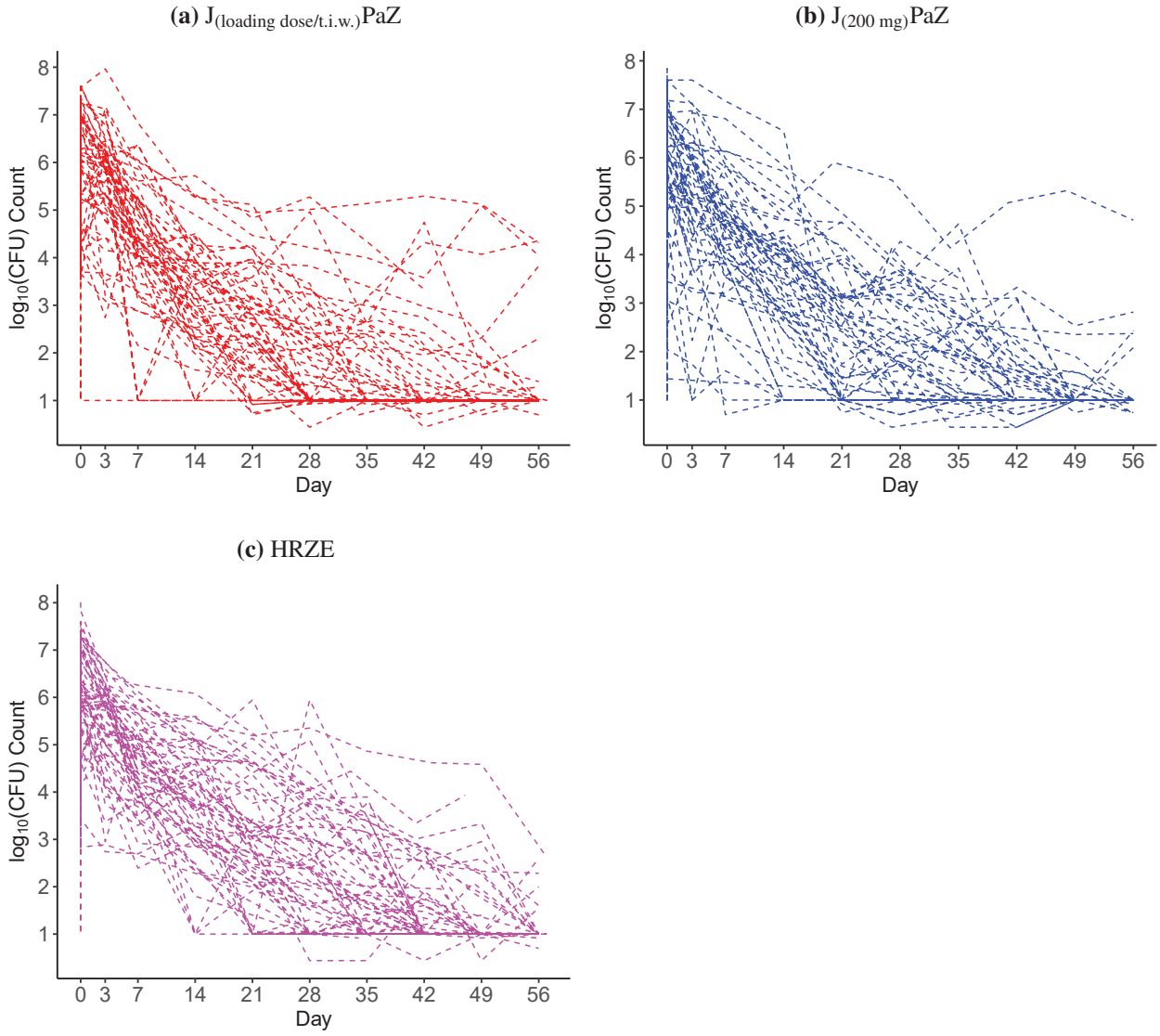
NONLINEAR MIXED-EFFECTS MODELING OF LONGITUDINAL COUNT DATA: BAYESIAN INFERENCE ABOUT MEDIAN COUNTS BASED ON THE MARGINAL ZERO-INFLATED DISCRETE WEIBULL DISTRIBUTION

Divan Aristo Burger and Emmanuel Lesaffre

SUPPLEMENTARY MATERIAL

Figure 1: TB dataset: observed $\log_{10}(\text{CFU})$ counts over time



CFU: Colony-forming unit. CFU counts of zero are displayed on the log-10 scale as 1.

