look at number of days of reduced activity in past two weeks due to illness or injury this shows bimodality.

```
ggplot(dvisits)+geom_jitter()+
  aes(x=actdays,y=doctorco)
```



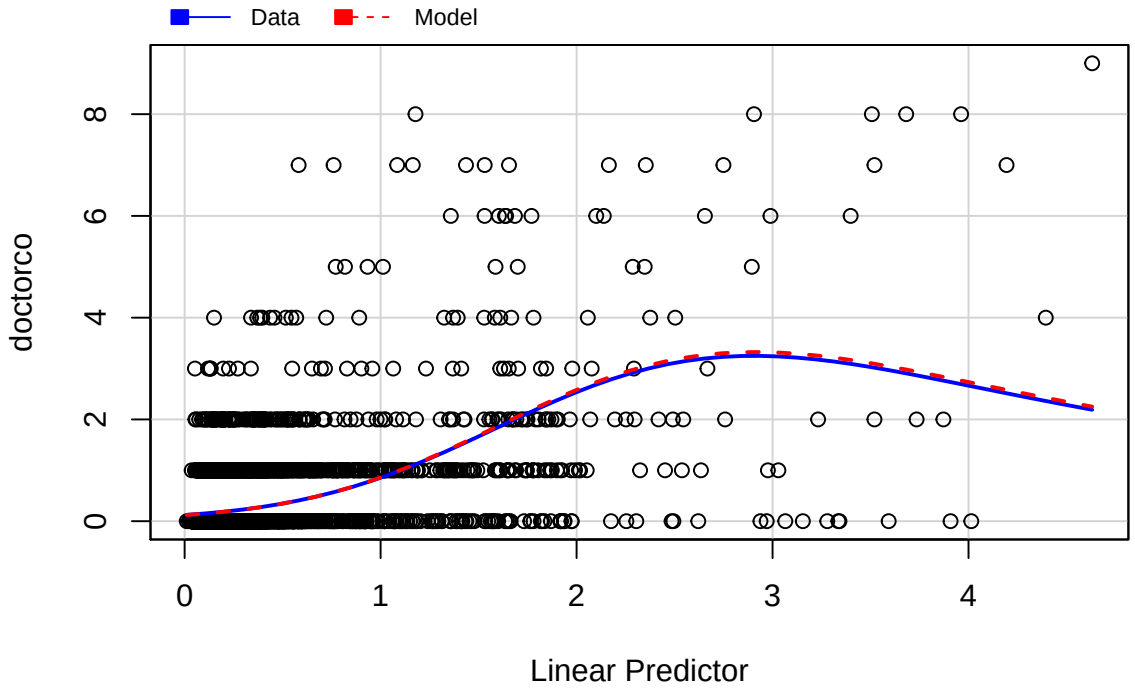
```
dvisits$actdays_cat <- 4*(dvisits$actdays>=14)+3*(dvisits$actdays<14& dvisits$actdays >= 5)+2*(dvisits$a
dvisits$actdays_14<- 1*(dvisits$actdays>=14)
```

We will add indicator to deal with the actdays 14 people.

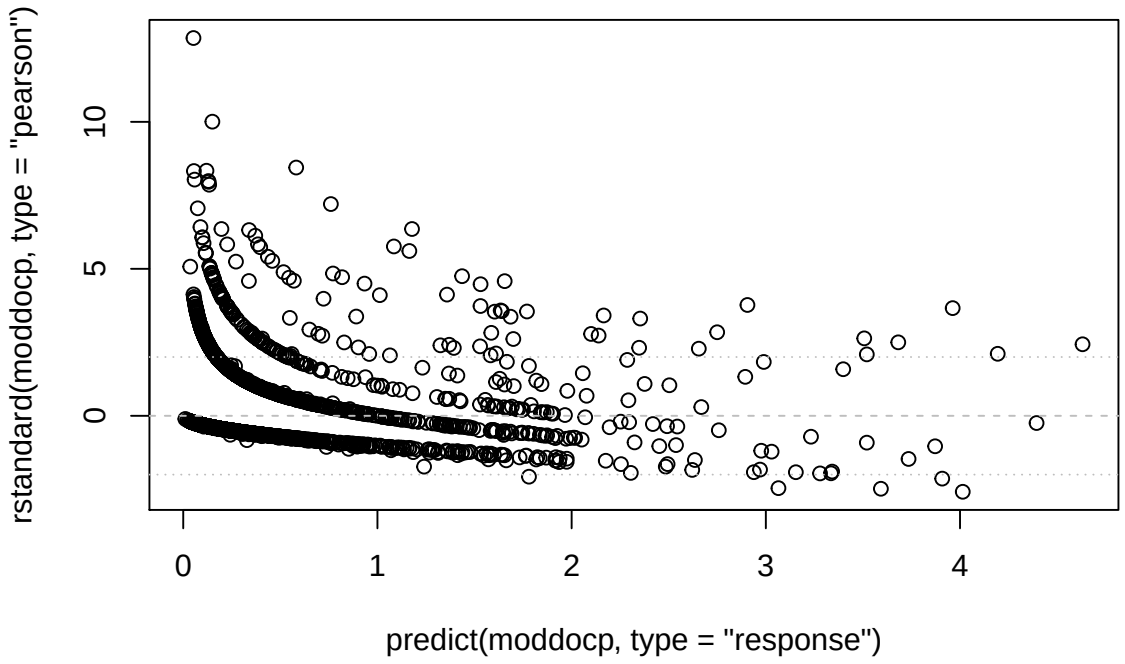
```
moddocp <- glm(doctorco ~ sex +  levyplus * freepoor
                  * freerepa + (age+agesq)*illness *income*actdays_cat *actdays_14* hscore + chcond1+ch
                    data=dvisits)
AIC(moddocp)
```

```
## [1] 6394.304
```

