STAT 528 - Advanced Regression Analysis II

Data Separation (part II)

Daniel J. Eck
Department of Statistics
University of Illinois

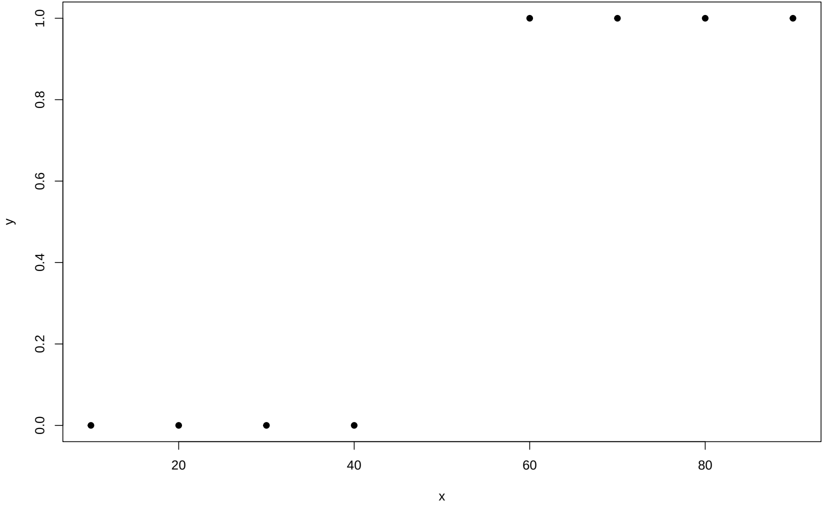
Today's Learning Objectives

- data separation introduction
- statistical inference when separation exists

This slide deck will only contain data analyses and computational tools.

Recall our 8 point data set which is taken from Section 6.5.1 in Agresti.

```
x <- (1:9 * 10)[-5]
y <- c(0,0,0,0,1,1,1,1)
plot(x, y, pch = 19)
```



Last time we demonstrated that there is a hyperplane which separates the successes and failures.

```
## separation vector
b <- c(-50, 1)

## model matrix
M <- cbind(1, x)

## check condition
cbind(M %*% b, y)
```

```
##          y
## [1,] -40 0
## [2,] -30 0
## [3,] -20 0
## [4,] -10 0
## [5,]  10 1
## [6,]  20 1
## [7,]  30 1
## [8,]  40 1
```

As we can see below the glm output is nonsense.

```r
m1 <- glm(y ~ x, family = "binomial")
```

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

```r
## summary table
summary(m1)
```

```
##
## Call:
## glm(formula = y ~ x, family = "binomial")
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -1.045e-05  -2.110e-08   0.000e+00   2.110e-08   1.045e-05
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -118.158 296046.174       0        1
## x               2.363   5805.939       0        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.1090e+01  on 7  degrees of freedom
## Residual deviance: 2.1827e-10  on 6  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 25
```

The data exhibits complete degeneracy, and statistical inference is essentially meaningless.

Luckily the computational checks have warned the user that a potential problem has occurred.

Note that these warning messages do not describe what the problem is or provide any guidance for how users should handle this problem.

Morevover, these checks are purely computational and do not necessarily imply that separation has occurred.

```r
w <- c(x, 1e3)
z <- c(0,0,0,1,0,1,1,1,1)
m1_alt <- glm(z ~ w, family = "binomial")
```

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

```r
summary(m1_alt)
```

```
##
## Call:
## glm(formula = z ~ w, family = "binomial")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5422  -0.4014   0.0000   0.4014   1.5422
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.13039    2.74687  -1.504    0.133
## w            0.08261    0.05075   1.628    0.104
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12.3653  on 8  degrees of freedom
## Residual deviance:  5.9245  on 7  degrees of freedom
## AIC: 9.9245
##
## Number of Fisher Scoring iterations: 8
```

Notice that in the Agresti example the number of Fisher scoring iterations is 25 (maximum allowable iterations). Look at what happens when change the defaults:

```
m2 <- glm(y ~ x, family = "binomial",
          control = list(maxit = 1e4, epsilon = 1e-100))
```

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

```
summary(m2)
```

```
##
## Call:
## glm(formula = y ~ x, family = "binomial", control = list(maxit = 10000,
##     epsilon = 1e-100))
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.107e-08  -2.107e-08   0.000e+00   2.107e-08   2.107e-08
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.546e+02  4.939e+07       0        1
## x            3.092e+00  8.664e+05       0        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.1090e+01  on 7  degrees of freedom
## Residual deviance: 3.5527e-15  on 6  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 33
```

Returning to the model fit with the default settings, we see large canonical parameter estimates and mean value parameter estimates that are at the boundary of the closure of their parameter space ($0 < p < 1$).