SUMMARY STATISTICS

Table 1: Summary statistics of observed CFU count over time

Treatment	Day	n	Mean	SD	CV	Minimum	Median	Maximum	Zeros (%)
$J_{(\text{loading dose})}$PaZ	Day 0	162	5728792	9343449	163	0	1441000	40000000	1.9
	Day 3	55	3315936	12627118	381	0	679250	92400000	1.8
	Day 7	52	343078	1038131	303	0	22850	6765000	7.7
	Day 14	48	28087	87399	311	0	1788	525250	6.3
	Day 21	48	8768	24204	276	0	568	130000	6.3
	Day 28	48	8776	32969	376	0	73	188100	37.5
	Day 35	41	4257	21558	506	0	0	136950	53.7
	Day 42	49	5737	29068	507	0	0	196625	61.2
	Day 49	43	6256	27312	437	0	0	132000	72.1
	Day 56	48	1278	4682	366	0	0	22667	81.3
$J_{(200\ \text{mg})}$PaZ	Day 0	157	5127638	10585053	206	0	770000	68200000	1.9
	Day 3	52	2024968	5999341	296	0	176700	40000000	3.8
	Day 7	51	574902	2226334	387	0	42000	14750000	2.0
	Day 14	51	94971	499563	526	0	4015	3550000	9.8
	Day 21	47	22015	116787	530	0	270	800000	29.8
	Day 28	42	9833	53110	540	0	72	345000	23.8
	Day 35	41	1768	6979	395	0	15	42900	39.0
	Day 42	44	2821	17849	633	0	0	118500	54.5
	Day 49	46	4583	30960	676	0	0	210000	71.7
	Day 56	45	1174	7674	654	0	0	51500	77.8
HRZE	Day 0	167	6618662	12675684	192	0	1595000	102300000	0.6
	Day 3	55	778651	1184944	152	550	212500	5637500	0.0
	Day 7	55	193311	356931	185	248	45000	1806750	0.0
	Day 14	53	63521	179552	283	0	6600	1210000	5.7
	Day 21	51	34406	126824	369	0	1172	880000	9.8
	Day 28	52	26171	126473	483	0	312	885000	23.1
	Day 35	48	2816	11172	397	0	55	73150	33.3
	Day 42	50	997	5861	588	0	1	41525	50.0
	Day 49	46	1118	5781	517	0	0	38500	65.2
	Day 56	45	27	89	335	0	0	413	82.2

n = Number of CFU counts. CFU: Colony-forming unit. CV: Coefficient of variation. SD: Standard deviation.

REGRESSION CURVE

Figure 2: Example plot of regression curve for $\log_{10}(\text{CFU})$ count over time