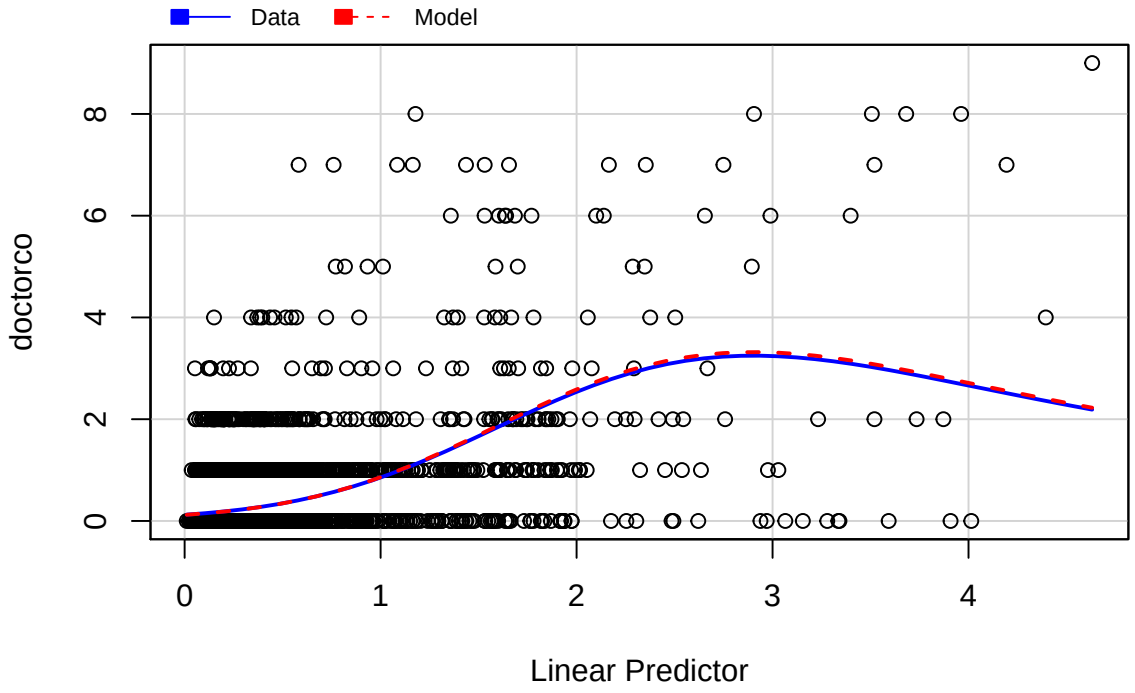
```
marginalModelPlot(moddocp)
```




```
plot(predict(moddocp, type="response"), rstandard(moddocp, type="pearson"));
abline(h=0,lty=2,col="grey")
abline(h=2,lty=3,col="grey")
abline(h=-2,lty=3,col="grey")
```




```
moddocqp <- glm(doctorco ~ sex +  levyplus * freepoor
                  * freerepa + (age+agesq)*illness *income*actdays_cat *actdays_14* hscore + chcond1+ch
                    data=dvisits)
marginalModelPlot(moddocqp)
```




```
library(GGally)
```

```
##
## Attaching package: 'GGally'
## The following object is masked from 'package:faraway':
##
```

```
##     happy
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

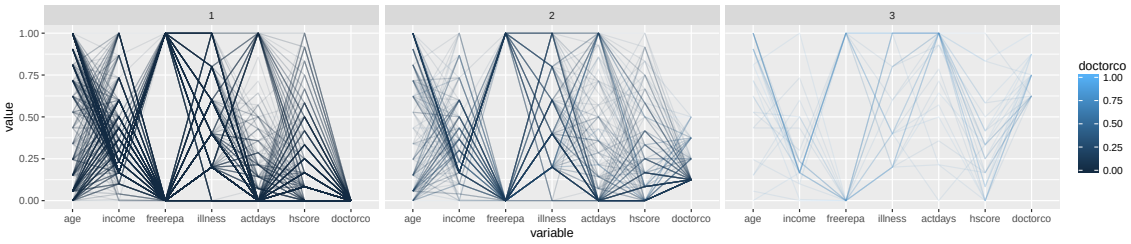
```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

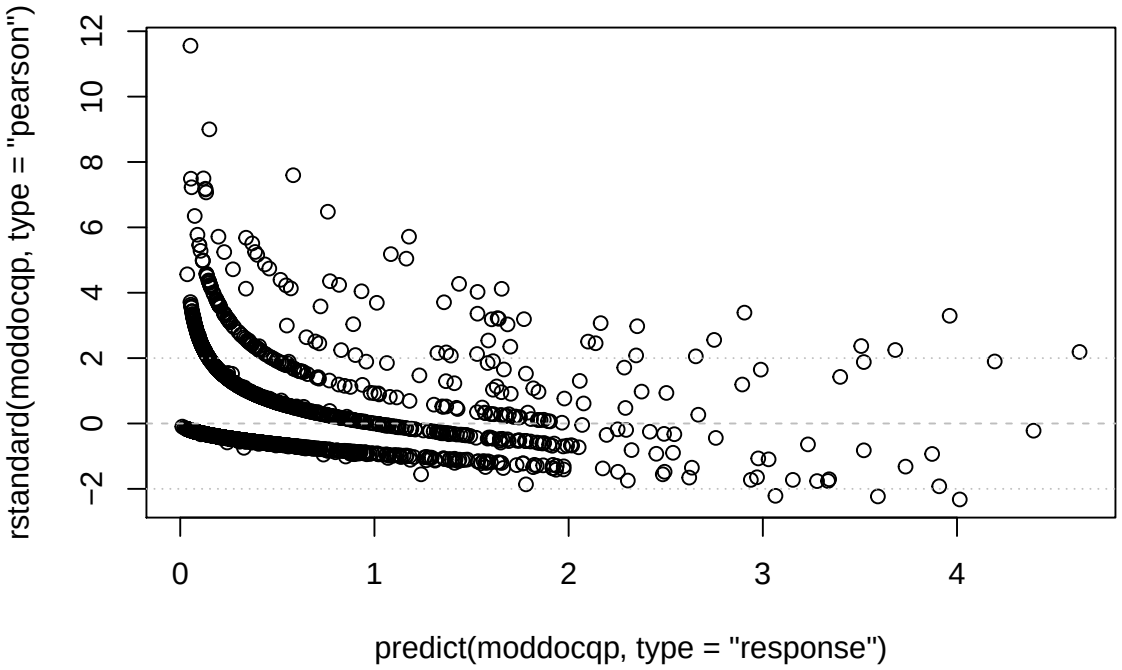
```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
dvisits$doctorco5 <- factor(1*(dvisits$doctorco>5) -1*(dvisits$doctorco<2))
ggparcoord(dvisits[dvisits$doctorco>0,], columns = c(2,4,7,8,9,10,13),  scale = 'uniminmax',alphaLines=0
```



2. Plot the residuals and the fitted values-why are there lines of observations on the plot?

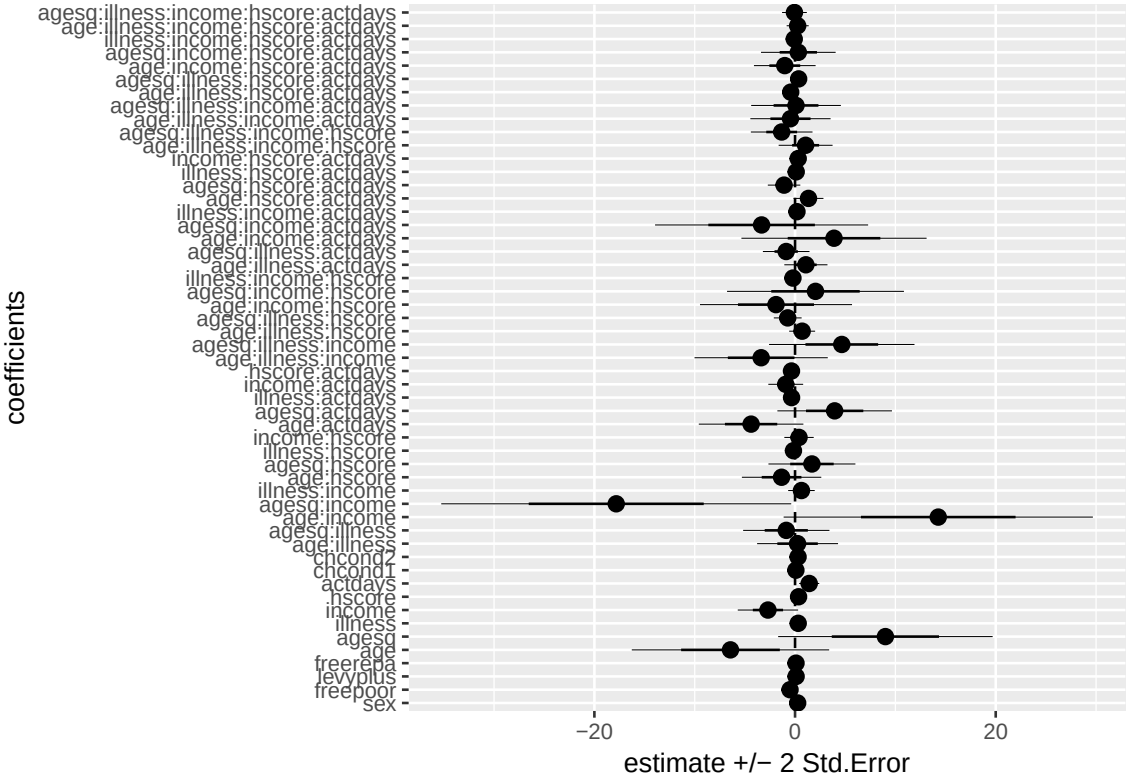
```
plot(predict(moddocqp, type="response"), rstandard(moddocqp, type="pearson"));
abline(h=0,lty=2,col="grey")
abline(h=2,lty=3,col="grey")
abline(h=-2,lty=3,col="grey")
```



