Multivariate logistic distribution

June 16, 2019

1 Probit and multivariate logit model

We are interested in modeling multivariate binary outcomes (n observations and k response variables). One way to model this is to use a probit model (this is suggested in the Stan manual; it is different from the mixed model trick but I'll get to that later):

$$\begin{aligned} y_{n,k} &= 1(z_{n,k} > 1) \\ z_n &= x_n\beta + \epsilon_n \\ \epsilon_n &\sim \mathrm{N}(0, \Sigma) \end{aligned} \tag{1}$$

where Σ is a covariance matrix. To avoid identifiability issues, Σ is constrained to have 1's on the diagonals. While this is a convenient way of modeling multivariate binary outcomes, its parameters are difficult to interpret. Instead, the multivariate logistic regression model proposed by O'brien and Dunson (2004) allows marginal distributions to follow a logistic distribution; the parameters of this model can be easily translated to odds ratio:

$$\begin{aligned} y_{n,k} &= 1(z_{n,k} > 1) \\ z_n &= x_n\beta + \log[F_\nu(e_n)/\{1 - F_\nu(e_n)\}] \\ e_n &\sim \mathrm{t}(0, \nu, \Sigma) \end{aligned} \tag{2}$$

where F_ν is the cumulative distribution function of a univariate t distribution with ν degrees of freedom and $t(0, \nu, \Sigma)$ is a multivariate t distribution with ν degrees of freedom, mean 0, and covariance Σ. Again, to avoid identifiability issues, Σ is constrained to have 1's on the diagonals. Sampling directly from this distribution is not trivial (Stan doesn't provide inverse function of CDF of a univariate t distribution); instead, we can use the approximation propsed by O'brien and Dunson (2004) and apply importance sampling to get to the correct posterior. Specifically, the distribution of z_n can be approximated by a multivariate t-distribution with mean $x_n\beta$, variance $\tilde{\sigma}^2 = \pi^2(\nu - 2)/3\nu$ ("a value chosen to make the variances of the univariate t and logistic distributions equal"), and $\tilde{\nu} = 7.3$ degrees of freedom ("a value chosen to minimize the integrated squared distance between the univariate t and univariate logistic densities").

Let's try some examples. We simulate 500 observations using (2) with intercepts only $\beta = (-1, 0, 1)$ and following correlation (covariance) matrix:

$$\Sigma = \begin{pmatrix} 1 & 0.1 & 0.3 \\ 0.1 & 1 & -0.5 \\ 0.3 & -0.5 & 1 \end{pmatrix}.$$

This is what the data looks like:

```
load("../data/logit_t_intercept_only.rda")

head(simulate_dd)

##      V1 V2 V3
## [1,]  1  1  0
## [2,]  1  0  1
## [3,]  1  0  1
## [4,]  1  1  0
## [5,]  1  0  0
## [6,]  1  0  0
```

Then, I fit model (1) and model (2) to this data. First, looking at model (1):

```
load("../analysis/logit_t_intercept_only_logit_gaussian.rda")

ee <- rstan::extract(fit)
```

Looking at estimates of β (intercepts):

```
data.frame(
        median=apply(ee$beta, 2, median),
        lwr=apply(ee$beta, 2, quantile, 0.025),
        upr=apply(ee$beta, 2, quantile, 0.975)
)

##        median        lwr        upr
## 1  0.606912208  0.4954737  0.7326054
## 2  0.001469309 -0.1093642  0.1080785
## 3 -0.566471453 -0.6756364 -0.4523101
```

These are quite different from the β that I simulated with because the logit-normal doesn't preserve the marginal logit property. Interpreting each β as log-odds will be misleading. For example, these are "estimated" odds ratioes (naively assuming that these β estiamtes are log-odds):

```
data.frame(
        median=exp(apply(ee$beta, 2, median)),
        lwr=exp(apply(ee$beta, 2, quantile, 0.025)),
        upr=exp(apply(ee$beta, 2, quantile, 0.975))
)

##      median       lwr       upr
## 1 1.8347573 1.6412755 2.0804941
## 2 1.0014704 0.8964039 1.1141352
## 3 0.5675244 0.5088325 0.6361569
```

The observed odds ratios are quite different:

```
apply(simulate_dd, 2, mean)/(1-apply(simulate_dd, 2, mean))

##       V1       V2       V3
## 2.731343 1.008032 0.396648
```

Can we at least get correlation right?

```
apply(ee$Omega, 2:3, median)

##
##             [,1]       [,2]       [,3]
##   [1,] 1.0000000  0.1131334  0.2271666
##   [2,] 0.1131334  1.0000000 -0.4350445
##   [3,] 0.2271666 -0.4350445  1.0000000

apply(ee$Omega, 2:3, quantile, 0.025)

##
##               [,1]       [,2]        [,3]
##   [1,]  1.00000000 -0.0344030  0.08280362
##   [2,] -0.03440300  1.0000000 -0.55617938
##   [3,]  0.08280362 -0.5561794  1.00000000

apply(ee$Omega, 2:3, quantile, 0.975)

##
##             [,1]       [,2]       [,3]
##   [1,] 1.0000000  0.2550047  0.3808753
##   [2,] 0.2550047  1.0000000 -0.3081267
##   [3,] 0.3808753 -0.3081267  1.0000000
```

Seems like we're doing OK in terms of estimating the correlation.