Informally, estimates are “at infinity.’ ’

```
## submodel canonical parameter estimates
betahat <- m1$coefficients
betahat
```

```
## (Intercept)           x
##  -118.15802     2.36316
```

```
## saturated model canonical parameter estimates
thetahat <- as.numeric(M %*% betahat)
thetahat
```

```
## [1] -94.52642 -70.89481 -47.26321 -23.63160  23.63160  47.26321  70.89481
## [8]  94.52642
```

```
## saturated model mean value parameter estimates
phat <- predict(m1, type = "response")
phat
```

```
##            1            2            3            4            5            6
## 2.220446e-16 2.220446e-16 2.220446e-16 5.456633e-11 1.000000e+00 1.000000e+00
##            7            8
## 1.000000e+00 1.000000e+00
```

Canonical statistic on the boundary of its support

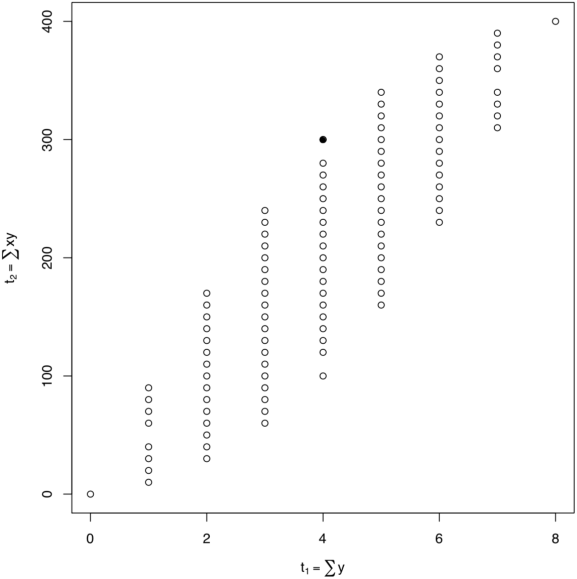
Recall that the submodel can be written as

$$\langle y, M\beta \rangle - c(M\beta) = \langle M^T y, \beta \rangle - c_\beta(\beta)$$

Also notice that the observed value of the canonical statistic $M^T y$

for the above submodel is on the boundary of the support of values for $M^T Y$.

This implies that the MLE does not exist in the traditional sense. See Geyer (2009) for technical details that are beyond the scope of this course).


$t_2 = \sum xy$
0
100
200
300
400
0
2
4
6
8
$t_1 = \sum y$

Note that, by the observed equals expected property, $M^T y$ is the MLE of the submodel mean-value parameter vector. The variance of the submodel canonical statistic is the Fisher information matrix.

We now use R scripts to compute the Fisher information matrix. An eigenvector decomposition reveals that the Fisher information matrix is numerically singular.

```
invFI <- vcov(m1)
FI <- solve(invFI)
eigen(FI)
```

```
## eigen() decomposition
## $values
## [1] 7.715655e-07 1.140566e-11
##
## $vectors
##             [,1]        [,2]
## [1,] 0.01922747 -0.99981514
## [2,] 0.99981514  0.01922747
```

This implies that $\widehat{\text{Var}}(M^T Y) = 0$.

Therefore, the MLE solution to this problem is that the observed data which are on the boundary are the only possible data that could have occurred.

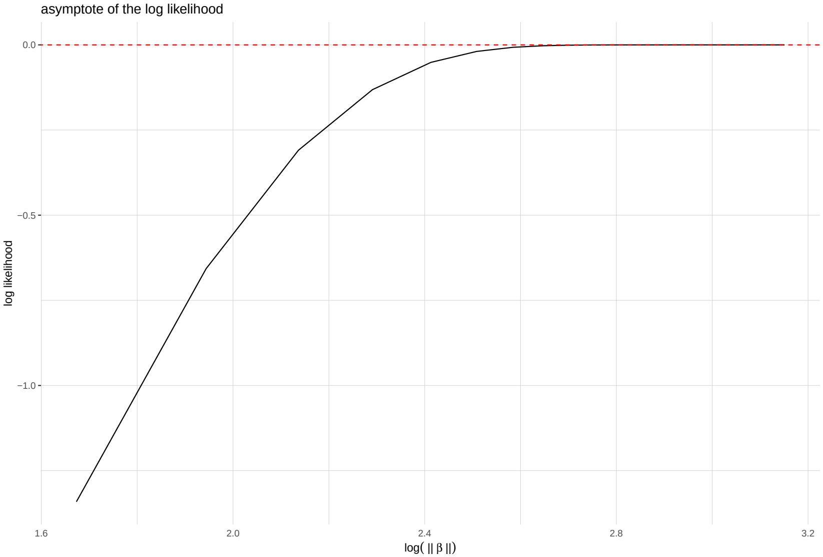
Of course, these are estimates and not actual parameters. The problem is how to make inferential statements about model parameters given this degeneracy.