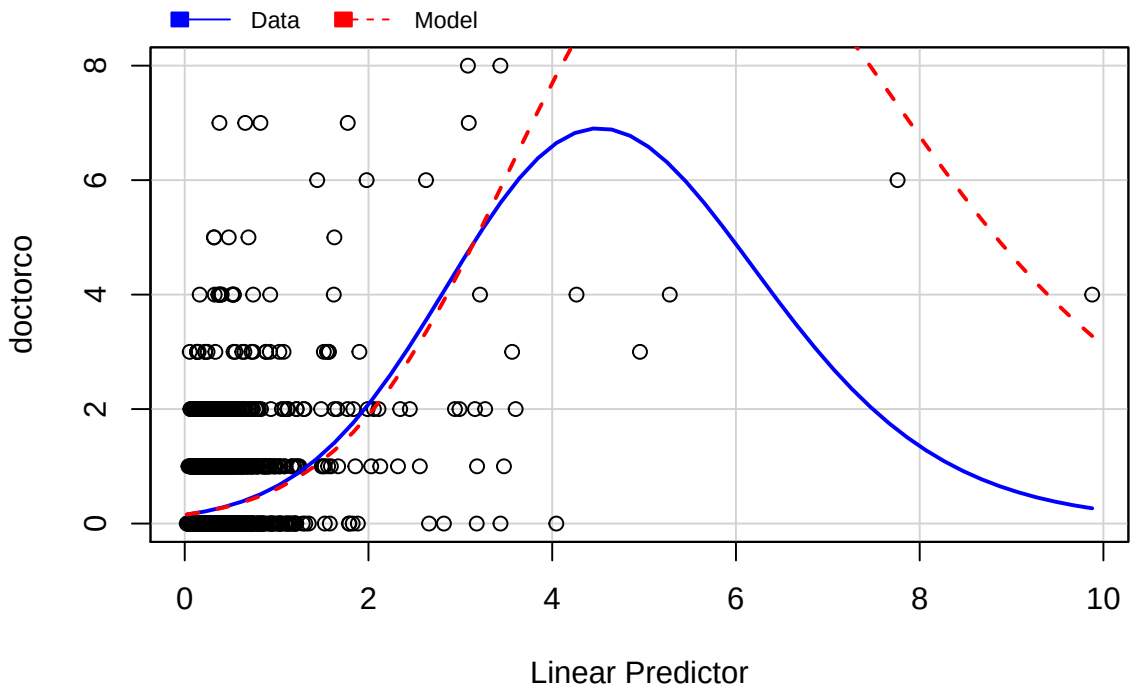
3. What sort of person would be predicted to visit the doctor the most under your selected model?

We split the model for actdays<14 to see what effects are large.

```
moddocqp0 <- glm(doctorco ~ sex +  freepoor*levyplus * freerepa + (age+agesq)*illness * income* hscore*a
                 data=dvisits[dvisits$actdays<14,])
coefplot_my(moddocqp0)
```



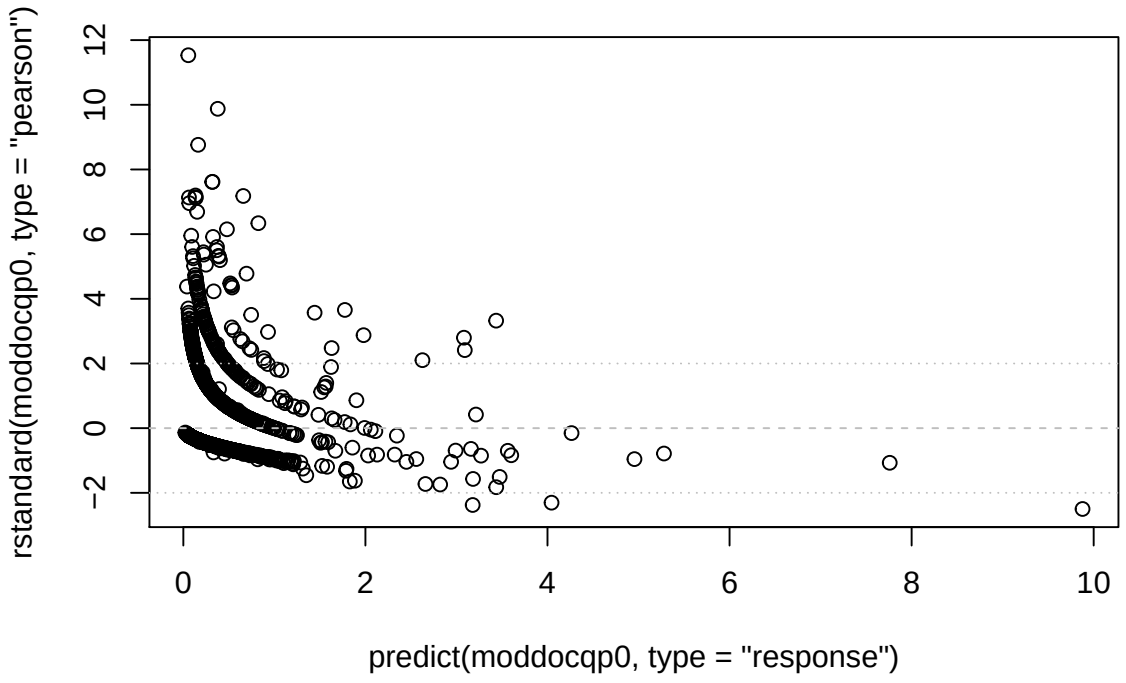
```
marginalModelPlot(moddocqp0)
```




```
AIC(moddocqp0)
```

```
## [1] NA
```