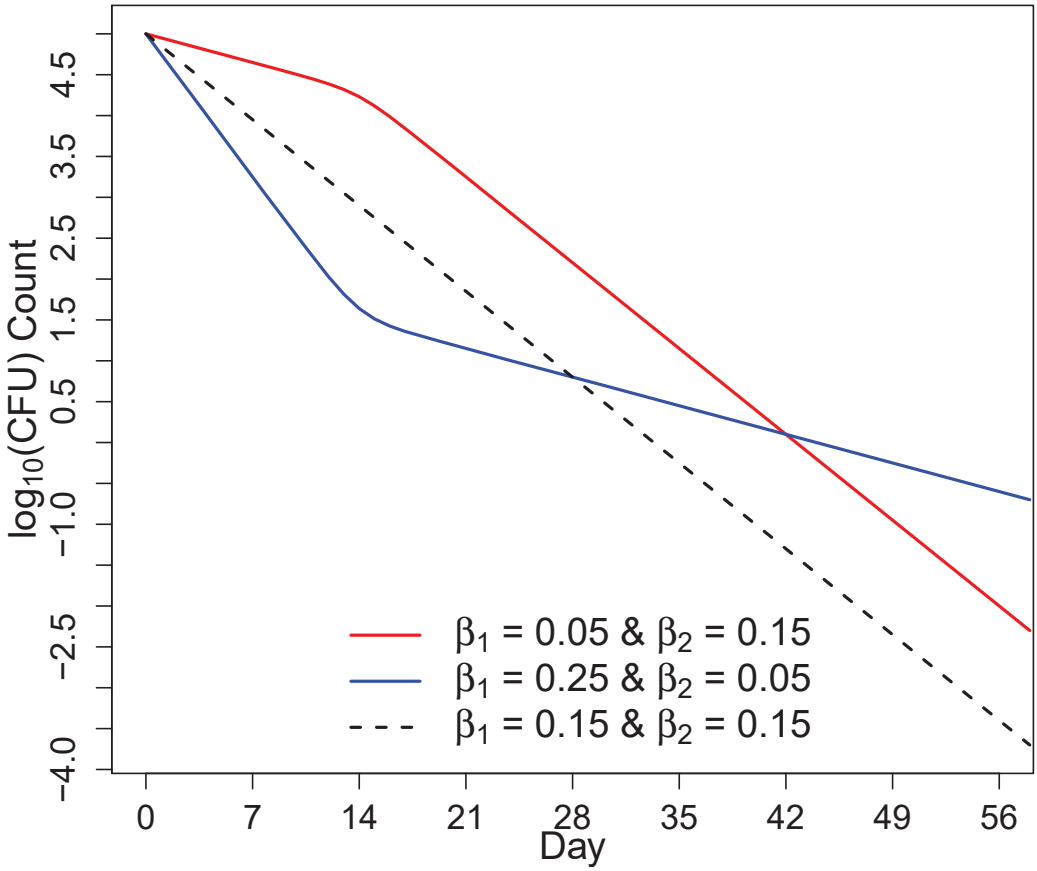
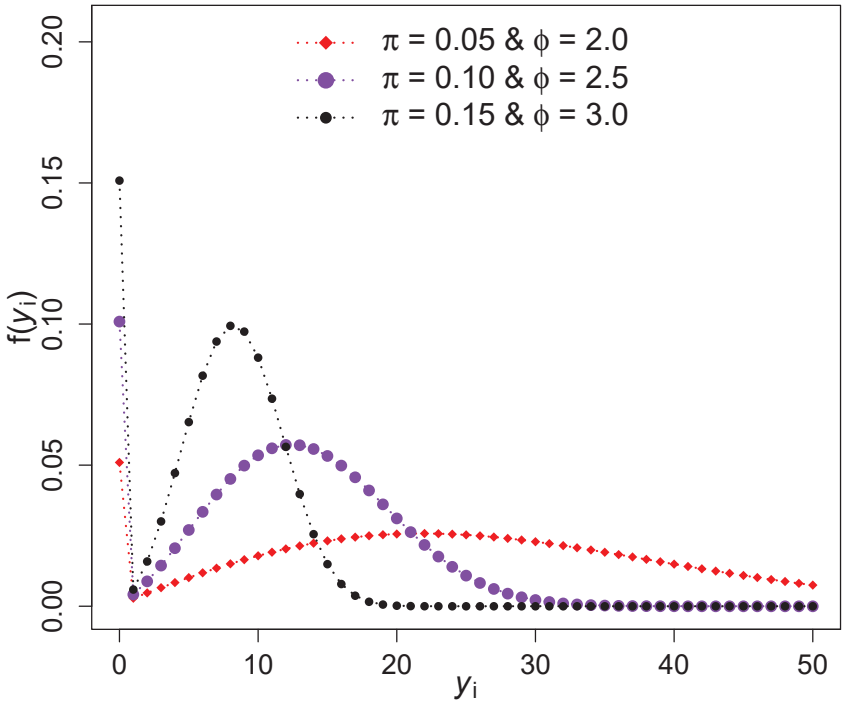
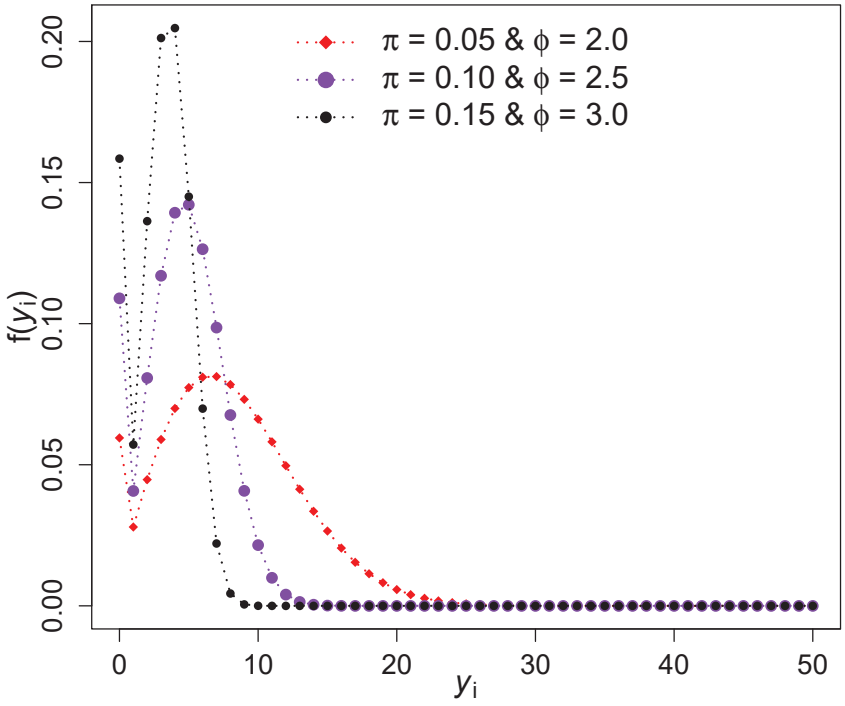