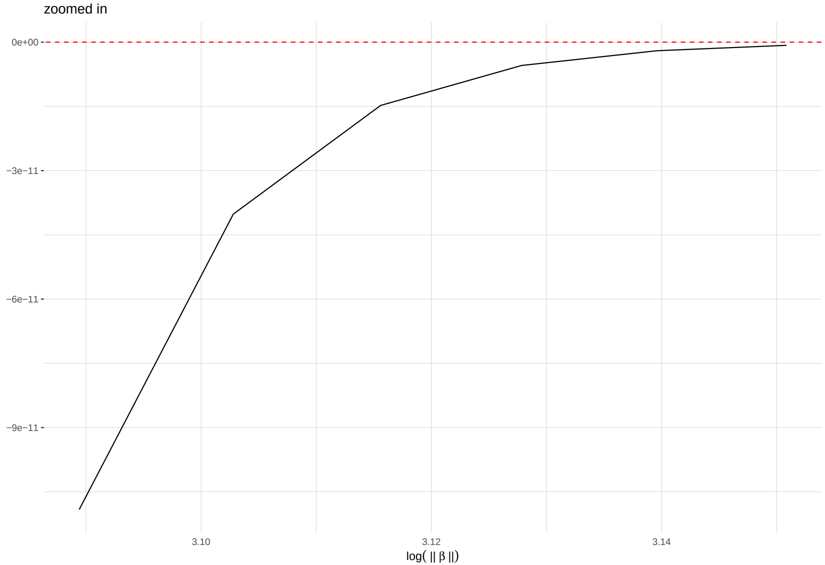
Inference in the complete separation setting

The exponential family in this example is completely degenerate. The MLE does not exist in the traditional sense, but may (does) exist in the completion of the exponential family (the set which includes all exponential family distributions and all limits of distributions).

Conventional maximum likelihood computations come close, in a sense, to finding the MLE in the completion of the exponential family. They go uphill on the likelihood function until they meet their convergence criteria and stop.

```r
asymptote <- t(sapply(1:30, function(iter){
  m1 <- glm(y ~ x, family = "binomial", control = list(maxit = iter, epsilon = 1e-20))
  c(sqrt(log(crossprod(coef(m1)))), logLik(m1))
}))
asymptote <- as.data.frame(asymptote)
```


asymptote of the log likelihood
0.0
−0.5
−1.0
log likelihood
1.6
2.0
2.4
2.8
3.2
log(|| β ||)


zoomed in
0e+00
−3e−11
−6e−11
−9e−11
3.10
3.12
3.14
log(|| β ||)

At this point, canonical parameter estimates $\hat{\theta}$ and $\hat{\beta}$ are still infinitely far away from the MLE in the completion.

But mean value parameter estimates $\hat{\mu}$ are close to the MLE in the completion, and the log likelihood is nearly maximized.

And the corresponding probability distributions are close in total variation norm to the MLE probability distribution in the completion.

Software and return to Agresti example

The `inference` function in the R package `glmdr` determines one-sided confidence intervals for mean value parameters corresponding to response values that are constrained to their observed values y_I

```
library(glmdr)
```

- See the `glmdr` directory in the stat528resources repo. You will have to install the package locally
- The `glmdr` package currently only works for logistic, binomial, and Poisson regression.

