Homework 04

Generalized Linear Models

Megha

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".

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

```
rb$fupacts <- round(rb$fupacts)
fit1 <- glm(fupacts ~ women_alone + couples + sex, data = rb, family = poisson)
display(fit1)
```

```
## glm(formula = fupacts ~ women_alone + couples + sex, family = poisson,
##     data = rb)
##             coef.est coef.se
## (Intercept)  3.12     0.02
## women_alone -0.57     0.03
## couples     -0.32     0.03
## sexman      -0.06     0.02
## ---
##   n = 434, k = 4
##   residual deviance = 12918.8, null deviance = 13298.6 (difference = 379.8)
```

```
#checking the fit of the model with the null hypothesis that the model fits well
1 - pchisq(13064.2, 434)
```

```
## [1] 0
```

```
#Checking for Overdispersion
tapply(rb$fupacts, rb$women_alone, function(x)c(mean=mean(x),variance=var(x)))
```

```
## $`0`
##     mean variance
##  18.5625 802.9229
##
## $`1`
##      mean  variance
##  12.39726 533.30316
```

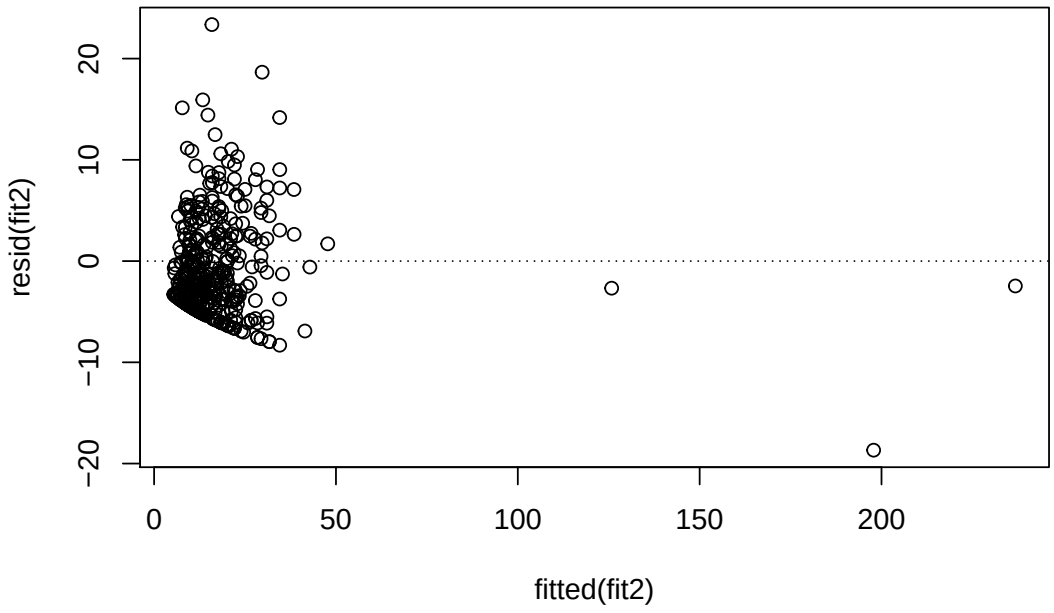
The model does not fit well. There is overdispersion by a huge factor.

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?

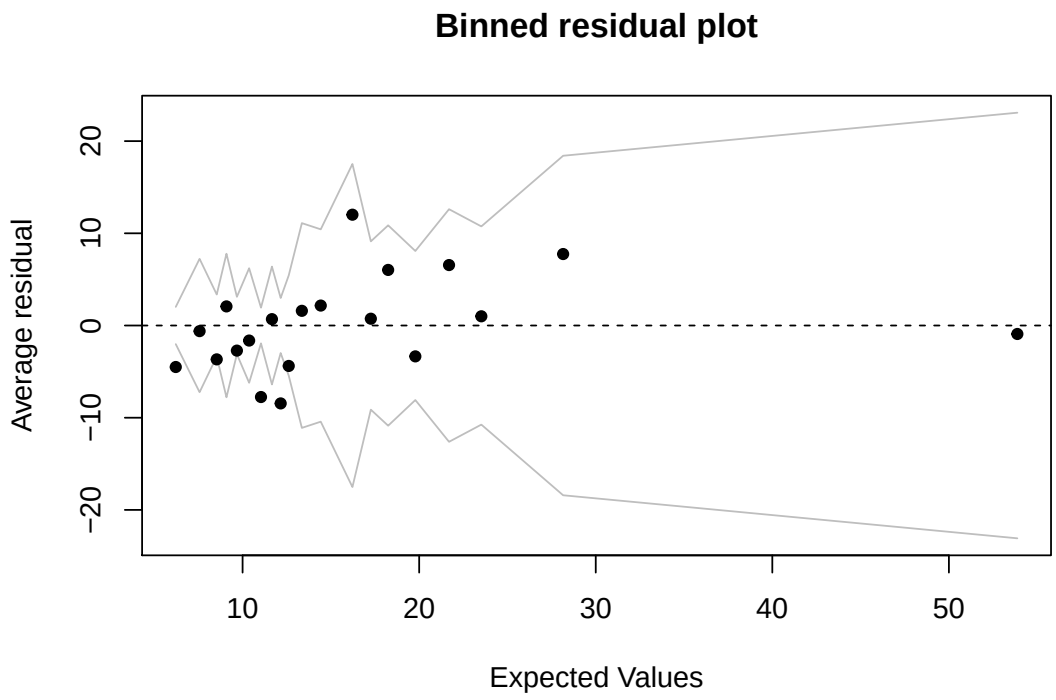
```
fit2 <- glm(fupacts ~ sex + couples + women_alone + bs_hiv + bupacts, data = rb, family = poisson)
display(fit2)
```

```
## glm(formula = fupacts ~ sex + couples + women_alone + bs_hiv +
##     bupacts, family = poisson, data = rb)
##                coef.est coef.se
## (Intercept)     2.90     0.02
## sexman         -0.11     0.02
## couples        -0.41     0.03
## women_alone    -0.66     0.03
## bs_hivpositive -0.44     0.04
## bupacts         0.01     0.00
## ---
##   n = 434, k = 6
##   residual deviance = 10200.4, null deviance = 13298.6 (difference = 3098.2)
```

```
plot(fitted(fit2), resid(fit2))
abline(0,0,lty = 3)
```




```
binnedplot(fitted(fit2), resid(fit2, type = "response"))
```




```
dispersiontest(fit2, trafo = 1)
```

```
##
##  Overdispersion test
##
## data:  fit2
## z = 5.5689, p-value = 1.282e-08
## alternative hypothesis: true alpha is greater than 0
## sample estimates:
##    alpha
## 28.65146
```

