Homework1

Bohao Tang

April 8, 2018

1

It's reasonable at a glance to assume that the number of moons conditionally follow a possion distribution. So we use possion glm here to analysis the data.

We may answer the question by fit the model of formula `Moons ~ Distance + Diameter * Mass`, compare it with its submodels and correct for overdispersion.

But in this dataset, even you fit a more complicated formula, say `Moons ~ Distance * Diameter * Mass`, the residual deviance is significant large:

```r
library(tidyverse)
planets = read.csv("Ex0206.csv")

fit = glm(Moons ~ Distance * Diameter * Mass, family = poisson, data=planets)
1 - pchisq( fit$deviance, fit$df.residual )
```

```
## [1] 0.03626231
```

We may deal it as overdispersion or raise power in our formula to gain complexity. If we know nothing about the data, this is fine. But as for this dataset, it's obvious that the probability to form a moon (proportional to poisson mean) has nothing to do with an **exponential of polynomial** of this three variables. So it's a very bad statement if we answer the question through this way.

However I'm not going to do an entire mathematical modeling here, but at least add some variables based on some basic knowledge of physics.

First, it's reasonable to assume that the probability to form a moon p_m is propotional to the probability of a random particle near the planet surface having velocity smaller than the escape velocity v_e. From Maxwell velocity distribution, we can approximate $log(p_m)$ as $c_0 + c_1 v_e^2 + c_2 log(v_e)$ where c_0, c_1, c_2 are unknown parameters to be regressioned. And v_e is proportional to $\sqrt{\frac{Mass}{Diameter}}$. Notice that $log(v_e)$ is linear combination of $log(Mass)$ and $log(Diameter)$.

Second, the sun and the space near planet may also influence the probability, then we add surface area, surface gravity, sun gravity, density and so on. But they are more reasonable to appear in log scale and you will find out they are all linear combination of $log(Mass), log(Distance), log(Diameter)$. So we just add this three term.

You will find that even we just change variables to the log scale and add the v_e^2 term. We use less variables than above, but the residual deviance is insignificant as below. (Notice that if you don't add the v_e^2 term, the deviance will become significant. So the escape velocity assumption seems relativly reasonable)

```r
fit = glm(Moons ~ log(Distance) + log(Mass) + log(Diameter) + I(Mass/Diameter),
          family = poisson, data = planets)
summary(fit)
```

```
##
## Call:
## glm(formula = Moons ~ log(Distance) + log(Mass) + log(Diameter) +
##     I(Mass/Diameter), family = poisson, data = planets)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5056  -0.9164  -0.3249   0.1029   1.6445
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -0.06243    0.55283  -0.113  0.91008
## log(Distance)     0.25984    0.14727   1.764  0.07767 .
## log(Mass)        -0.79051    0.24382  -3.242  0.00119 **
## log(Diameter)     3.08865    0.63846   4.838 1.31e-06 ***
## I(Mass/Diameter)  0.03069    0.01221   2.512  0.01199 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 388.253  on 12  degrees of freedom
## Residual deviance:  14.006  on  8  degrees of freedom
## AIC: 58.022
##
## Number of Fisher Scoring iterations: 5
```

```r
1 - pchisq( fit$deviance, fit$df.residual )
```

```
## [1] 0.08160063
```

We can use best subset selection to select a best model after I add those variables. But for simplicity we just use the model showed above. Now the chisq goodness-of-fit test shows the fit is acceptable without adding overdispersion term.

```r
1 - pchisq(sum(fit$residuals^2 / fit$fitted.values), fit$df.residual)
```

```
## [1] 0.1970039
```