We now fit the logistic regression model using the `glmdr` fitting function instead of `glm`. The user is prompted with a description of the problem which provides guidance on how to handle the problem.

```r
m_glmdr <- glmdr(y ~ x)

## summary information
summary(m_glmdr)
```

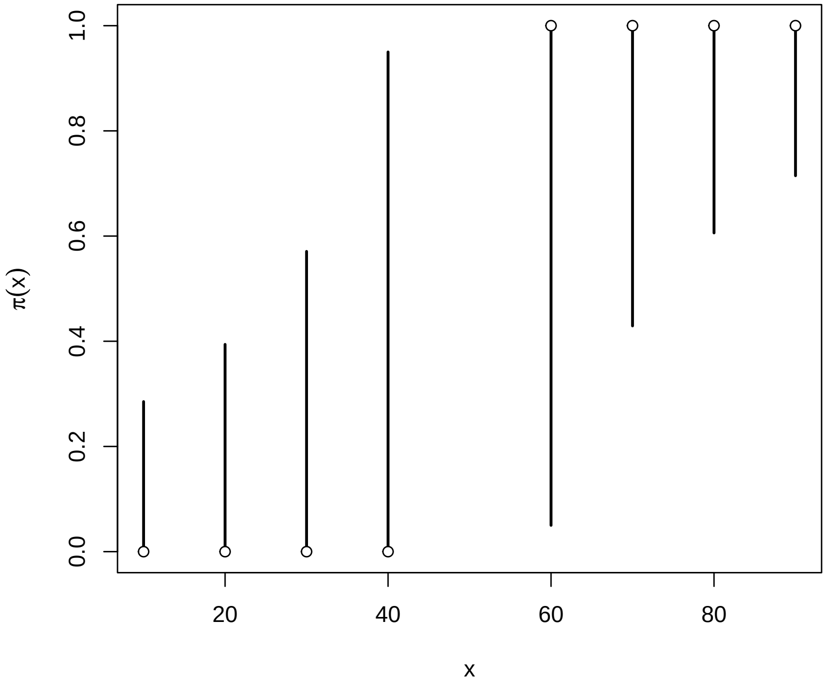
```
## MLE exists in Barndorff-Nielsen completion
## it is completely degenerate
## the MLE says the response actually observed is the only
## possible value that could ever be observed

## $overview
## NULL
##
## $type
## [1] "degenerate"
##
## $linearity
## [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##
## attr(,"class")
## [1] "summary.glmdr"
```

We then use the `inference` function to obtain one sided confidence intervals for mean-value parameters corresponding to components Y_I that are constrained to be their observed values.

```
## one-sided CIs
CIs <- inference(m_glmdr)
CIs
```

```
##       lower     upper
## 1 0.0000000 0.2852500
## 2 0.0000000 0.3940359
## 3 0.0000000 0.5708292
## 4 0.0000000 0.9500000
## 5 0.0500000 1.0000000
## 6 0.4291708 1.0000000
## 7 0.6059641 1.0000000
## 8 0.7147500 1.0000000
```


1.0
0.8
0.6
0.4
0.2
0.0
π(x)
20
40
60
80
x

Commentary on Agresti

The $n = 8$ data point example that we analyzed comes from Section 6.5.1 in Agresti.

However, this textbook provides no model based inferential solution to this problem.

This week, we provided such a solution that exists within the exponential family modeling and maximum likelihood estimation paradigms.

To be fair to Agresti, “solutions” to complete separation are discussed in Sections 7.4.8.

Not completely degenerate

In the Agresti example we noticed that the estimated Fisher information matrix was completely degenerate. T

his need not be so in generality, the Fisher information matrix can exhibit partial degeneracy.

When this is so the limiting conditional model (LCM) is not completely degenerate, data pairs (y_i, x_i) corresponding to response vectors which are left unconstrained form the LCM and parameter estimation can be conducted in a traditional manner.

We will now explore an example where with partial degeneracy.

Endometrial example

We will consider the endometrial example in which the a histology grade and risk factors for 79 cases of endometrial cancer are analyzed. the variables are:

- **NV**: neovasculization with coding 0 for absent and 1 for present
- **PI**: pulsality index of arteria uterina
- **EH**: endometrium heigh
- **HG**: histology grade with coding 0 for low grade and 1 for high grade

```
library(enrichwith)
data(endometrial)
head(endometrial, 3)
```

```
##   NV PI   EH HG
## 1  0 13 1.64  0
## 2  0 16 2.26  0
## 3  0  8 3.14  0
```