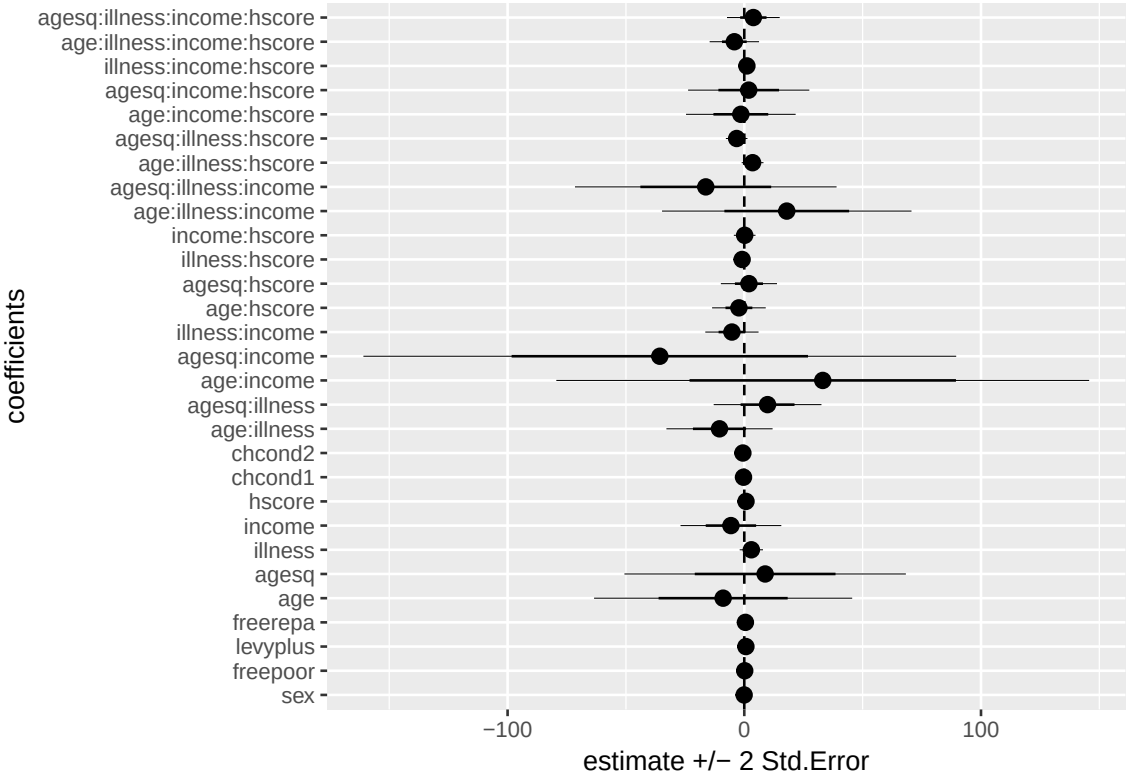
```
plot(predict(moddocqp0, type="response"), rstandard(moddocqp0, type="pearson"));
abline(h=0,lty=2,col="grey")
abline(h=2,lty=3,col="grey")
abline(h=-2,lty=3,col="grey")
```



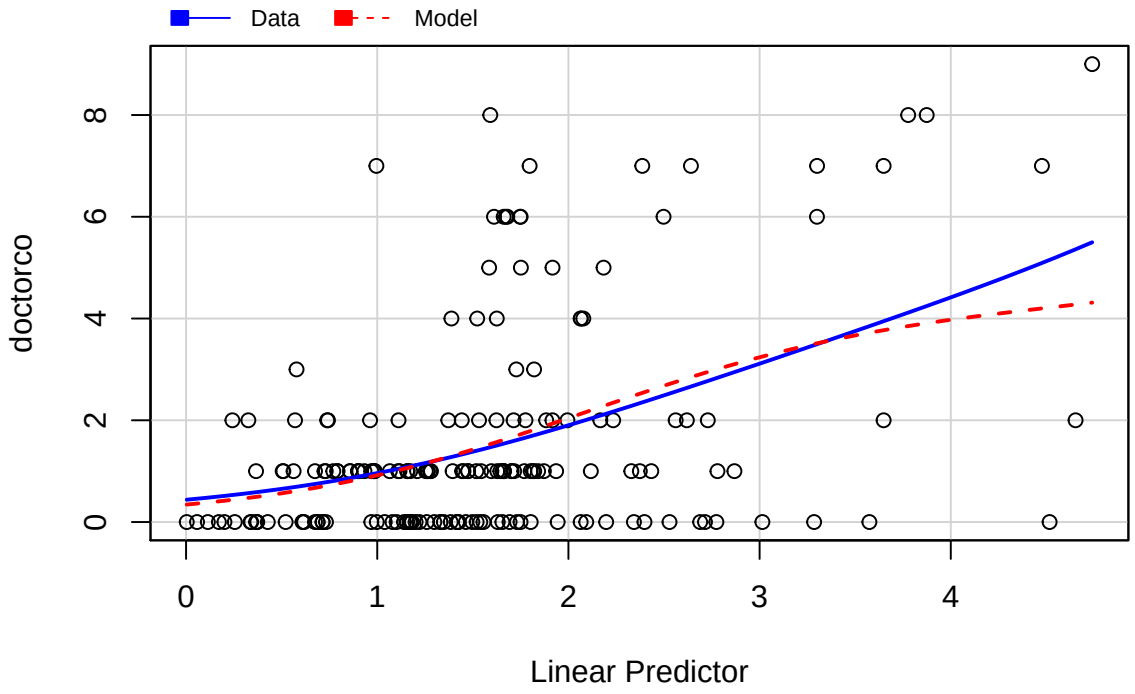
Based on this model older person with more income tend to visit the doctor more. However, this model does not seem to fit the data well.

When you look at only actdays=14 patients the story is inconclusive.

```
moddocqp14 <- glm(doctorco ~ sex +  freepoor*levyplus * freerepa + (age+agesq)*illness * income* hscore
                  data=dvisits[dvisits$actdays>=14,])
coefplot_my(moddocqp14)
```



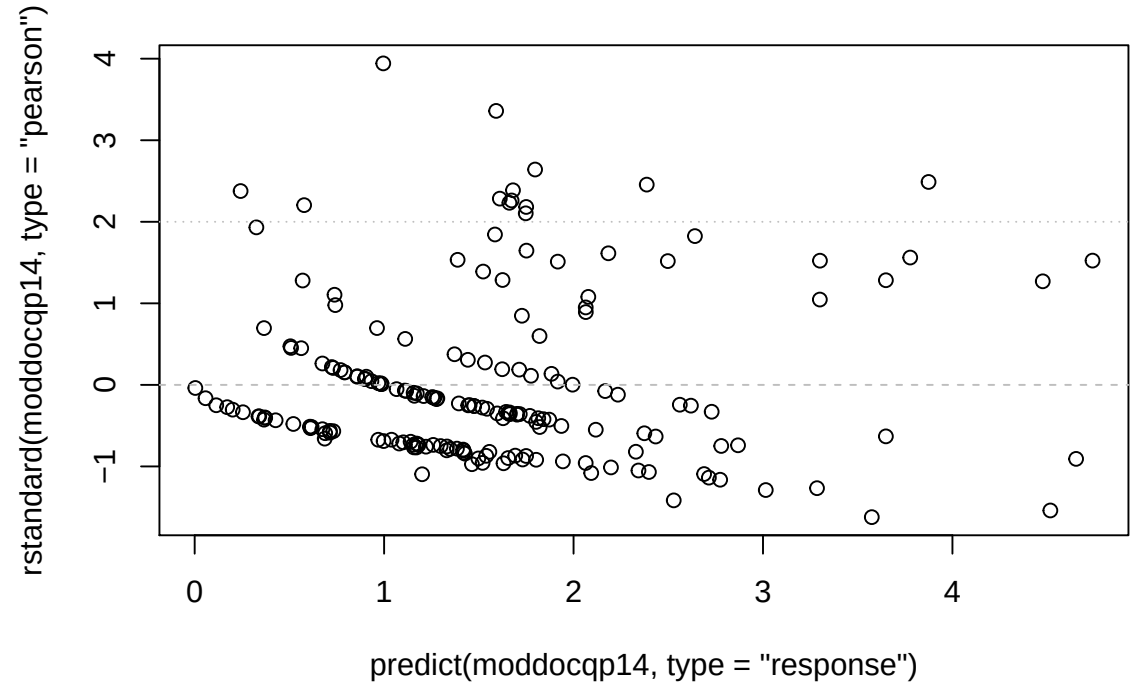

```
marginalModelPlot(moddocqp14)
```




```
AIC(moddocqp14)
```

```
## [1] NA
```