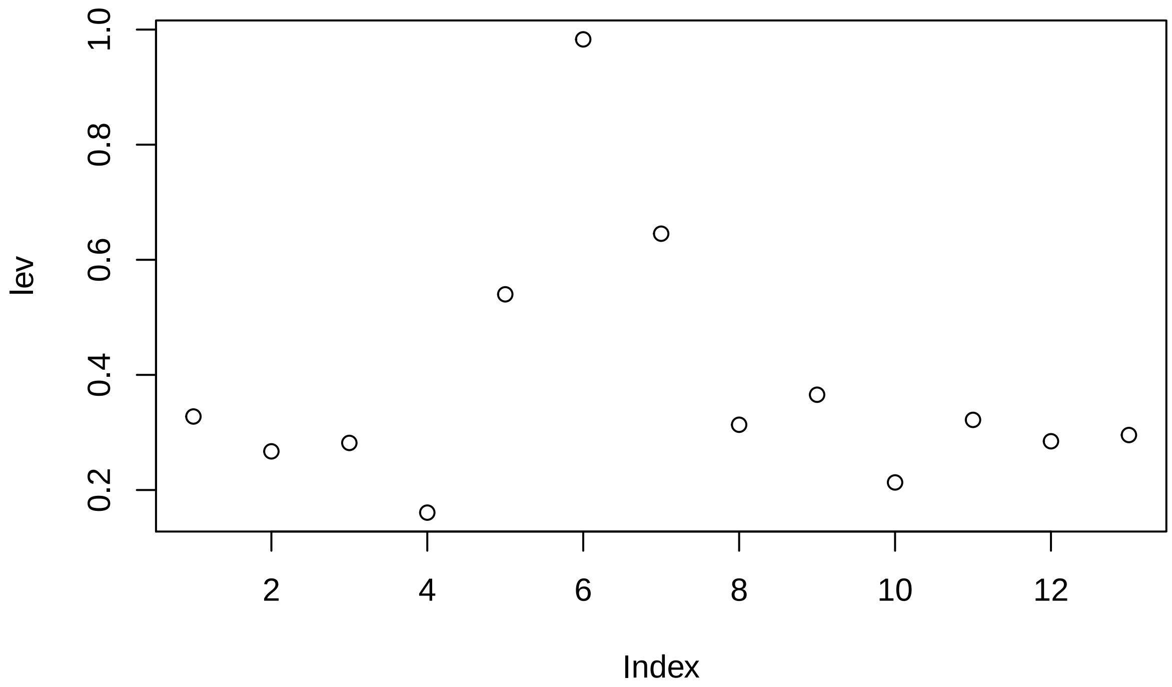
Then we have the moon number formula as (only in the mean value sense):

$$Moon = \frac{Distance^{0.2598} Diameter^{3.0887}}{Mass^{0.7905}} e^{0.03069 \frac{Mass}{Diameter} - 0.06243}$$

And for the question, it's not precise to state which is more influencial. If you just see the model, and say `Diameter` is more significant and in the formula it has bigger power. But the `Diameter` and `Mass` may not influence the result directly, maybe is density more influencial and somthing like this. For example, as shown in this formula, you can't directly tell if a larger (in diameter) planet will have more moons. But you can say for the planets with fixed escape velocity, the larger it is, the more moons it will have. This is a highly reasonable result. Since if a planet have a fixed escape velocity, things are equally hard to escape it in some sense, so the larger diameter it has, it will have more space near the planet to form moons. (Actually you will find that if you fixed v_e and `Distance`, the Moon number will approximatly proportional to `Diameter^2`, which is just the surface area of the planet, thus the indicator of space near the planet).

Then we plot the leverage to find isolated influencial points.

```r
lev = hat(model.matrix(fit))
plot(lev)
```



Then there are only one isolated (extremely) influencial point as `Ceres` for too small (in Mass and Diameter). Notice that if you do formula `Moons ~ Distance + Diameter * Mass`, there will be three this influential point.

2

We use clogit of package `survival` to answer these question. That's a function to maximize conditionally likelihood.