We begin with a standard logistic regression fit.

```r
m <- glm(HG ~ ., data = endometrial, family = "binomial",
         x = TRUE, y = TRUE)
summary(m)
```

```
##
## Call:
## glm(formula = HG ~ ., family = "binomial", data = endometrial,
##     x = TRUE, y = TRUE)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.50137  -0.64108  -0.29432   0.00016   2.72777
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   4.30452    1.63730   2.629 0.008563 **
## NV           18.18556 1715.75089   0.011 0.991543
## PI           -0.04218    0.04433  -0.952 0.341333
## EH           -2.90261    0.84555  -3.433 0.000597 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 104.903  on 78  degrees of freedom
## Residual deviance:  55.393  on 75  degrees of freedom
## AIC: 63.393
##
## Number of Fisher Scoring iterations: 17
```

We observe **quasi-complete separation** in NV (a categorical variable with two levels), where we note that a 2×2 contingency table with an empty (zero) cell is an example of quasi-complete separation.

```
b <- c(0,1,0,0)
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose
```

```
foo <- setDT(as.data.frame(cbind(m$y, m$x %*% b)))
colnames(foo) <- c("y", "sep")
foo[, .(.N), by = c("y", "sep")]
```

```
##    y sep  N
## 1: 0   0 49
## 2: 1   0 17
## 3: 1   1 13
```

We now use glmdr to do our fitting. The user is prompted with a description of the problem which provides guidance on how to handle the problem.

```r
m_glmdr <- glmdr(HG ~ ., data = endometrial, family = "binomial")
summary(m_glmdr)
```

```
## MLE exists in Barndorff-Nielsen completion
## it is conditional on components of the response
## corresponding to object$linearity == FALSE being
## conditioned on their observed values

## $overview
## NULL
##
## $type
## [1] "lcm"
##
## $summary
##
## Call:
## stats::glm(formula = HG ~ ., family = "binomial", data = endometrial,
##     subset = c("1", "2", "3", "4", "5", "6", "7", "8", "9", "10",
##     "11", "12", "13", "14", "15", "16", "17", "18", "19", "20",
##     "21", "27", "28", "29", "30", "31", "32", "33", "34", "35",
##     "36", "37", "38", "39", "40", "41", "42", "43", "44", "45",
##     "46", "47", "52", "53", "54", "55", "56", "57", "58", "59",
##     "60", "61", "62", "63", "64", "65", "66", "67", "68", "69",
##     "70", "72", "73", "74", "77", "79"), x = TRUE, y = TRUE)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5014  -0.6634  -0.3856   0.2126   2.7278
##
```

The summary table is strange. The variable driving quasi-complete separation has been dropped from consideration.

We only observe partially degeneracy of Fisher Information (the variance matrix for the submodel canonical statistic).

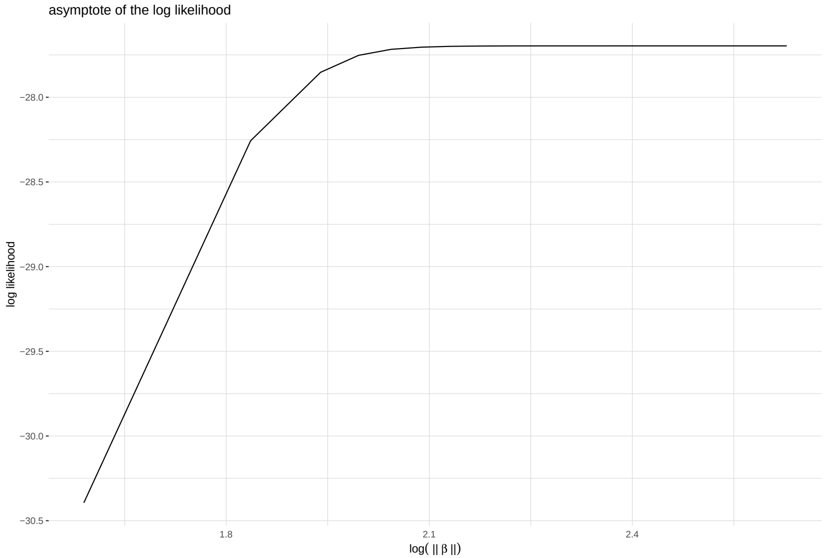
```
solve(vcov(m))
```

```
##              (Intercept)           NV           PI           EH
## (Intercept) 8.971281e+00 3.396969e-07 1.499626e+02 1.334686e+01
## NV          3.396969e-07 3.396969e-07 8.916423e-06 3.821298e-07
## PI          1.499626e+02 8.916423e-06 3.045253e+03 2.164762e+02
## EH          1.334686e+01 3.821298e-07 2.164762e+02 2.133680e+01
```