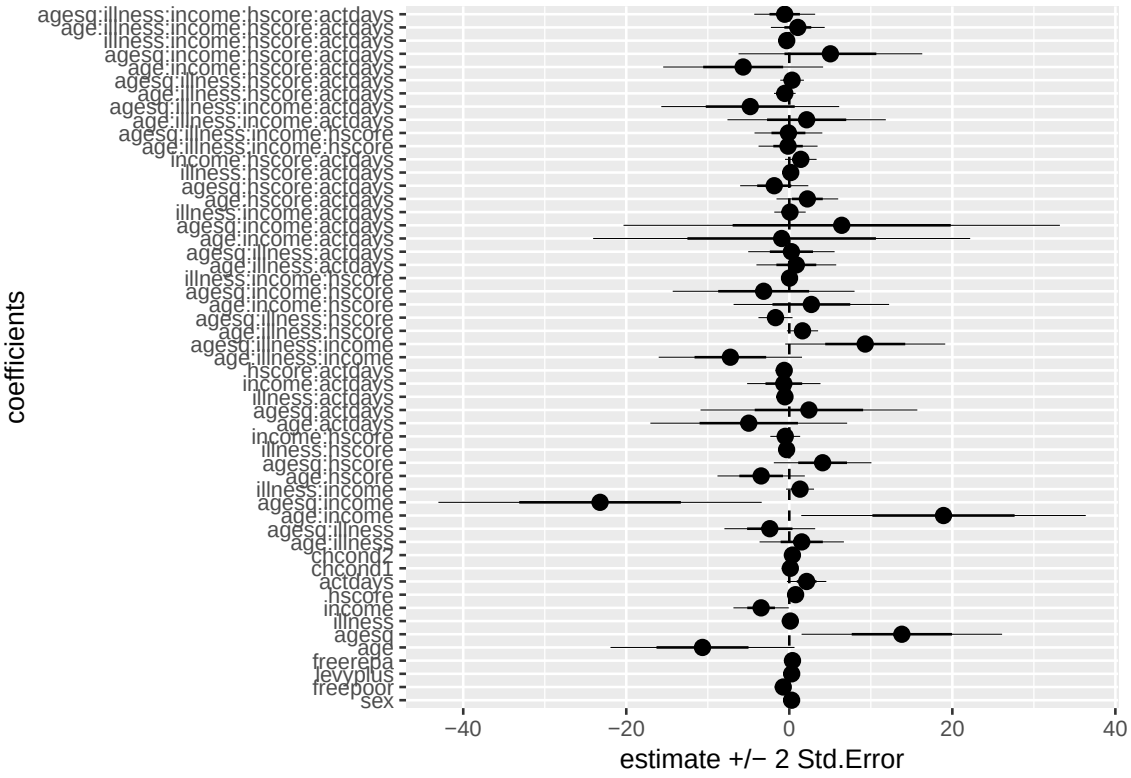
```
plot(predict(moddocqp14, type="response"), rstandard(moddocqp14, type="pearson"));
abline(h=0,lty=2,col="grey")
abline(h=2,lty=3,col="grey")
abline(h=-2,lty=3,col="grey")
```



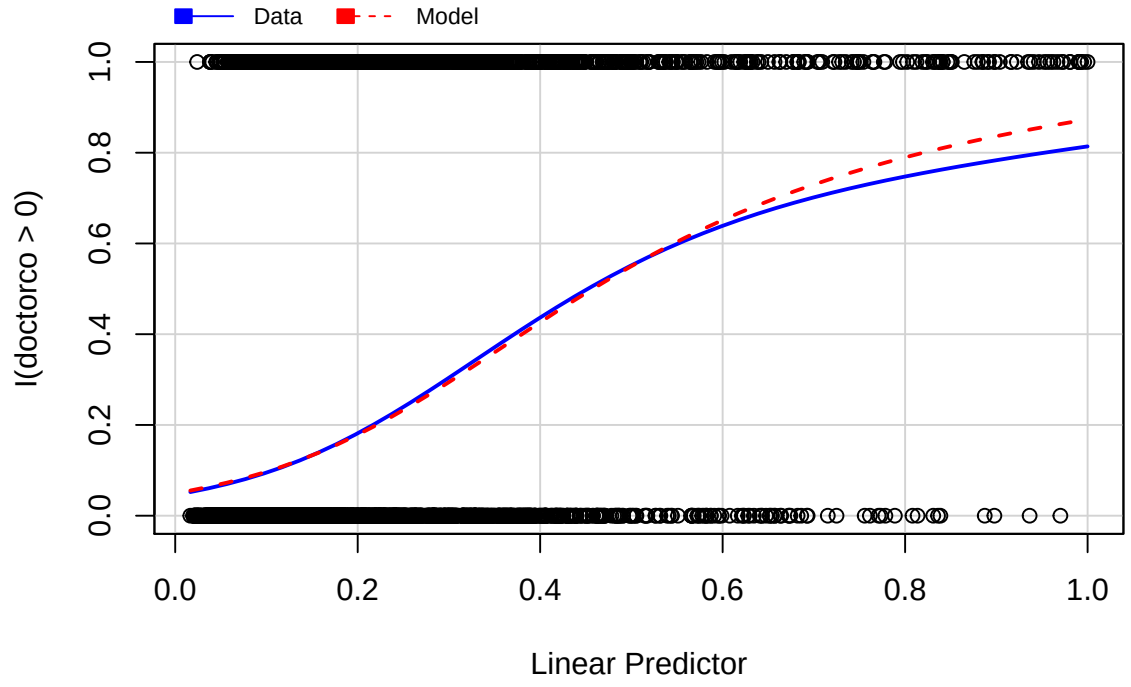
How about we only look at patients that are have no visit and none zero visits separately?

zero visits vs nonzero

```
moddocqp000 <- glm(I(doctorco>0) ~ sex +  freepoor*levyplus * freerepa + (age+agesq)*illness * income* h
                   data=dvisits[dvisits$actdays<14,])
coefplot_my(moddocqp000)
```