Let's look at the estimates from the multivariate logisti regression model:

```r
load("../analysis/logit_t_intercept_only_logit_t.rda")

ee <- rstan::extract(fit)
```

Note that we have to take weighted quantiles this time. Look at estimates of β:

```r
## wquant from King et al.
wquant <- function (x, weights, probs = c(0.025, 0.975)) {
    which <- !is.na(weights)
    x <- x[which]
    weights <- weights[which]

    if (all(is.na(x)) || length(x) == 0) return(rep(NA, length(probs)))

    idx <- order(x)
    x <- x[idx]
    weights <- weights[idx]
    w <- cumsum(weights)/sum(weights)
    rval <- approx(w,x,probs,rule=1)
    rval$y
}

data.frame(
        median=apply(ee$beta, 2, wquant, weights=weights, probs=0.5),
        lwr=apply(ee$beta, 2, wquant, weights=weights, probs=0.025),
        upr=apply(ee$beta, 2, wquant, weights=weights, probs=0.975)
)

##          median        lwr        upr
## 1  0.995853424  0.8220403  1.2176655
## 2  0.006185303 -0.1761347  0.1640782
## 3 -0.921081770 -1.1175128 -0.7324204
```

These estimates match with the true values. Moreover, the estimates of odds ratios match with the observed odds (kind of obvious but just checking it):

```r
data.frame(
        median=exp(apply(ee$beta, 2, wquant, weights=weights, probs=0.5)),
        lwr=exp(apply(ee$beta, 2, wquant, weights=weights, probs=0.025)),
        upr=exp(apply(ee$beta, 2, wquant, weights=weights, probs=0.975))
)

##      median       lwr      upr
## 1 2.7070336 2.2751371 3.379290
## 2 1.0062045 0.8385050 1.178306
## 3 0.3980882 0.3270923 0.480744
```

Correlations look reasonable:

```
apply(ee$Omega, 2:3, wquant, weights=weights, probs=0.5)

##
##              [,1]       [,2]       [,3]
##    [1,] 1.0000000  0.1181755  0.2581493
##    [2,] 0.1181755  1.0000000 -0.4389358
##    [3,] 0.2581493 -0.4389358  1.0000000

apply(ee$Omega, 2:3, wquant, weights=weights, probs=0.025)

##
##                [,1]        [,2]       [,3]
##    [1,]  1.00000000 -0.02844068  0.1227903
##    [2,] -0.02844068  1.00000000 -0.5390483
##    [3,]  0.12279029 -0.53904828  1.0000000

apply(ee$Omega, 2:3, wquant, weights=weights, probs=0.975)

##
##              [,1]       [,2]       [,3]
##    [1,] 1.0000000  0.2657659  0.4059446
##    [2,] 0.2657659  1.0000000 -0.3141494
##    [3,] 0.4059446 -0.3141494  1.0000000
```

2 Improving probit model

O'brien and Dunson (2004) suggests using a t distribution for approximating the multivariate logit distribution but we can also use the probit model. Instead of using (1), we can rewrite it as

$$\begin{aligned} y_{n,k} &= 1(z_{n,k} > 1) \\ z_n &= x_n\beta + \epsilon_n \\ \epsilon_n &\sim \mathrm{N}(0, \sigma^2\Sigma) \end{aligned} \tag{3}$$

where Σ is a correlation matrix and $\sigma^2 = \pi^2/3$. This allows the marginal distribution of z_n to have same variance as a standard logistic distribution. We fit this model to the simulated data above:

```
load("../analysis/logit_t_intercept_only_logit_gaussian2.rda")

ee <- rstan::extract(fit)
```

Looking at estimates of β:

```r
data.frame(
        median=apply(ee$beta, 2, median),
        lwr=apply(ee$beta, 2, quantile, 0.025),
        upr=apply(ee$beta, 2, quantile, 0.975)
)
```

```
##        median        lwr        upr
## 1  1.12086705  0.9278208  1.3546859
## 2  0.01113124 -0.1854714  0.2043039
## 3 -1.03712738 -1.2560094 -0.8314504
```

This matches the true value much better. Converting these to odds ratios is consistent with the observed odds ratios.

```r
data.frame(
        median=exp(apply(ee$beta, 2, median)),
        lwr=exp(apply(ee$beta, 2, quantile, 0.025)),
        upr=exp(apply(ee$beta, 2, quantile, 0.975))
)
```

```
##      median       lwr       upr
## 1 3.0675127 2.5289920 3.8755434
## 2 1.0111934 0.8307126 1.2266709
## 3 0.3544715 0.2847882 0.4354173
```

We should be able to apply importance sampling over this to make better inference but I'm not going to try it here...

3 Mixed model approach

We can also use the mixed model trick to model multivariate binary outcomes. Then, our model becomes

$$\text{logit}\,\Pr() \tag{4}$$

References

O'brien, S. M. and D. B. Dunson (2004). Bayesian multivariate logistic regression. *Biometrics 60*(3), 739–746.