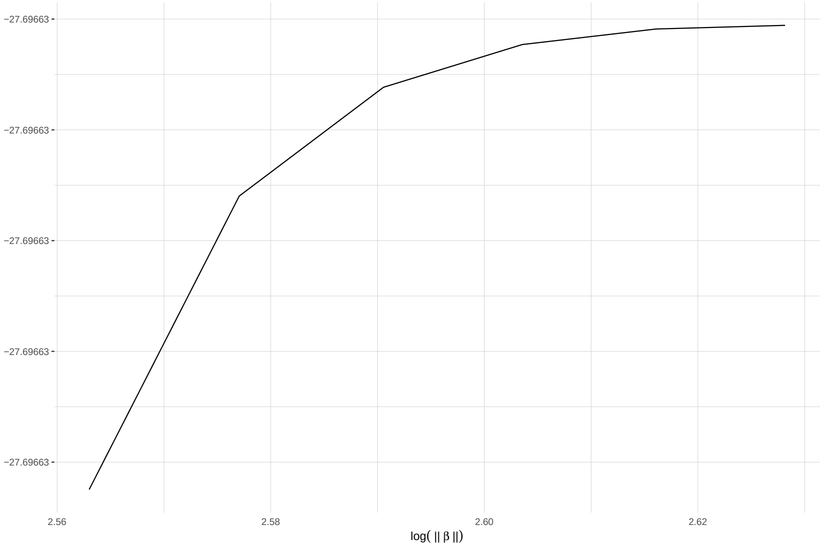
The reason for this strangeness is beyond the scope of this course. The interested reader can check out Lemma 1 and Corollary 1 in Section 6.1.3. in Eck and Geyer (2021).

We now investigate what is going on under the hood.

```r
asymptote <- t(sapply(1:30, function(iter){
  m_test <- glm(HG ~ ., family = "binomial", data = endometrial,
                control = list(maxit = iter, epsilon = 1e-20))
  c(sqrt(log(crossprod(coef(m_test)))), logLik(m_test), coef(m_test))
}))
asymptote <- as.data.frame(asymptote)
```


asymptote of the log likelihood
log likelihood
−28.0
−28.5
−29.0
−29.5
−30.0
−30.5
1.8
2.1
2.4
log(|| β ||)

zoomed in


−27.69663
−27.69663
−27.69663
−27.69663
−27.69663
2.56
2.58
2.60
2.62
log(|| β ||)

```r
asymptote %>% mutate(iter = 1:30) %>%
  dplyr::select(3:6) %>% as.data.frame()
```

```
##    (Intercept)        NV          PI        EH
## 1     2.090959  2.355064 -0.02213019 -1.612126
## 2     3.599433  3.144596 -0.03327115 -2.502950
## 3     4.203858  4.148140 -0.04030884 -2.852134
## 4     4.300150  5.172614 -0.04205976 -2.900986
## 5     4.304349  6.181031 -0.04217827 -2.902550
## 6     4.304501  7.183901 -0.04218293 -2.902599
## 7     4.304516  8.184948 -0.04218334 -2.902605
## 8     4.304517  9.185332 -0.04218340 -2.902606
## 9     4.304518 10.185474 -0.04218340 -2.902606
## 10    4.304518 11.185526 -0.04218340 -2.902606
## 11    4.304518 12.185545 -0.04218340 -2.902606
## 12    4.304518 13.185552 -0.04218340 -2.902606
## 13    4.304518 14.185554 -0.04218340 -2.902606
## 14    4.304518 15.185555 -0.04218340 -2.902606
## 15    4.304518 16.185556 -0.04218340 -2.902606
## 16    4.304518 17.185556 -0.04218340 -2.902606
## 17    4.304518 18.185556 -0.04218340 -2.902606
## 18    4.304518 19.185556 -0.04218340 -2.902606
## 19    4.304518 20.185556 -0.04218340 -2.902606
## 20    4.304518 21.185556 -0.04218340 -2.902606
## 21    4.304518 22.185556 -0.04218340 -2.902606
## 22    4.304518 23.185556 -0.04218340 -2.902606
## 23    4.304518 24.185556 -0.04218340 -2.902606
## 24    4.304518 25.185556 -0.04218340 -2.902606
## 25    4.304518 26.185556 -0.04218340 -2.902606
## 26    4.304518 27.185553 -0.04218340 -2.902606
## 27    4.304518 28.185557 -0.04218340 -2.902606
## 28    4.304518 29.185576 -0.04218340 -2.902606
## 29    4.304518 30.185606 -0.04218340 -2.902606
## 30    4.304518 31.185428 -0.04218340 -2.902606
```

We now obtain inference for all mean-value parameters in two steps:

- We first use traditional methods to obtain inferences for mean-value parameters that are unconstrained (the data points forming the LCM).
- Then we can use the `inference` function to obtain one-sided confidence intervals for components of the response vector that are constrained at their observed values.