The overdispersion factor is 28.65 which is very high. There is clear evidence of overdispersion.

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

```
fit3 <- glm(fupacts ~ sex + couples + women_alone + bs_hiv + bupacts, data = rb, family = quasipoisson)
display(fit3)
```

```
## glm(formula = fupacts ~ sex + couples + women_alone + bs_hiv +
##     bupacts, family = quasipoisson, data = rb)
##                coef.est coef.se
## (Intercept)     2.90     0.13
## sexman         -0.11     0.13
## couples        -0.41     0.15
## women_alone    -0.66     0.17
## bs_hivpositive -0.44     0.19
## bupacts         0.01     0.00
## ---
##   n = 434, k = 6
```

```
##   residual deviance = 10200.4, null deviance = 13298.6 (difference = 3098.2)
##   overdispersion parameter = 30.0
```

```
pfupacts <- predict(fit3, type = "response")
z <- (rb$fupacts - pfupacts)/sqrt(pfupacts)
n <- fit3$df.null + 1
k <- fit3$df.null + 1 - fit3$df.residual
cat("overdispersion ratio: ", sum(z^2)/(n - k), "\n")
```

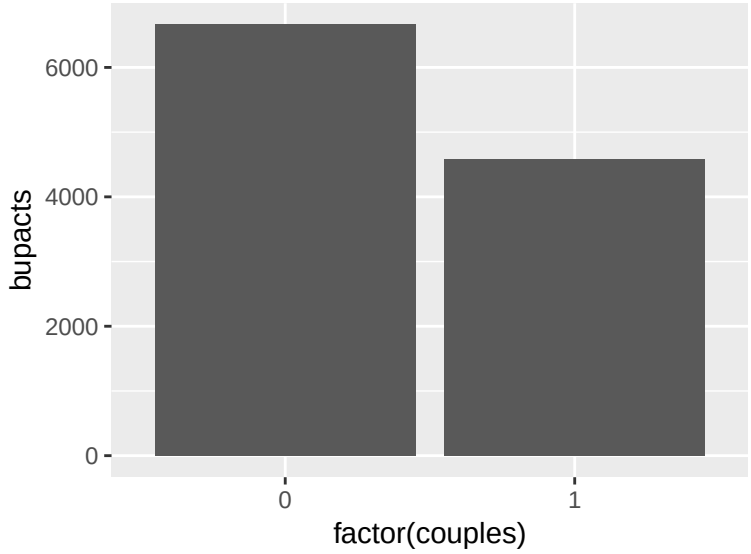
```
## overdispersion ratio:  30.00404
```

```
cat("p-value of overdispersion: ", pchisq(sum(z^2), n-k), "\n")
```

```
## p-value of overdispersion:  1
```

```
ggplot(rb, aes(x=factor(couples), y=bupacts)) + geom_histogram(stat = "identity")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



The treatment overall had a positive impact on reducing the number of unprotected sex acts. For the treatment where only the women took part, i.e., coefficient of women_alone = e^(-0.66) = 0.5168, implies that there was around a 48% redcution in unprotected sex acts post the treatment. Surprisingly, when couples took part in the treatment, there was only around a 33% reduction. This can also be seen from the histogram. But, overall, there is a decrease in unprotected sex acts, implying that the intervention had a positive effect.

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?

Yes. It may be that the women or men who received treatment alone overlapped with the couples that received the intervention. In this case, the predictors will not be independent of each other.

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)

```r
#Taking the wells data from Chapter 5
wells <- read.table("http://www.stat.columbia.edu/~gelman/arm/examples/arsenic/wells.dat", header=TRUE)
wells_dt <- data.table(wells)

fit4 <- glm(switch ~ log(arsenic) + dist + assoc + educ, data = wells_dt, family = binomial(link = "logi
display(fit4)
```

```
## glm(formula = switch ~ log(arsenic) + dist + assoc + educ, family = binomial(link = "logit"),
##     data = wells_dt)
##              coef.est coef.se
## (Intercept)   0.37     0.08
## log(arsenic)  0.89     0.07
## dist         -0.01     0.00
## assoc        -0.12     0.08
## educ          0.04     0.01
## ---
##   n = 3020, k = 5
##   residual deviance = 3875.6, null deviance = 4118.1 (difference = 242.5)
```

```r
fit5 <- glm(switch ~ log(arsenic) + dist + assoc + educ, data = wells_dt, family = binomial(link = "prob
display(fit5)
```