Figure 3: Probability mass function of zero-inflated discrete Weibull distribution

(a) $q = 0.999$



(b) $q = 0.99$



BAYESIAN MODEL SPECIFICATION

The Bayesian estimation procedures proposed in this manuscript are based on vague prior distributions as follows:

Multivariate normal prior distributions are specified for $\boldsymbol{\delta}_j$, namely:

$$\boldsymbol{\delta}_j \sim \text{ Normal}\left(\mathbf{0}, 10^4 \times \boldsymbol{I}\right)$$

where $\mathbf{0}$ and $\boldsymbol{I}$ respectively denote the corresponding vector of zeros and identity matrix.

The matrix-generalized half-t (MGH-t) prior distribution for the random effects covariance matrix is specified instead of the widely used Wishart prior distribution, since the latter often results in poor confidence interval coverage, especially when the variance components are small.[47] The MGH-t prior distribution for $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1}$ is hierarchically specified by Wishart and gamma distributions as[28]:

$$\begin{aligned} \boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1} | \boldsymbol{\Omega}_j &\sim \text{Wishart}\left(v+2, 2v\boldsymbol{\Omega}_j\right) \\ \omega_{jz} &\sim \text{Gamma}\left(0.5, 1/A^2\right) \end{aligned}$$

where $\boldsymbol{\Omega}_j$ denote diagonal matrices with ω_{jz} on the diagonal ($z = 1,2,3$). Here, $v+2$ and $2v\boldsymbol{\Omega}_j$ are respectively the degrees of freedom and inverse scale matrices of the Wishart distribution. The conditional density functions of $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}$ and ω_{jz} are written as:

$$\begin{aligned} P\left(\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1} | \boldsymbol{\Omega}_j\right) &\propto \left|\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}\right|^{\frac{v}{2}-1} \exp\left[-v \cdot \text{tr}\left(\boldsymbol{\Omega}_j \boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1}\right)\right]; \ \boldsymbol{\Omega}_j = \text{diag}\left(\omega_{j1}, \omega_{j2}, \omega_{j3}\right) \\ P\left(\omega_{jz}\right) &\propto \omega_{jz}^{-\frac{1}{2}} \exp\left(-\frac{1}{A^2}\omega_{jz}\right) \end{aligned}$$