First we will use traditional methods for inferences mean-value parameters that are unconstrained.

```
m2 <- update(m, subset = m_glmdr$linearity)
summary(m2)
```

```
##
## Call:
## glm(formula = HG ~ ., family = "binomial", data = endometrial,
##     subset = m_glmdr$linearity, x = TRUE, y = TRUE)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5014  -0.6634  -0.3856   0.2126   2.7278
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.30452    1.63720   2.629 0.008559 **
## NV                NA         NA      NA       NA
## PI          -0.04218    0.04433  -0.952 0.341310
## EH          -2.90261    0.84549  -3.433 0.000597 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 75.307  on 65  degrees of freedom
## Residual deviance: 55.393  on 63  degrees of freedom
## AIC: 61.393
##
## Number of Fisher Scoring iterations: 5
```

```r
## get estimates of mean-value parameters in the LCM
preds <- predict(m2, se.fit = TRUE, type = "response")
head(cbind(preds$fit, preds$se.fit))
```

```
##          [,1]        [,2]
## 1 0.268128294 0.074778103
## 2 0.050675634 0.034991612
## 3 0.005782436 0.008046224
## 4 0.007327976 0.010484202
## 5 0.436719783 0.095750088
## 6 0.068407171 0.053127413
```

Then we can use the `inference` function to obtain one-sided confidence intervals for components of the response vector that are constrained at their observed values.

```
## get one-sided CIs for constrained responses
preds_constrained <- inference(m_glmdr)
cbind(endometrial[!m_glmdr$linearity, ], preds_constrained)
```

```
##    NV PI   EH HG     lower upper
## 22  1 38 0.97  1 0.7071451     1
## 23  1 22 1.14  1 0.7432730     1
## 24  1  7 0.88  1 0.9205964     1
## 25  1 25 0.91  1 0.8325905     1
## 26  1 15 0.58  1 0.9518366     1
## 48  1 22 1.44  1 0.5479196     1
## 49  1 40 1.18  1 0.5467740     1
## 50  1  5 0.93  1 0.9160397     1
## 51  1  0 1.17  1 0.8703443     1
## 71  1 49 0.27  1 0.9205150     1
## 75  1 11 1.01  1 0.8703894     1
## 76  1 21 0.98  1 0.8277338     1
## 78  1 19 1.02  1 0.8231611     1
```

We can test for the significance of the NV variable in the presence of quasi-complete separation using traditional means. Methods get harder when the degeneracy exists in the null model as explained in Section 3.15 of Geyer (2009).

```
m_small <- glm(HG ~ PI + EH, data = endometrial, family = "binomial",
         x = TRUE, y = TRUE)
anova(m_small, m, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: HG ~ PI + EH
## Model 2: HG ~ NV + PI + EH
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        76     64.751
## 2        75     55.393  1   9.3576 0.002221 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(m); AIC(m_small)
```

```
## [1] 63.39326
```

```
## [1] 70.7509
```

Other approaches: the problem with priors

We consider three different approaches:

- Flat improper priors: will not work when separation exists
- The weakly informative prior advocated here and implemented in the `bayesglm`
- Jeffrey's prior based approaches advocated for by Ioannis Kosmidis and David Firth in several papers and implemented in the `brglm2` package

We will demonstrate inferential inconsistencies between the later two approaches.

We first show that the bayesglm defaults produce p-values for the NV variable that are close to 0.05.