```
## glm(formula = switch ~ log(arsenic) + dist + assoc + educ, family = binomial(link = "probit"),
##     data = wells_dt)
##              coef.est coef.se
## (Intercept)   0.23     0.05
## log(arsenic)  0.54     0.04
## dist         -0.01     0.00
## assoc        -0.08     0.05
## educ          0.03     0.01
## ---
##   n = 3020, k = 5
##   residual deviance = 3875.5, null deviance = 4118.1 (difference = 242.5)
```

The logit and the probit models have been fit to the wells data from chapter 5. Both the models yield similar results, i.e., the results given by the probit model are equal to the results given by the logit model scaled by a factor of 1.6. For example, the intercept for the probit model, 0.23, is equal to the intercept of the logit model 0.37 scaled by 1.6, 0.37/1.6 = 0.23. And the same for the coefficient of log arsenic, 0.89/1.6 = 0.55, and other coefficients.

Comparing logit and probit:

construct a dataset where the logit and probit mod- els give different estimates.

Tobit model for mixed discrete/continuous data:

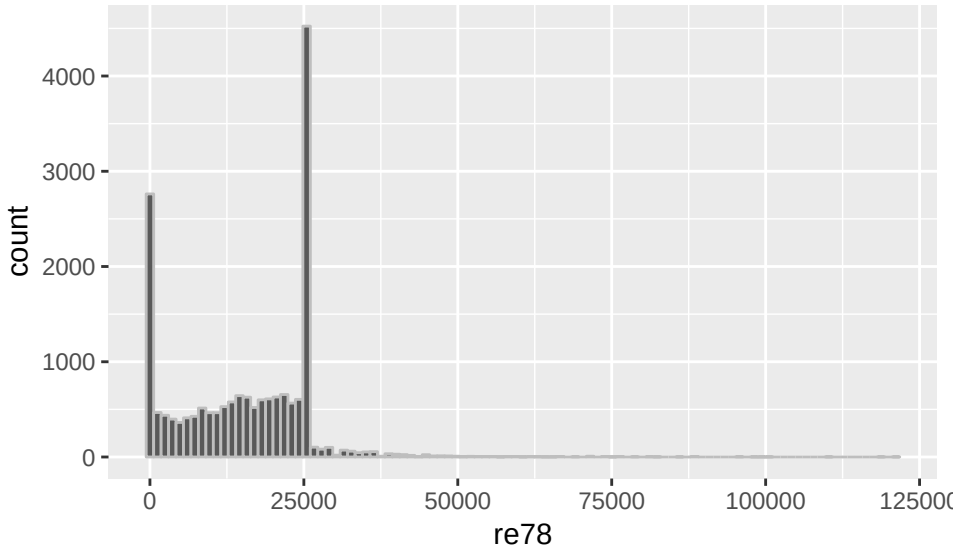
experimental data from the National Supported Work example are available in the folder `lalonde`. Use the treatment indicator and pre-treatment variables to predict post-treatment (1978) earnings using a tobit model. Interpret the model coefficients.

- 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)

```
lalonde<-read.dta("http://www.stat.columbia.edu/~gelman/arm/examples/lalonde/NSW.dw.obs.dta")

#checking the distribution of the outcome re78
ggplot(lalonde, aes(x = re78)) + geom_histogram(binwidth = (max(lalonde$re78) - min(lalonde$re78))/100,
```




```
#fitting tobit regression
fit6 <- tobit(re78 ~ age + educ + black + married + re74 + re75 + hisp + nodegree + sample + treat + edu
fit6
```

```
##
## Call:
## tobit(formula = re78 ~ age + educ + black + married + re74 +
##     re75 + hisp + nodegree + sample + treat + educ_cat4, data = lalonde)
##
## Coefficients:
## (Intercept)          age         educ        black      married
##    801.2798    -156.4444     103.0584    -756.7916    -284.1282
##        re74         re75         hisp     nodegree       sample
##      0.3341       0.5562      19.7268    1048.6776    2138.1988
##       treat    educ_cat4
##      0.0000     360.4360
##
## Scale: 8437
```

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.

```
congress<-read.csv("congress(1).csv",header=TRUE)

#filtering data for the year 1988
cong <- filter(congress, year == "1988")

#omitting the rows that have NAs
cong <- na.omit(cong)
```

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.

```
fit_cong <- lm(Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote, data = cong)
summary(fit_cong)
```

```
##
## Call:
## lm(formula = Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote,
##     data = cong)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50041 -0.03824  0.00203  0.05492  0.26074
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    8.736e-01  1.601e-02  54.572  < 2e-16 ***
## x1            -9.213e-06  2.016e-04  -0.046    0.964
## x2            -1.160e-04  3.117e-04  -0.372    0.710
## incumbent      5.610e-02  8.253e-03   6.798 3.77e-11 ***
## contestedTRUE -9.812e-03  1.409e-02  -0.696    0.487
## Rep_vote      -3.488e-06  1.543e-07 -22.610  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08795 on 410 degrees of freedom
## Multiple R-squared:  0.8851, Adjusted R-squared:  0.8837
## F-statistic: 631.8 on 5 and 410 DF,  p-value: < 2.2e-16
```

The R-squared for the linear regression model shows that the model explains around 88% of the variation of the dependent variable. But the coefficients are very small in value to explain sufficient variation in the outcome. And, three out of the six coefficients are not statistically significant.

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.