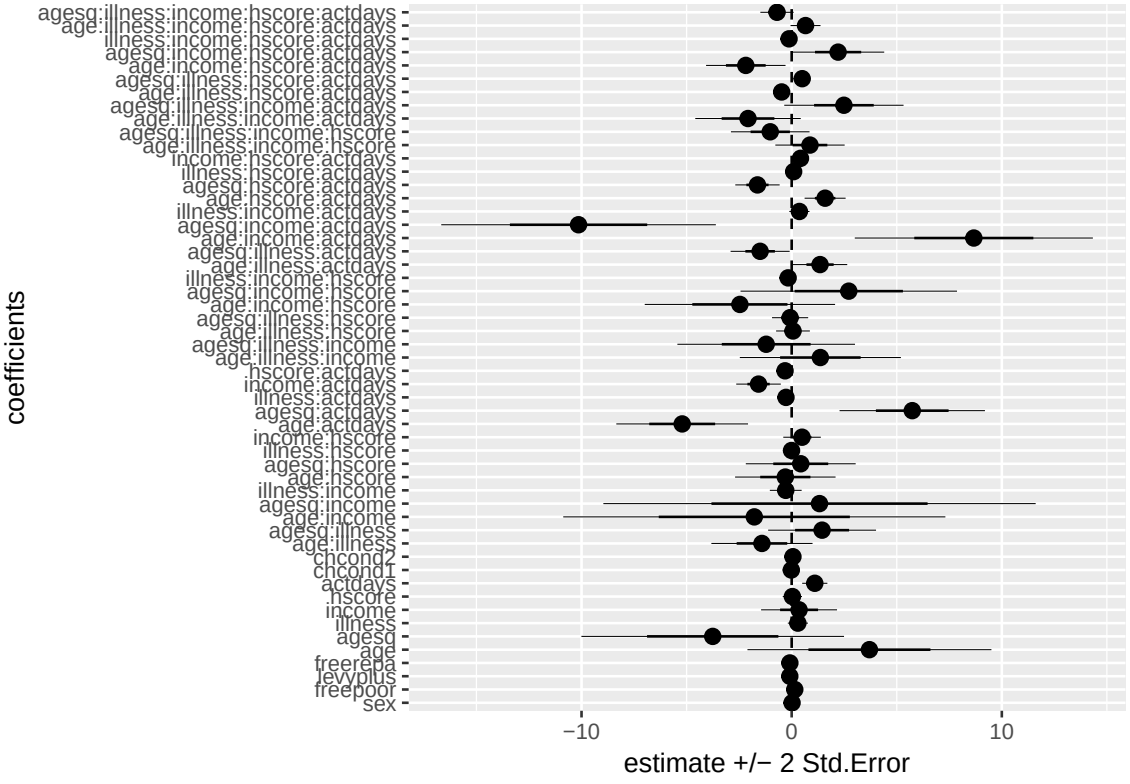



```
marginalModelPlot(moddocqp000)
```

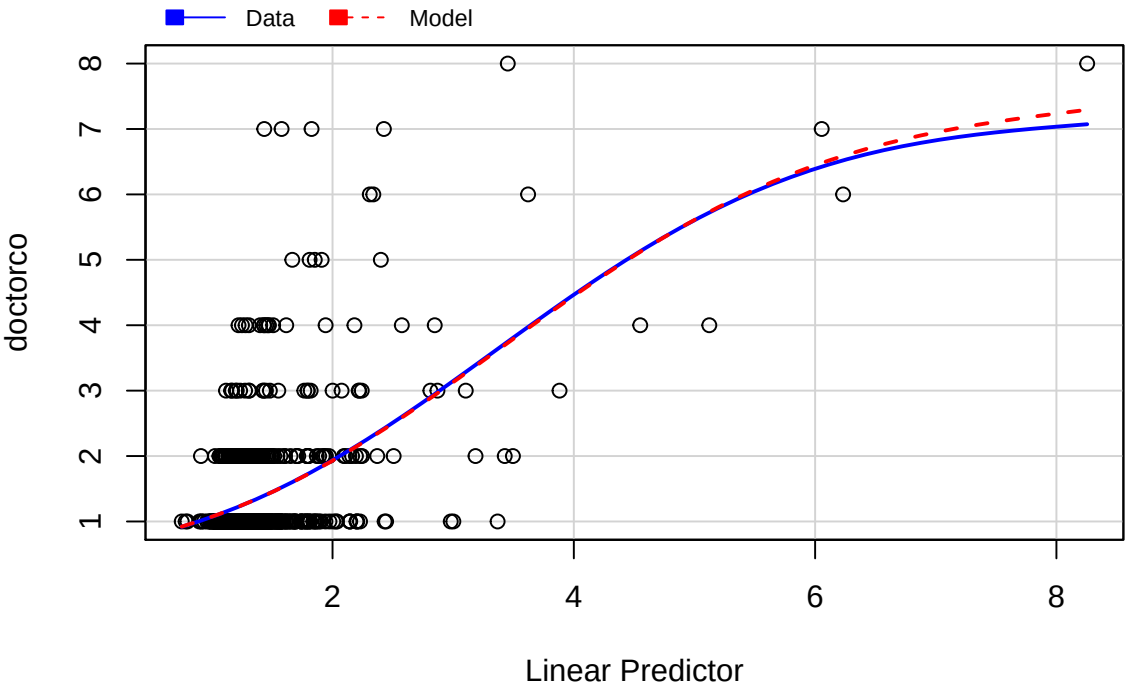


nonezero visits:

```
moddocqp014 <- glm(doctorco ~ sex +  freepoor*levyplus * freerepa + (age+agesq)*illness * income* hscor
                   data=dvisits[dvisits$actdays<14&dvisits$doctorco>0,])
coefplot_my(moddocqp014)
```