From the law of total probability, the set of nuisance parameters $\boldsymbol{\Omega}_j$ integrated out results in the MGH-t prior distribution, namely:

$$f\left(\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}\right) \propto \left|\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}\right|^{-\frac{v+6}{2}} \prod_{z=1}^{3} \left[v\left(\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1}\right)_{zz} + 1/A^2\right]^{-\frac{v+3}{2}}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j} > \mathbf{0}$, and $\left(\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1}\right)_{zz}$ is the z^{th} diagonal entry of $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1}$. This mixture representation results in the half-t prior distribution, namely half-t (v, A), for the standard deviation terms in $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}$, and the uniform prior distribution, namely $U(-1,1)$, for the correlation terms in $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}$. The corresponding quantities are set to $A = 50$ and $v = 2$. Therefore, this specification results in a weakly informative (heavy-tailed) prior for the standard deviation terms.

The parameters κ_j and γ_j are assumed to follow uniform prior distributions, namely:

$$\begin{aligned}\kappa_j &\sim \text{Uniform}\left(L_\kappa, U_\kappa\right)\\ \gamma_j &\sim \text{Uniform}\left(L_\gamma, U_\gamma\right)\end{aligned}$$

where L_κ, U_κ, L_γ, and U_γ are the pre-specified lower and upper bounds for κ_j and γ_j, respectively.

The dispersion & shape parameters ϕ_j and zero-inflation probabilities π_{jk} are assumed to follow vague gamma and uniform prior distributions, namely:

$$\begin{aligned}\phi_j &\sim \text{Gamma}\left(0.5, 0.5\right)\\ \pi_{jk} &\sim \text{Uniform}\left(0, U_\pi\right)\end{aligned} \tag{8}$$

where U_π is the upper bound for π_{jk}. Here, ϕ_j's prior variance is 2. Assuming the gamma prior distribution in Equation (8) is reasonable if the data are believed to be overdispersed (i.e., small values of ϕ_j).

The resulting joint posterior distribution of the model parameters is written as:

$$\begin{aligned}&P\left(\boldsymbol{\delta}_{ij}, \boldsymbol{\delta}_j, \boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}, \omega_{jz}, \kappa_j, \gamma_j, \phi_j, \pi_{jk}, j = 1,\ldots,J, i = 1,\ldots,N_j, k = 1,\ldots,T | \boldsymbol{y}\right)\\ &\propto \left(\prod_{j=1}^{J}\prod_{i=1}^{N_j}\prod_{k=1}^{T} f\left(y_{ijk}|\lambda_{ijk}, \phi_j, \pi_{jk}\right)\right)\left(\prod_{j=1}^{J}\prod_{i=1}^{N_j} P\left(\boldsymbol{\delta}_{ij}|\boldsymbol{\delta}_j, \boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}\right)\right)\left(\prod_{j=1}^{J} P\left(\boldsymbol{\delta}_j\right) P\left(\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}^{-1}|\boldsymbol{\Omega}_j\right)\right)\times\\ &\left(\prod_{j=1}^{J}\prod_{z=1}^{3} P\left(\omega_{jz}\right)\right)\left(\prod_{j=1}^{J} P\left(\kappa_j\right) P\left(\gamma_j\right) P\left(\phi_j\right)\right)\left(\prod_{j=1}^{J}\prod_{k=1}^{T} P\left(\pi_{jk}\right)\right)\end{aligned}$$

where $\boldsymbol{y}$ is the vector containing y_{ijk} for all $j = 1,\ldots,J$, $i = 1,\ldots,N_j$, and $k = 1,\ldots,T$. The MCMC Gibbs sampling algorithm is used to draw samples from the joint posterior distribution of the model parameters.[48] Software such as `JAGS`[29] can be employed to carry out the Gibbs sampling procedure.