```
fit7 <- tlm(Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote, data = cong)
summary(fit7)
```

```
## Location model :
##
## Call:
## tlm(lform = Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote,
##     data = cong)
##
## Residuals:
##       Min         1Q     Median         3Q        Max
## -0.603886  -0.034375   0.004475   0.027271   0.211074
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    9.347e-01  9.431e-03  99.104   <2e-16 ***
## x1             3.308e-05  1.188e-04   0.279    0.781
## x2            -5.250e-05  1.836e-04  -0.286    0.775
## incumbent      5.886e-02  4.862e-03  12.105   <2e-16 ***
## contestedTRUE -1.214e-01  8.300e-03 -14.625   <2e-16 ***
## Rep_vote      -3.018e-06  9.090e-08 -33.200   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Scale parameter(s) as estimated below)
##
##
## Scale Model :
##
## Call:
## tlm(lform = Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote,
##     data = cong)
##
## Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0000  -1.8724  -0.7263   1.2515   5.8839
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.32557    0.09806  -64.51   <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 : 22 in 0.01
## Heteroscedastic t Likelihood : 546.7111
```

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$Dem_pct <- ifelse(cong$Dem_pct >= 0.5, 1, 0)

fit8 <- glm(Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote, data = cong, family = binomial(link =
summary(fit8)
```

```
##
## Call:
## glm(formula = Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote,
##     family = binomial(link = "logit"), data = cong)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.35523  -0.03326   0.01201   0.04748   3.11452
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    8.605e+00  8.849e+00   0.972    0.331
## x1            -2.423e-02  1.972e-02  -1.228    0.219
## x2            -2.569e-02  1.694e-02  -1.517    0.129
## incumbent      1.872e+00  4.529e-01   4.133 3.58e-05 ***
## contestedTRUE  4.952e+00  8.778e+00   0.564    0.573
## Rep_vote      -1.192e-04  2.828e-05  -4.214 2.51e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 561.218  on 415  degrees of freedom
## Residual deviance:  54.275  on 410  degrees of freedom
## AIC: 66.275
##
## Number of Fisher Scoring iterations: 10
```

```
fit9 <- glm(Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote, data = cong, family = binomial(link =
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(fit9)
```

```
##
## Call:
## glm(formula = Dem_pct ~ x1 + x2 + incumbent + contested + Rep_vote,
##     family = binomial(link = "probit"), data = cong)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.31516  -0.01330   0.00220   0.02472   2.96323
##
## Coefficients:
```

```
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    3.764e+00  5.046e+00   0.746    0.456
## x1            -1.034e-02  9.904e-03  -1.044    0.296
## x2            -1.210e-02  9.037e-03  -1.339    0.180
## incumbent      1.112e+00  2.351e-01   4.730 2.24e-06 ***
## contestedTRUE  2.553e+00  5.041e+00   0.507    0.612
## Rep_vote      -5.572e-05  1.332e-05  -4.184 2.86e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 561.218  on 415  degrees of freedom
## Residual deviance:  54.116  on 410  degrees of freedom
## AIC: 66.116
##
## Number of Fisher Scoring iterations: 11
```

2. Fit a robit regression and assess model fit.
3. Which model do you prefer?

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)

fit10 <- glm(colonies ~ dose, data = salmonella, family = poisson)
display(fit10)
```

```
## glm(formula = colonies ~ dose, family = poisson, data = salmonella)
##             coef.est coef.se
## (Intercept) 3.32     0.05
## dose        0.00     0.00
## ---
##   n = 18, k = 2
##   residual deviance = 75.8, null deviance = 78.4 (difference = 2.6)
```

```
tapply(salmonella$colonies, salmonella$dose, function(x)c(mean=mean(x),variance=var(x)))
```

```
## $`0`
##     mean variance
## 21.66667 49.33333
##
## $`10`
##      mean  variance
## 18.333333  6.333333
##
## $`33`
##     mean variance
##       25       73
```

```
##
## $`100`
##      mean  variance
##  42.66667 274.33333
##
## $`333`
##     mean variance
## 37.33333 16.33333
##
## $`1000`
##      mean  variance
##  29.66667 126.33333
```

```
pcolonies <- predict(fit10, type = "response")
z_p <- (salmonella$colonies - pcolonies)/sqrt(pcolonies)
n_p <- fit10$df.null + 1
k_p <- fit10$df.null + 1 - fit10$df.residual
cat("overdispersion ratio: ", sum(z_p^2)/(n_p - k_p), "\n")
```

```
## overdispersion ratio:  5.087258
```

```
cat("p-value of overdispersion: ", pchisq(sum(z_p^2), n_p - k_p), "\n")
```

```
## p-value of overdispersion:  1
```

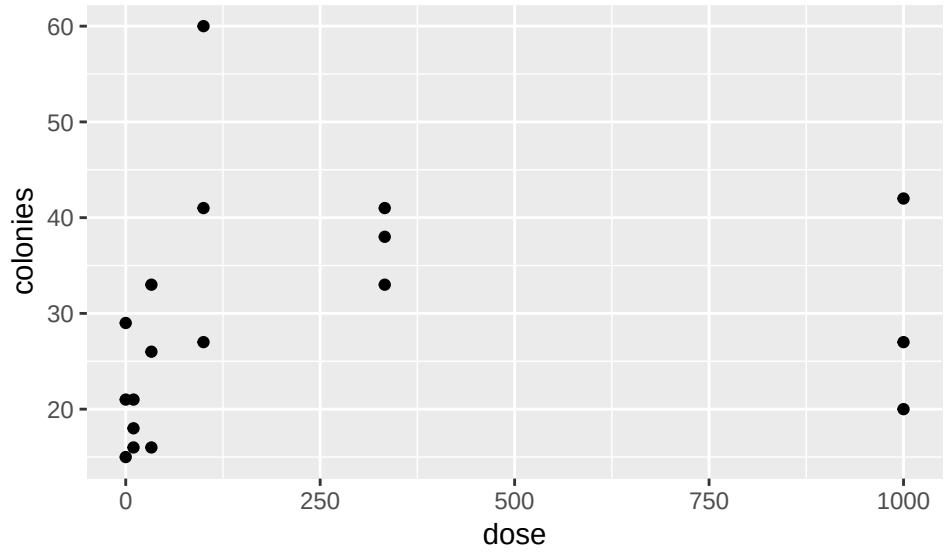
```
fit11 <- glm(colonies ~ dose, data = salmonella, family = quasipoisson)
display(fit11)
```