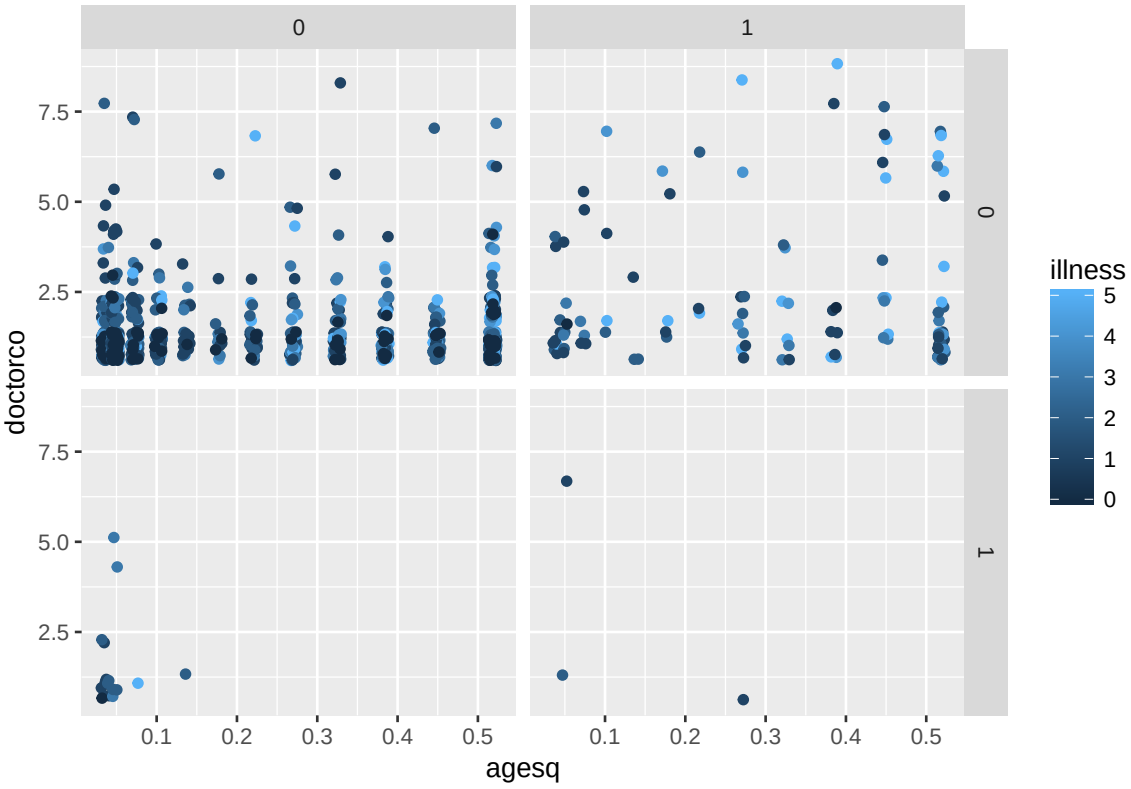

```
marginalModelPlot(moddocqp014)
```



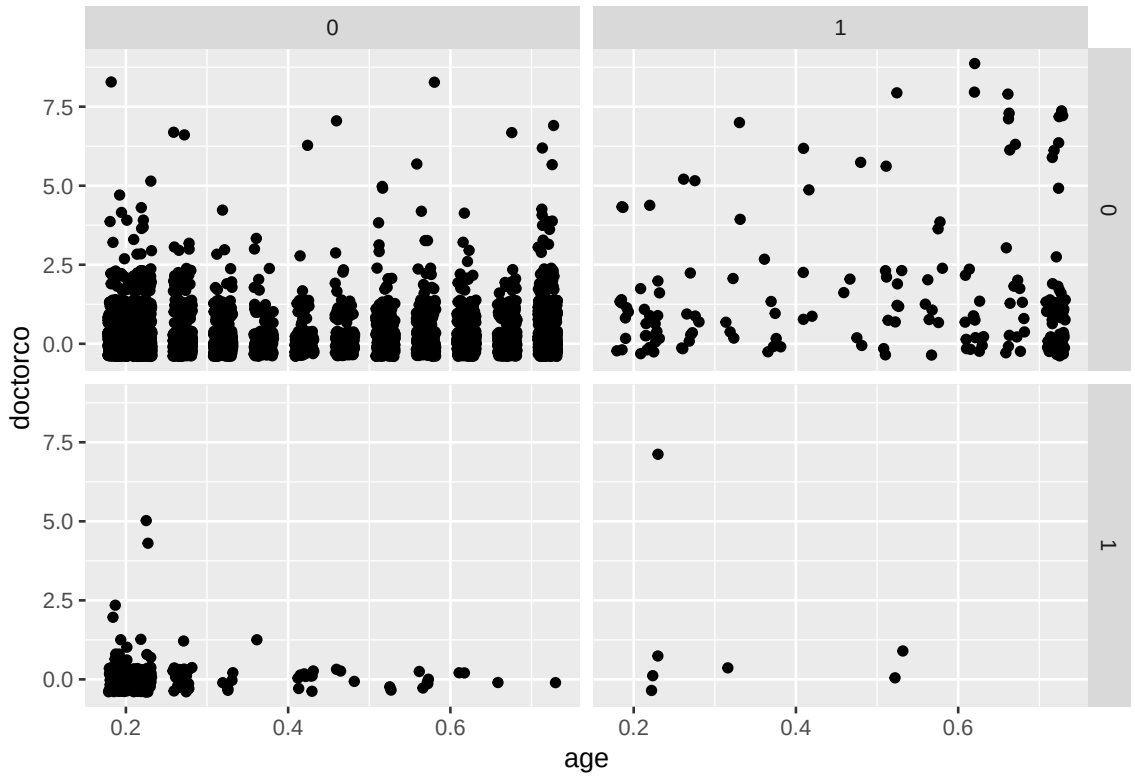
This is essentially a hurldle model.

```
moddocqp0 <- pscl::hurdle(doctorco ~  sex +  freepoor+levyplus + freerepa + (age+agesq)*illness * income
```

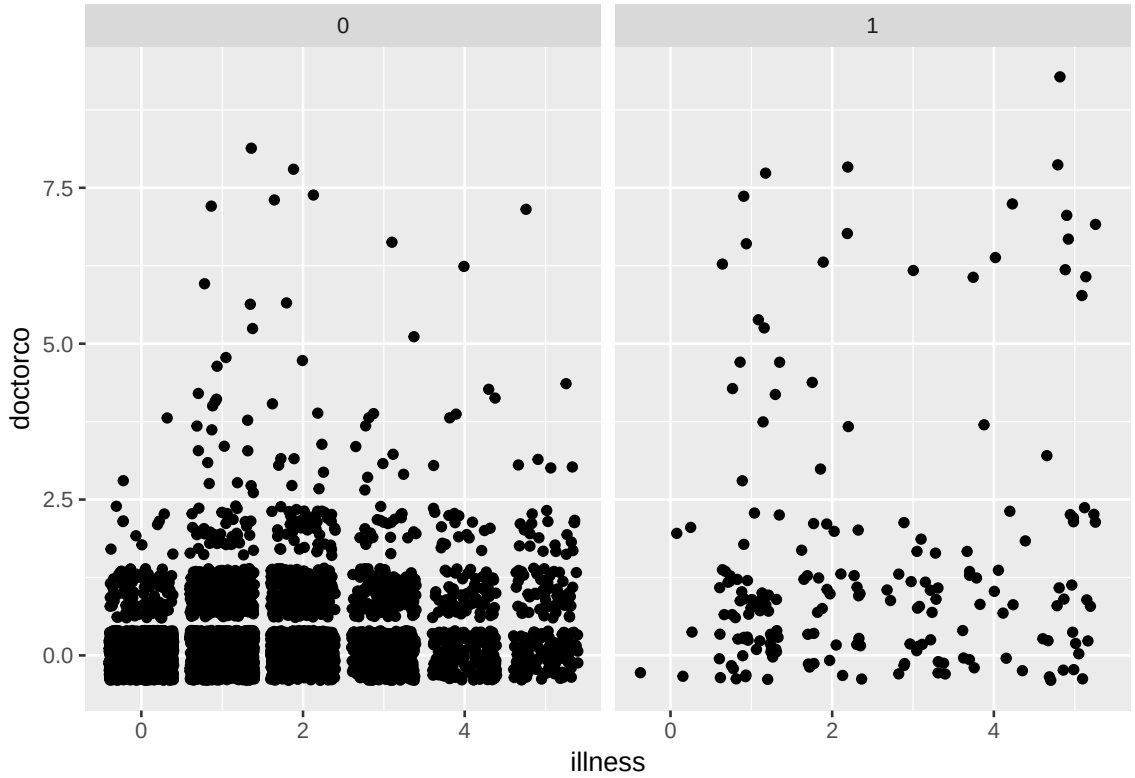
```
dvisitspos<-dvisits[dvisits$doctorco>0,]
ggplot(dvisitspos)+geom_jitter()+
  aes(x=agesq,y=doctorco,color=illness)+facet_grid(freepoor ~actdays_14)
```




```
ggplot(dvisits)+geom_jitter()+
  aes(x=age,y=doctorco)+facet_grid(freepoor ~actdays_14)
```