PROGRAMMING CODE: MZINB MODEL

```
library(Rfast)
library(runjags)
library(parallel)
library(coda)
setwd('C:/Program Files/R/R-3.5.1/bin')
NCORES <- detectCores() - 1 ###7 Cores

DATA1 <- read.csv('C:/Progra~1/R/R-3.5.1/bin/FIVE_CFU_O_MZINB_DATA.csv')
attach(DATA1) #See below.
DATA2 <- read.csv('C:/Progra~1/R/R-3.5.1/bin/FIVE_CFU_O_MZINB_PATKVT.csv') #Dataset containing the patient IDs the treatment groups to which the patients were assigned.
attach(DATA2)

###Notes:###
#Initially, we wrote our program to accommodate multiple longitudinal outcomes. For the current manuscript, only one longitudinal outcome was used. Hence, JVINDEX = 1.
#NSUBJID: Patient ID.
#KVTRTN: Treatment group.
#TIME: Observed time.
#PROTTIMC: Protocol timepoint.
#SUMCFU: Sum of the CFU counts (all plates).
#OFFSET: Offset constant.
#PATKVT: Patient ID.

DATA <- list(JVINDEX = JVINDEX, NSUBJID = NSUBJID, KVTRTN = KVTRTN, TIME = TIME, PROTTIMC = PROTTIMC, SUMCFU = SUMCFU, OFFSET = OFFSET, PATKVT = PATKVT, ZEROVEC = rep(0, 3), IDENMAT = diag(rep(0.0001, 3)))

BAYESMODEL <- '
data {
    C <- 10000
    for (i in 1:1792) {
        ONES[i] <- 1
    }
}
model {
    for (i in 1:1792) {
        LIKE[i] <- (PI[JVINDEX[i], KVTRTN[i], PROTTIMC[i]]*equals(SUMCFU[i], 0) + (1 - PI[JVINDEX[i], KVTRTN[i], PROTTIMC[i]])*(exp(loggam(SUMCFU[i] + PHI[JVINDEX[i], KVTRTN[i]]) - loggam(SUMCFU[i] + 1) - loggam(PHI[JVINDEX[i], KVTRTN[i]]))*(PHI[JVINDEX[i], KVTRTN[i]]/(PHI[JVINDEX[i], KVTRTN[i]] + LAMBDA[i]/(1 - PI[JVINDEX[i], KVTRTN[i], PROTTIMC[i]])))^PHI[JVINDEX[i], KVTRTN[i]]*(LAMBDA[i]/(1 - PI[JVINDEX[i], KVTRTN[i], PROTTIMC[i]])/(PHI[JVINDEX[i], KVTRTN[i]] + LAMBDA[i]/(1 - PI[JVINDEX[i], KVTRTN[i], PROTTIMC[i]])))^SUMCFU[i]))/C
        ONES[i] ~ dbern(LIKE[i])
        LINK[i] <- SALPHA[JVINDEX[i], NSUBJID[i]] - (SBETA2[JVINDEX[i], NSUBJID[i]] + SBETA1[JVINDEX[i], NSUBJID[i]])/2*TIME[i]/7 - (SBETA2[JVINDEX[i], NSUBJID[i]] - SBETA1[JVINDEX[i], NSUBJID[i]])/2*MGAMMA[JVINDEX[i], KVTRTN[i]]*log((exp((TIME[i]/7 - MKAPPA[JVINDEX[i], KVTRTN[i]])/MGAMMA[JVINDEX[i], KVTRTN[i]]) + exp(-(TIME[i]/7 - MKAPPA[JVINDEX[i], KVTRTN[i]])/MGAMMA[JVINDEX[i], KVTRTN[i]]))/(exp(MKAPPA[JVINDEX[i], KVTRTN[i]]/MGAMMA[JVINDEX[i], KVTRTN[i]]) + exp(-MKAPPA[JVINDEX[i], KVTRTN[i]]/MGAMMA[JVINDEX[i], KVTRTN[i]])))
        log(LAMBDA[i]) <- LINK[i] + OFFSET[i]
    }
    for (i in 1:172) {
        SALPHA[1, i] <- SDELTA[i, 1]
        SBETA1[1, i] <- SDELTA[i, 2]
        SBETA2[1, i] <- SDELTA[i, 3]
        SDELTA[i, 1:3] ~ dmnorm(MDELTA[PATKVT[i], 1:3], SGINV[PATKVT[i], 1:3, 1:3])
    }
    for (i in 1:3) {
        MALPHA[1, i] <- MDELTA[i, 1]
        MBETA1[1, i] <- MDELTA[i, 2]