```
## glm(formula = colonies ~ dose, family = quasipoisson, data = salmonella)
##             coef.est coef.se
## (Intercept) 3.32     0.12
## dose        0.00     0.00
## ---
##   n = 18, k = 2
##   residual deviance = 75.8, null deviance = 78.4 (difference = 2.6)
##   overdispersion parameter = 5.1
```

There is an overdispersion factor of 5.08 indicating that the model has overdispersion. We can fit a quasipoisson model to account for the overdispersion.

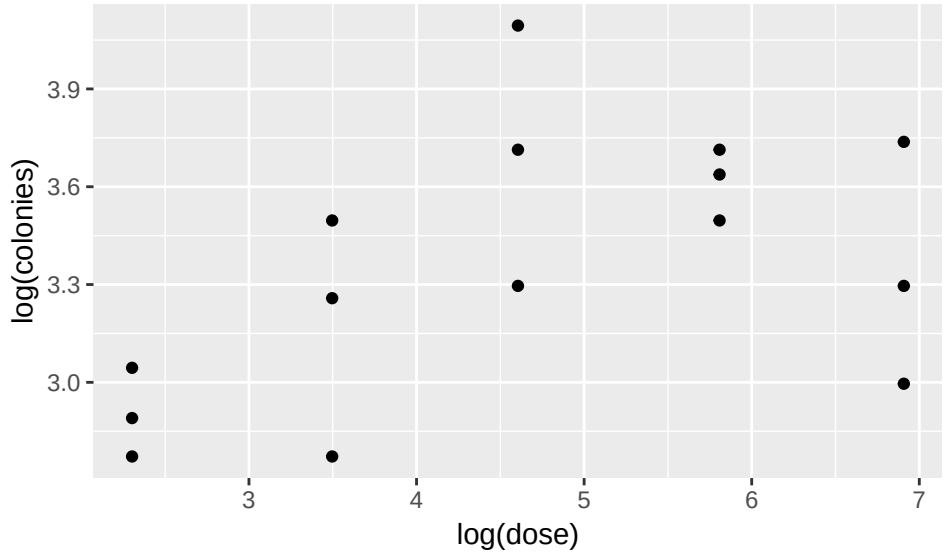
When you plot the data you see that the number of colonies as a function of dose is not monotonic especially around the dose of 1000.

```
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.

```
lsalmonella <- salmonella[salmonella$dose != 0,]
ggplot(lsalmonella) + geom_point(aes(x = log(dose), y = log(colonies)))
```



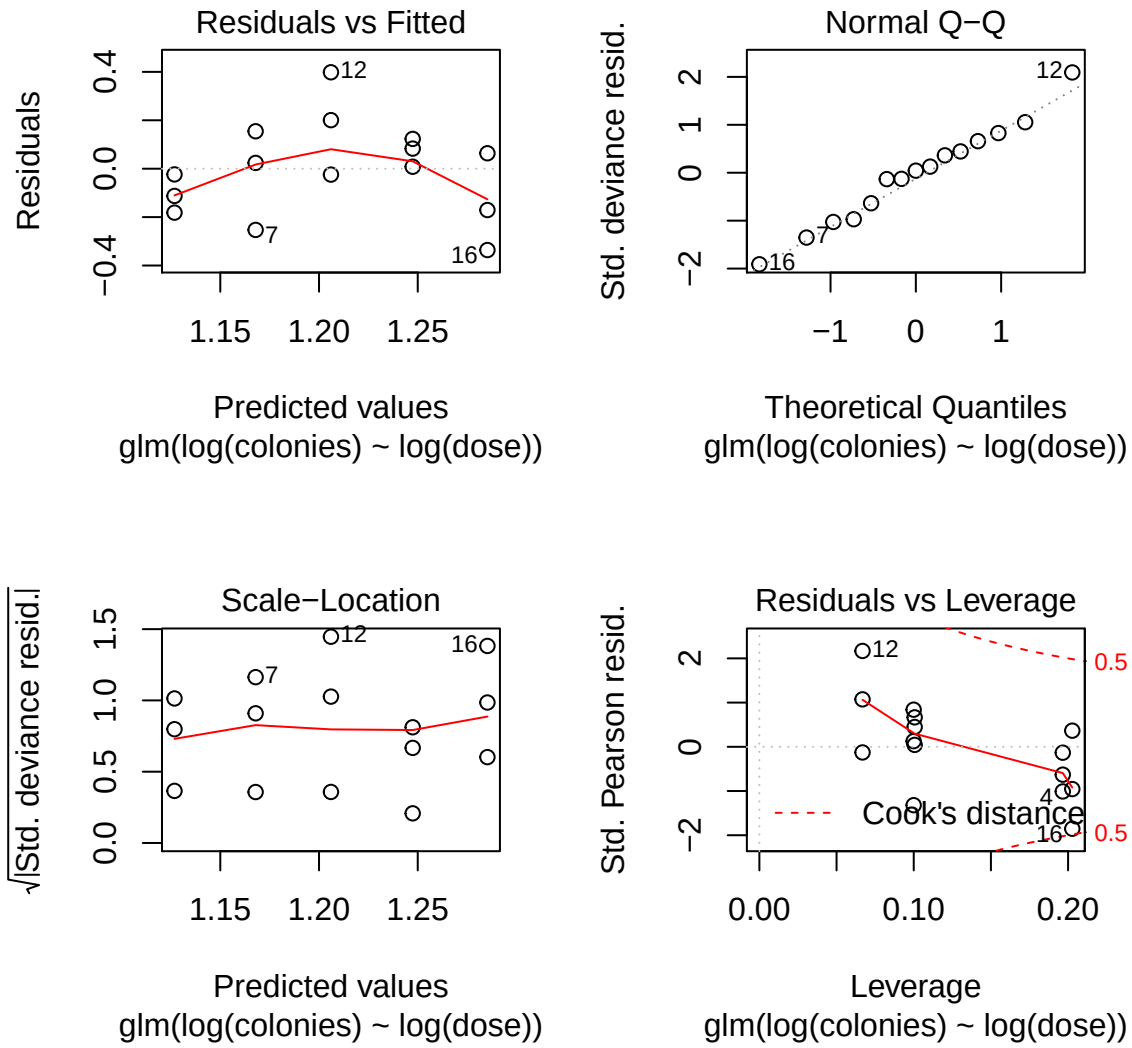
This shows that the trend is not monotonic. Hence when you fit the model and look at the residual you will see a trend.

```
fit12 <- glm(log(colonies) ~ log(dose), data = lsalmonella, family = quasipoisson)
display(fit12)
```

```
## glm(formula = log(colonies) ~ log(dose), family = quasipoisson,
##     data = lsalmonella)
##             coef.est coef.se
## (Intercept) 1.05     0.09
## log(dose)   0.03     0.02
## ---
```

```
##    n = 15, k = 2
##    residual deviance = 0.5, null deviance = 0.7 (difference = 0.2)
##    overdispersion parameter = 0.0
```

```
#plotting the residuals
plot(fit12)
```