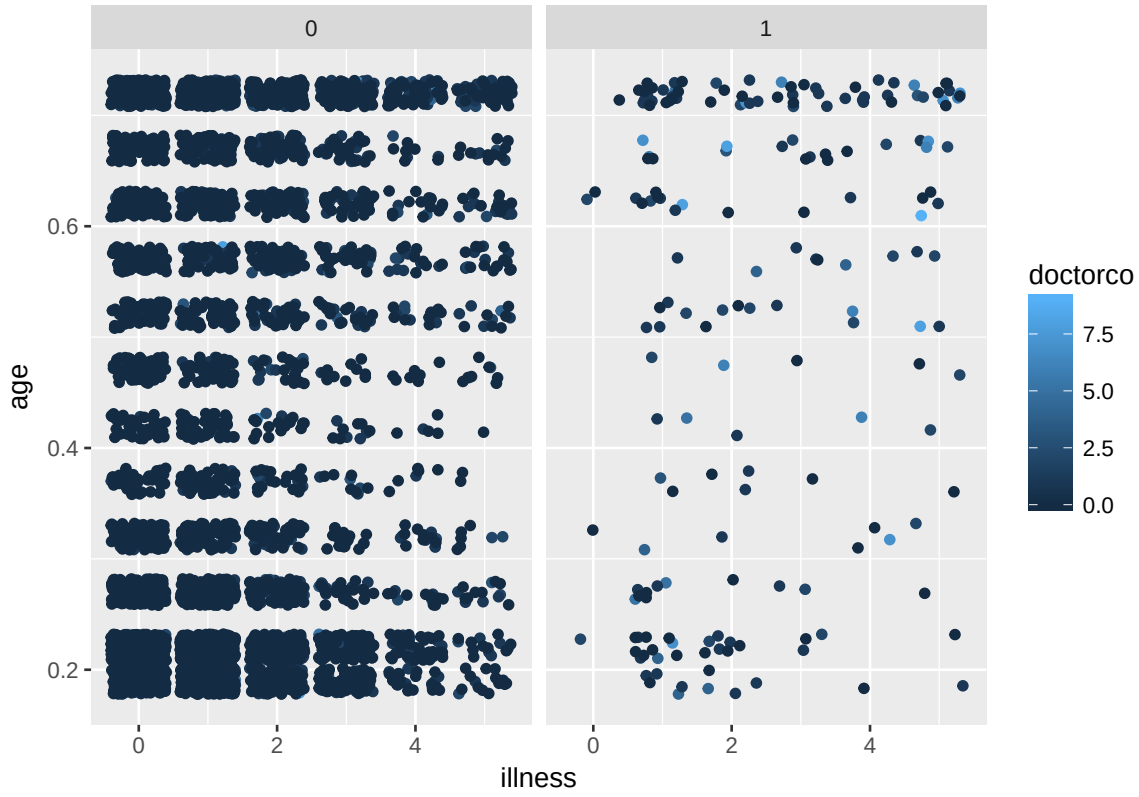



```
ggplot(dvisits)+geom_jitter()+
  aes(x=illness,y=doctorco)+facet_wrap(~actdays_14)
```




```
ggplot(dvisits)+geom_jitter()+
```

```
  aes(x=illness,y=age,color=doctorco)+facet_wrap(~actdays_14)
```



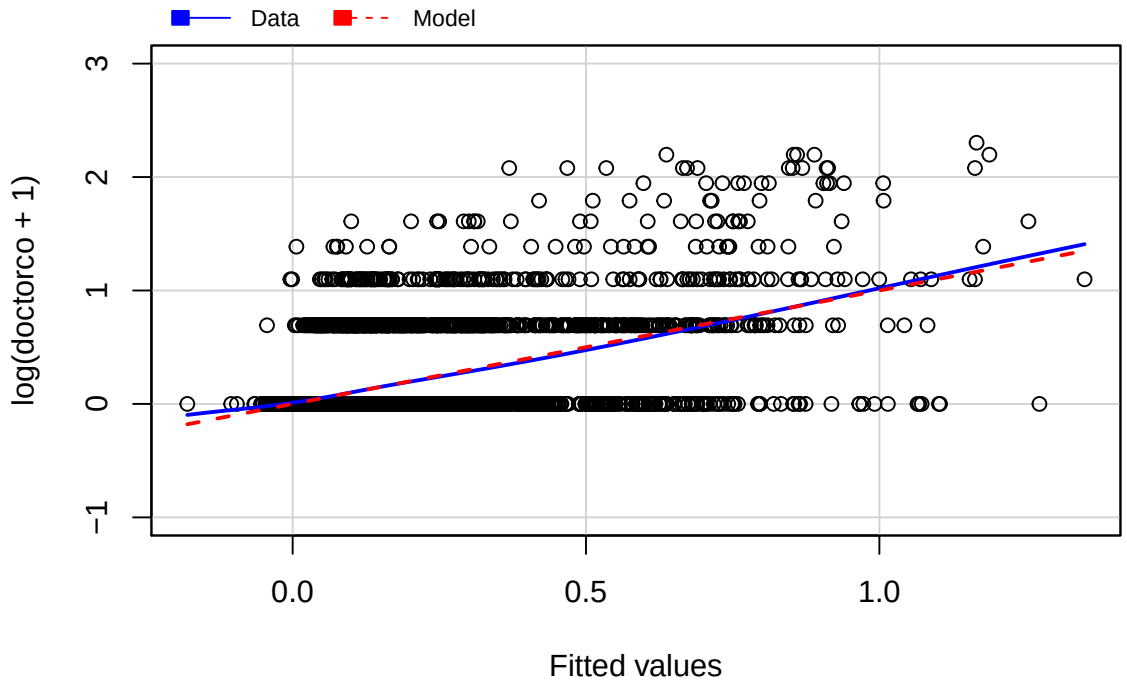
4. For the last person in the dataset, compute the predicted probability distribution for their visits to the doctor, i.e., give the probability they visit 0,1,2, etc. times.

```
round(dpois(c(0,1,2,3,4,5),lambda=tail(predict(moddocp,type="response"),1)),2)
```

```
## [1] 0.84 0.15 0.01 0.00 0.00 0.00
```

5. Fit a comparable (Gaussian) linear model and graphically compare the fits. Describe how they differ.

```
moddoclm <- lm( log(doctorco+1) ~ sex +  levyplus * freepoor
                  * freerepa + (age+agesq)*illness *income*actdays_cat *actdays_14* hscore + chcond1+ch
                    data=dvisits)
marginalModelPlot(moddoclm,ylim=c(-1,3))
```




```
AIC(moddoclm)
```

```
## [1] 3097.7
```

```
plot(predict(moddoclm), rstandard(moddoclm));
abline(h=0,lty=2,col="grey")
abline(h=2,lty=3,col="grey")
abline(h=-2,lty=3,col="grey")
```