        MBETA2[1, i] <- MDELTA[i, 3]
        OMEGA[i, 1, 1] ~ dgamma(0.5, 0.0004)
        for (x in 2:3) {
            OMEGA[i, x, x] ~ dgamma(0.5, 0.0004)
```

```
            for (y in 1:(x - 1)) {
                OMEGA[i, x, y] <- 0
                OMEGA[i, y, x] <- OMEGA[i, x, y]
            }
        }
        SGINV[i, 1:3, 1:3] ~ dwish(2*2*OMEGA[i, 1:3, 1:3], 4)
        SIGMA[i, 1:3, 1:3] <- inverse(SGINV[i, 1:3, 1:3])
        ALPSIGSQ[i] <- SIGMA[i, 1, 1]
        BT1SIGSQ[i] <- SIGMA[i, 2, 2]
        BT2SIGSQ[i] <- SIGMA[i, 3, 3]
        ALPBT1SIGSQ[i] <- SIGMA[i, 1, 2]
        ALPBT2SIGSQ[i] <- SIGMA[i, 1, 3]
        BT1BT2SIGSQ[i] <- SIGMA[i, 2, 3]
        MDELTA[i, 1:3] ~ dmnorm(ZEROVEC[1:3], IDENMAT[1:3, 1:3])
        for (k in 1:1) {
            MKAPPA[k, i] ~ dunif(0.4285714286, 1.5714285714)
            MGAMMA[k, i] ~ dunif(0.05, 2)
            PHI[k, i] ~ dgamma(0.5, 0.5)
            for (j in 1:10) {
                PI[k, i, j] ~ dunif(0, 1)
            }
        }
    }
}'
OMEGA <- array(NA, dim = c(3, 3, 3))
for (k in 1:3) {
    OMEGA[k, 1, 1] <- 1
    for (i in 2:3) {
        OMEGA[k, i, i] <- 1
    }
}
ORIGPARMS <- c('SALPHA', 'SBETA1', 'SBETA2', 'MALPHA', 'MBETA1', 'MBETA2', 'MKAPPA',
'MGAMMA', 'PHI', 'PI', 'ALPSIGSQ', 'BT1SIGSQ', 'BT2SIGSQ', 'ALPBT1SIGSQ',
'ALPBT2SIGSQ', 'BT1BT2SIGSQ')
ORIGPARMS
NEWPARAMS <- c('LAMBDA')
NEWPARAMS
ALLPARAMS <- c(ORIGPARMS, NEWPARAMS)
ALLPARAMS
SDELTA =
as.matrix(read.csv('C:/Progra~1/R/R-3.5.1/bin/FIVE_CFU_O_MZINB_Initial_SDELTA.csv'))
#Dataset containing initial values.
MDELTA = t(matrix(c(colmeans(SDELTA), colmeans(SDELTA), colmeans(SDELTA)), ncol = 3))
INITIAL <- replicate(NCORES, list(list(PHI = array(rep(0.1, 3), dim = c(1, 3)), OMEGA =
OMEGA, SDELTA = SDELTA, .RNG.name = 'base::Mersenne-Twister', .RNG.seed = sample.int(n
= 100000, size = 1))))
TIME <- proc.time()
SAMPLE <- run.jags(model = BAYESMODEL, data = DATA, inits = INITIAL, monitor =
ALLPARAMS, n.chains = NCORES, burnin = 15000, thin = 450, sample = 150, summarise =
FALSE, method = 'parallel', modules = 'glm', factories = 'bugs::MNormal sampler off')
proc.time() - TIME
READ1 <- as.matrix(as.mcmc(SAMPLE, vars = c(ORIGPARMS)))
READ2 <- as.matrix(as.mcmc(SAMPLE, vars = c(NEWPARAMS)))
write.csv(READ1, 'FIVE_CFU_O_MZINB_CODA1.csv')
write.csv(t(READ2), 'FIVE_CFU_O_MZINB_CODA2.csv')
SUMMARY <- summary(SAMPLE, confidence = c(0.95))
write.csv(SUMMARY, 'FIVE_CFU_O_MZINB_LOG.csv')
pdf('FIVE_CFU_O_MZINB_DIAGN.pdf')
plot(SAMPLE, vars = c(ORIGPARMS), plot.type = c('trace'), new.window = FALSE)
plot(SAMPLE, vars = c(ORIGPARMS), plot.type = c('autocorr'), new.window = FALSE)
dev.off()
extract(SAMPLE, what = 'samplers')
```