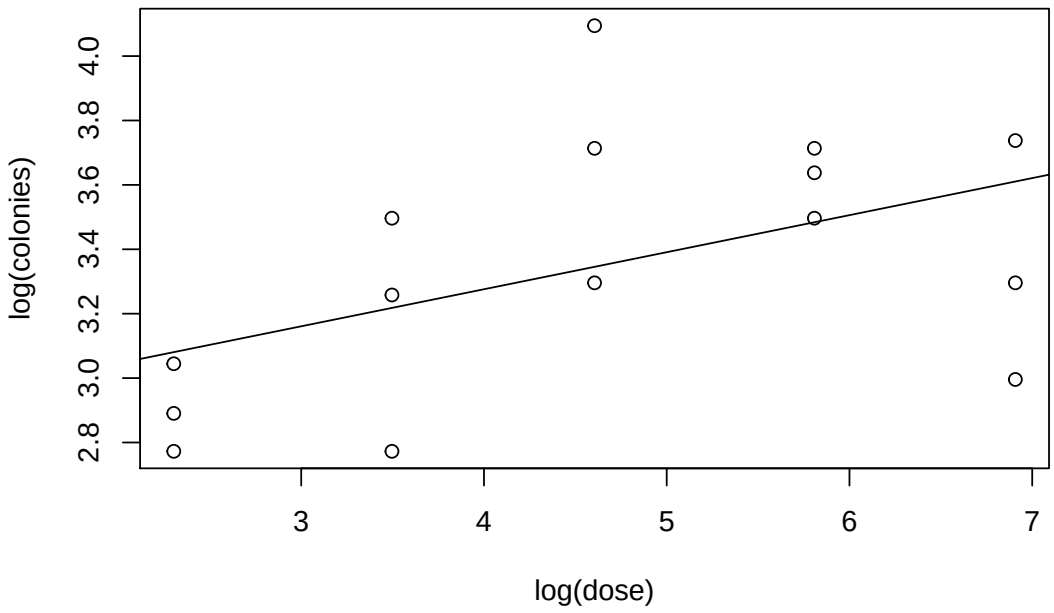


The residuals show a non-linear trend.

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

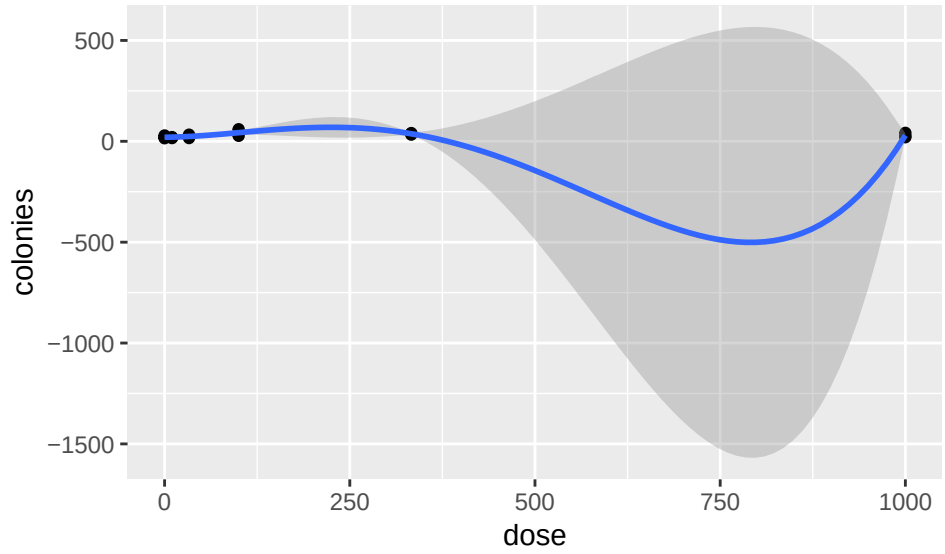
```
#plotting log colonies vs log dose
plot(x = log(lsalmonella$dose), y = log(lsalmonella$colonies), xlab = "log(dose)", ylab = "log(colonies)

#the fitted line does not pass through even one point in the data
abline(lm(log(lsalmonella$colonies) ~ log(lsalmonella$dose)))
```