(1)

```r
library(survival)

cancer = read.table("endometrial.txt", header = TRUE)

cfit = clogit(Case ~ Est + Gall + Hyp + Ob + Non,
              data = cancer)
summary(cfit)
```

```
## Call:
## coxph(formula = Surv(rep(1, 315L), Case) ~ Est + Gall + Hyp +
##     Ob + Non, data = cancer, method = "exact")
##
##   n= 315, number of events= 63
##
```

```
##         coef exp(coef) se(coef)      z Pr(>|z|)    
## Est   1.8604    6.4264   0.4405  4.223 2.41e-05 ***
## Gall  1.1164    3.0537   0.3831  2.914  0.00356 ** 
## Hyp  -0.1425    0.8672   0.3259 -0.437  0.66196    
## Ob    0.3248    1.3838   0.3197  1.016  0.30970    
## Non   0.5140    1.6720   0.4841  1.062  0.28828    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
##      exp(coef) exp(-coef) lower .95 upper .95
## Est     6.4264     0.1556    2.7100    15.239
## Gall    3.0537     0.3275    1.4414     6.470
## Hyp     0.8672     1.1531    0.4579     1.642
## Ob      1.3838     0.7227    0.7394     2.590
## Non     1.6720     0.5981    0.6474     4.318
## 
## Rsquare= 0.137   (max possible= 0.626 )
## Likelihood ratio test= 46.52  on 5 df,   p=7.107e-09
## Wald test            = 32.88  on 5 df,   p=3.98e-06
## Score (logrank) test = 42.42  on 5 df,   p=4.837e-08
```

So we can tell that `Estrogen usage` and `Gallbladder disease` are significant and the rest `Hypertension`, `Obesity` and `Non-estrogen drug` are not significant.

(2)

Fit the model:

```
cfit.sig = clogit(Case ~ Est + Gall, data = cancer)
summary(cfit.sig)
```

```
## Call:
## coxph(formula = Surv(rep(1, 315L), Case) ~ Est + Gall, data = cancer, 
##     method = "exact")
## 
##   n= 315, number of events= 63 
## 
##        coef exp(coef) se(coef)    z Pr(>|z|)    
## Est  2.0154    7.5040   0.4234 4.76 1.94e-06 ***
## Gall 1.1510    3.1612   0.3798 3.03  0.00244 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
##      exp(coef) exp(-coef) lower .95 upper .95
## Est      7.504     0.1333     3.273    17.207
## Gall     3.161     0.3163     1.502     6.655
## 
## Rsquare= 0.13   (max possible= 0.626 )
## Likelihood ratio test= 43.89  on 2 df,   p=2.952e-10
## Wald test            = 31.51  on 2 df,   p=1.44e-07
## Score (logrank) test = 40.47  on 2 df,   p=1.628e-09
```

And see the statistics in the `Likelihood ratio test` term, subtruct to statistics and test on chisq with df be the difference of df among two models. That is:

```r
1 - pchisq(46.52 - 43.89, 3)
```

```
## [1] 0.4522544
```

Far larger than 0.05, so we don't suffer a significant loss for dropping those variables. Therefore we can say extra factors add no predictive information in this sense.

(3)

That can be found by fit the model:

```r
cfit.co = clogit(Case ~ Est * Gall, data = cancer)
summary(cfit.co)
```

```
## Call:
## coxph(formula = Surv(rep(1, 315L), Case) ~ Est * Gall, data = cancer,
##     method = "exact")
##
##   n= 315, number of events= 63
##
##             coef exp(coef) se(coef)      z Pr(>|z|)
## Est       2.7094   15.0208   0.6113  4.432 9.34e-06 ***
## Gall      2.9634   19.3634   0.8454  3.505 0.000456 ***
## Est:Gall -2.2266    0.1079   0.9410 -2.366 0.017976 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##          exp(coef) exp(-coef) lower .95 upper .95
## Est        15.0208    0.06657   4.53228   49.7819
## Gall       19.3634    0.05164   3.69302  101.5269
## Est:Gall    0.1079    9.26792   0.01706    0.6824
##
## Rsquare= 0.145   (max possible= 0.626 )
## Likelihood ratio test= 49.48  on 3 df,   p=1.029e-10
## Wald test            = 25.5  on 3 df,   p=1.215e-05
## Score (logrank) test = 41.38  on 3 df,   p=5.423e-09
```

Since the interaction term `Est:Gall` is significant, we can say this two factor have interaction in causing Endometrial cancer.