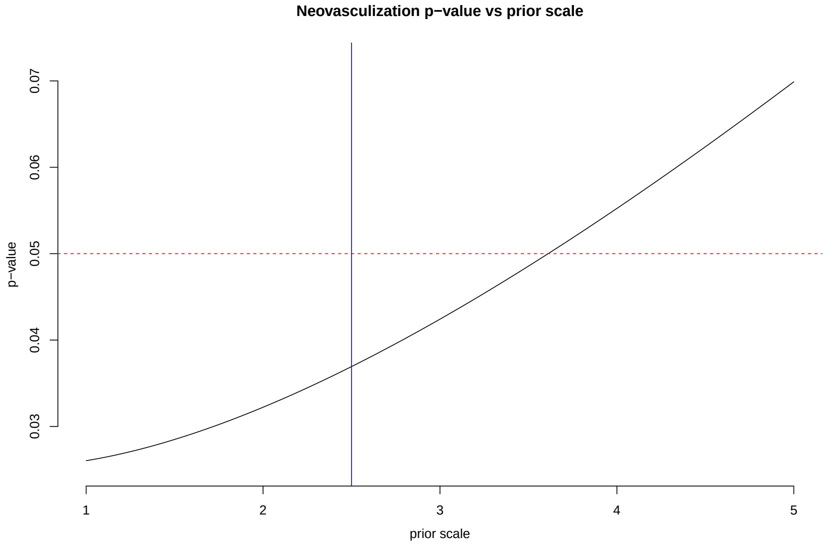
Modest changes to these defaults can change decisions about this variable's significance when testing at the 0.05 level.

```r
library(arm) # for bayesglm
dat <- endometrial
dat[, 2:3] <- scale(dat[, 2:3]) * 0.5

bayes_mod1 <- bayesglm(HG~.,data=dat,family="binomial", prior.scale = 1)
bayes_mod <- bayesglm(HG~.,data=dat,family="binomial") # default value 2.5
bayes_mod5 <- bayesglm(HG~.,data=dat,family="binomial", prior.scale = 5)
bayes_mod10 <- bayesglm(HG~.,data=dat,family="binomial", prior.scale = 10)

# p-values for NV variable
c(summary(bayes_mod1)$coef[2,4], summary(bayes_mod)$coef[2,4],
  summary(bayes_mod5)$coef[2,4], summary(bayes_mod10)$coef[2,4])
```

```
## [1] 0.02603921 0.03693106 0.06989051 0.15449528
```


Neovasculization p−value vs prior scale
p−value
0.03
0.04
0.05
0.06
0.07
prior scale
1
2
3
4
5

Different brglm fitting options yield different results, although these differences do not materialize in different conclusions for the NV variable when testing at the 0.05 significance level.

```r
library(brglm2) # for brglm2
brglm_mod <- glm(HG~.,data=endometrial,family = "binomial",
                 method = "brglm_fit", type = "MPL_Jeffreys")
brglm_mod_AS_mean <- glm(HG~.,data=endometrial,family = "binomial",
                 method = "brglm_fit", type = "AS_mean")
brglm_mod_AS_median <- glm(HG~.,data=endometrial,family = "binomial",
                         method = "brglm_fit", type = "AS_median")
brglm_mod_AS_mixed <- glm(HG~.,data=endometrial,family = "binomial",
                           method = "brglm_fit", type = "AS_mixed")

c(summary(brglm_mod)$coef[2,4], summary(brglm_mod_AS_mean)$coef[2,4],
  summary(brglm_mod_AS_median)$coef[2,4], summary(brglm_mod_AS_mixed)$coef[2,4])
```

```
## [1] 0.05890214 0.05890214 0.09226858 0.05890214
```

Discussion

Which prior do we use?

- **Gelman**: Our prior [Cauchy $\mu = 0$; $\sigma = 2.5$] has the advantage of always giving answers, even when there is complete separation in logistic regression (a common problem, even when the sample size is large and the number of predictors is small), and also automatically applying more shrinkage to higher-order interactions.
- **Kosmidis and Firth**: Penalization of the likelihood by Jeffreys' invariant prior, or a positive power thereof, is shown to produce finite-valued maximum penalized likelihood estimates in a broad class of binomial generalized linear models. The class of models includes logistic regression, where the Jeffreys-prior penalty is known additionally to reduce the asymptotic bias of the maximum likelihood estimator

- **Gelman**: Our approach is similar [to Jeffrey's] but is parameterized in terms of the coefficients and thus allows us to make use of prior knowledge on that scale… We recommend this prior distribution as a default choice for routine applied use.
- **Kosmidis and Firth**: The apparent finiteness and shrinkage properties of the reduced-bias estimator, together with the fact that the estimator has the same first-order asymptotic distribution as the maximum likelihood estimator, are key reasons for the increasingly widespread use of Jeffreys-prior penalized logistic regression in applied work.
- **Gelman**: Scale-free prior distributions such as Jeffreys' do not include enough prior information. As we have seen in specific examples and also in the corpus of datasets, this weakly-informative prior distribution yields estimates that make more sense and perform better predictively, compared to maximum likelihood.

What we have here is a failure to communicate.