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 beny line with 4th order polynomial.

```
library(ggplot2)
ggplot(salmonella, aes(x = dose, y = colonies))+
  geom_point()+
  geom_smooth(method = "glm", formula = y ~ poly(x, 4, raw = TRUE))
```



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.

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

```
fit11 <- glm(colonies ~ dose, data = salmonella, family = quasipoisson)
display(fit11)
```

```
## glm(formula = colonies ~ dose, family = quasipoisson, data = salmonella)
##             coef.est coef.se
## (Intercept) 3.32     0.12
## dose        0.00     0.00
## ---
##   n = 18, k = 2
##   residual deviance = 75.8, null deviance = 78.4 (difference = 2.6)
##   overdispersion parameter = 5.1
```

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)
```

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

```
fit13 <- glm(incidents ~ ., data = ships, family = poisson)
display(fit13)
```

```
## glm(formula = incidents ~ ., family = poisson, data = ships)
##             coef.est coef.se
## (Intercept) -5.71     1.22
## typeB        0.81     0.20
## typeC       -1.20     0.33
## typeD       -0.86     0.29
## typeE       -0.22     0.23
## year         0.05     0.01
## period       0.06     0.01
## service      0.00     0.00
## ---
##   n = 40, k = 8
##   residual deviance = 174.0, null deviance = 730.3 (difference = 556.3)
```

The important predictors in this model are the types of ships, types B, C, and D. For a unit increase in the number of type B ships, the number of incidents is multiplied by a factor of 2.24. For a unit increase in the number of type C ships, the number of incidents is reduced by about 70%. And, for a unit increase in the number of type D ships, the number of incidents decreases by around 57.7%.

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)
```

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?

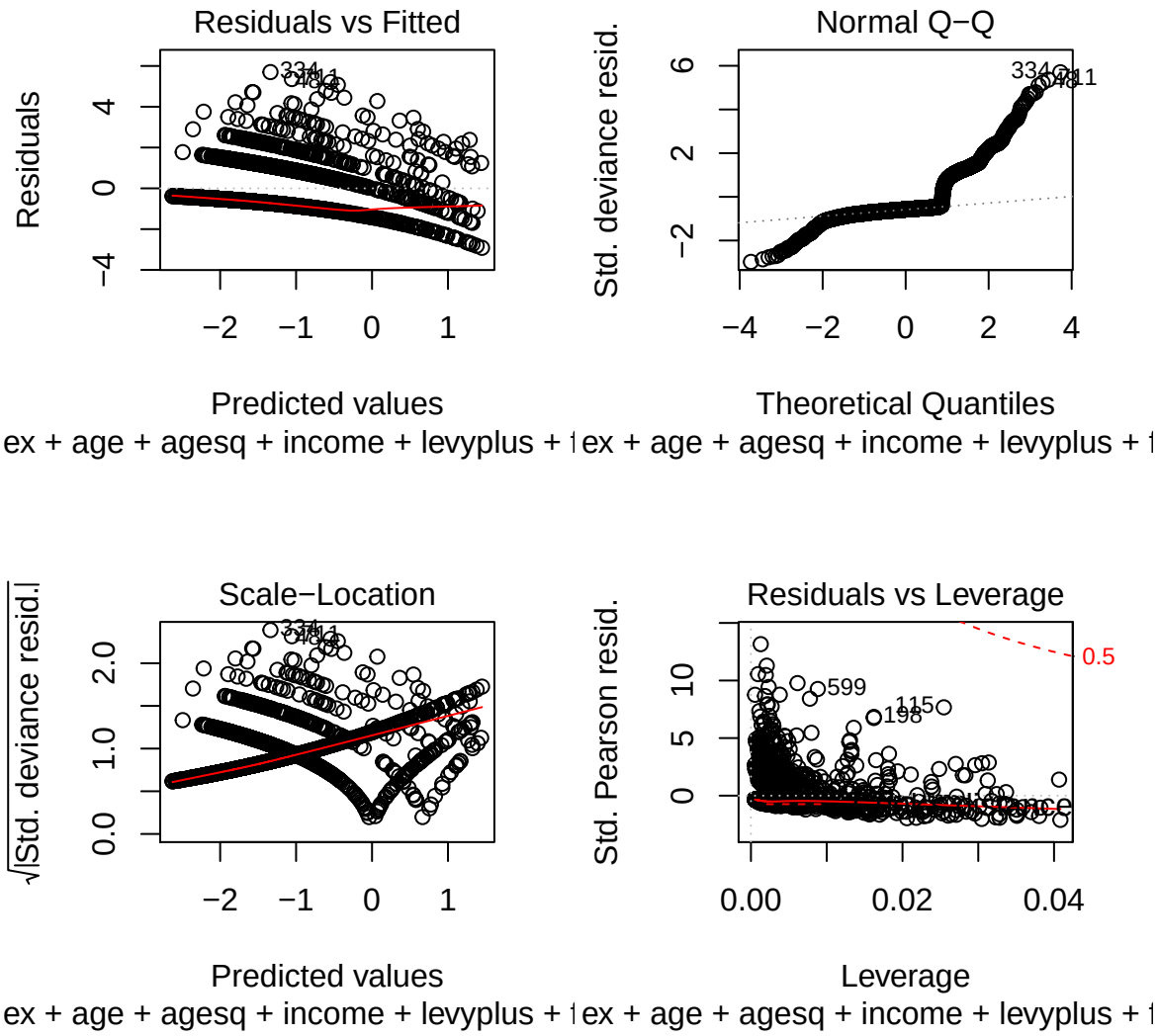
```
fit14 <- glm(doctorco ~ sex + age + agesq + income + levyplus + freepoor + freerepa + illness + actdays
summary(fit14)
```

```
##
## Call:
## glm(formula = doctorco ~ sex + age + agesq + income + levyplus +
##     freepoor + freerepa + illness + actdays + hscore + chcond1 +
##     chcond2, family = poisson, data = dvisits)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9170  -0.6862  -0.5743  -0.4839   5.7005
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.223848   0.189816 -11.716   <2e-16 ***
## sex          0.156882   0.056137   2.795   0.0052 **
## age          1.056299   1.000780   1.055   0.2912
## agesq       -0.848704   1.077784  -0.787   0.4310
## income      -0.205321   0.088379  -2.323   0.0202 *
## levyplus     0.123185   0.071640   1.720   0.0855 .
## freepoor    -0.440061   0.179811  -2.447   0.0144 *
## freerepa     0.079798   0.092060   0.867   0.3860
## illness      0.186948   0.018281  10.227   <2e-16 ***
## actdays      0.126846   0.005034  25.198   <2e-16 ***
## hscore       0.030081   0.010099   2.979   0.0029 **
## chcond1      0.114085   0.066640   1.712   0.0869 .
## chcond2      0.141158   0.083145   1.698   0.0896 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 5634.8  on 5189  degrees of freedom
## Residual deviance: 4379.5  on 5177  degrees of freedom
## AIC: 6737.1
##
## Number of Fisher Scoring iterations: 6
```

The residual deviance is quite high.

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

```
plot(fit14)
```



Since the number of doctor visits takes discrete values, the residuals take finitely many values. Here, each line represents a different posiible value and hence, we see lines of observations.

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

Age seems to be a significant predictor, but, its coefficient is not statistically significant. The coefficients of sex, income, levyplus, freepoor, illness, actdays and hscore are statistically significant. Older females with more illness may tend to have the more doctor visits.

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.

```
last_person <- predict(fit14, dvisits[5190,], type = "response")
#the mean for the predicted number of visits for the last person is 0.1533.
#Therefore, considering lambda = 0.153
#Calculating probabilities of visits = 0,1,2,...etc.
pr <- 0
for (i in 0:4){
  pr[i] <- print(paste0("Prob. of ", i, " visits: ", dpois(i, lambda = 0.153)))
```

```
}
```

```
## [1] "Prob. of 0 visits: 0.858129721811394"
## [1] "Prob. of 1 visits: 0.131293847437143"
## [1] "Prob. of 2 visits: 0.0100439793289415"
## [1] "Prob. of 3 visits: 0.000512242945776013"
## [1] "Prob. of 4 visits: 1.95932926759325e-05"
```

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

```
fit15 <- lm(doctorco ~ sex + age + agesq + income + levyplus + freepoor + freerepa + illness + actdays +
summary(fit15)
```

```
##
## Call:
## lm(formula = doctorco ~ sex + age + agesq + income + levyplus +
##     freepoor + freerepa + illness + actdays + hscore + chcond1 +
##     chcond2, data = dvisits)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1352 -0.2588 -0.1435 -0.0433  7.0327
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.027632   0.072220   0.383  0.70202
## sex          0.033811   0.021604   1.565  0.11764
## age          0.203201   0.410016   0.496  0.62020
## agesq       -0.062103   0.458716  -0.135  0.89231
## income      -0.057323   0.033089  -1.732  0.08326 .
## levyplus     0.035179   0.024882   1.414  0.15748
## freepoor    -0.103314   0.052471  -1.969  0.04901 *
## freerepa     0.033241   0.038157   0.871  0.38371
## illness      0.059946   0.008357   7.173 8.39e-13 ***
## actdays      0.103192   0.003657  28.216  < 2e-16 ***
## hscore       0.016976   0.005190   3.271  0.00108 **
## chcond1      0.004384   0.023740   0.185  0.85349
## chcond2      0.041617   0.035863   1.160  0.24592
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7139 on 5177 degrees of freedom
## Multiple R-squared:  0.2018, Adjusted R-squared:    0.2
## F-statistic: 109.1 on 12 and 5177 DF,  p-value: < 2.2e-16
```

```
predict(fit15, dvisits[5190,], type = "response")
```

```
##      5190
## 0.1606531
```

The Gaussian and the Poisson model are not very different from each other for this data. The Poisson model yields more statistically significant coefficients than the Gaussian model. But, the Poisson model has a residual deviance of 4379.5 for 5190 degrees of freedom, which shows it is not a very good fit. Though the Gaussian model yields lower standard errors, the model has an R-Squared value of 0.2, which shows that the model explains 20% of the variation in the number of visits to the doctor. Thus, the Gaussian model is also not a very good fit.