PROGRAMMING CODE: MZIDW MODEL

```
library(Rfast)
library(runjags)
library(parallel)
library(coda)
setwd('C:/Program Files/R/R-3.5.1/bin')
NCORES <- detectCores() - 1 ###7 Cores

DATA1 <- read.csv('C:/Progra~1/R/R-3.5.1/bin/FIVE_CFU_O_MZIDW_DATA.csv')
attach(DATA1) #See below.
DATA2 <- read.csv('C:/Progra~1/R/R-3.5.1/bin/FIVE_CFU_O_MZIDW_PATKVT.csv') #Dataset containing the patient IDs the treatment groups to which the patients were assigned.
attach(DATA2)

###Notes:###
#Initially, we wrote our program to accommodate multiple longitudinal outcomes. For the current manuscript, only one longitudinal outcome was used. Hence, JVINDEX = 1.
#NSUBJID: Patient ID.
#KVTRTN: Treatment group.
#TIME: Observed time.
#PROTTIMC: Protocol timepoint.
#SUMCFU: Sum of the CFU counts (all plates).
#OFFSET: Offset constant.
#PATKVT: Patient ID.

DATA <- list(JVINDEX = JVINDEX, NSUBJID = NSUBJID, KVTRTN = KVTRTN, TIME = TIME, PROTTIMC = PROTTIMC, SUMCFU = SUMCFU, OFFSET = OFFSET, PATKVT = PATKVT, ZEROVEC = rep(0, 3), IDENMAT = diag(rep(0.0001, 3)))

BAYESMODEL <- '
data {
    C <- 10000
    for (i in 1:1792) {
        ONES[i] <- 1
    }
}
model {
    for (i in 1:1792) {
        LIKE[i] <- (PI[JVINDEX[i], KVTRTN[i], PROTTIMC[i]]*equals(SUMCFU[i], 0) + (1 - PI[JVINDEX[i], KVTRTN[i],
        PROTTIMC[i]])*(exp(log(0.5)*(SUMCFU[i]/MU[i])^PHI[JVINDEX[i], KVTRTN[i]]) - exp(log(0.5)*((SUMCFU[i] + 1)/MU[i])^PHI[JVINDEX[i], KVTRTN[i]])))/C
        ONES[i] ~ dbern(LIKE[i])
        MU[i] <- LAMBDA[i]*(log(0.5/(1 - PI[JVINDEX[i], KVTRTN[i],
        PROTTIMC[i]]))/log(0.5))^(-1/PHI[JVINDEX[i], KVTRTN[i]])
        LINK[i] <- SALPHA[JVINDEX[i], NSUBJID[i]] - (SBETA2[JVINDEX[i], NSUBJID[i]] + SBETA1[JVINDEX[i], NSUBJID[i]])/2*TIME[i]/7 - (SBETA2[JVINDEX[i], NSUBJID[i]] - SBETA1[JVINDEX[i], NSUBJID[i]])/2*MGAMMA[JVINDEX[i],
        KVTRTN[i]]*log((exp((TIME[i]/7 - MKAPPA[JVINDEX[i],
        KVTRTN[i]])/MGAMMA[JVINDEX[i], KVTRTN[i]]) + exp(-(TIME[i]/7 - MKAPPA[JVINDEX[i], KVTRTN[i]])/MGAMMA[JVINDEX[i],
        KVTRTN[i]]))/(exp(MKAPPA[JVINDEX[i], KVTRTN[i]]/MGAMMA[JVINDEX[i], KVTRTN[i]])
        + exp(-MKAPPA[JVINDEX[i], KVTRTN[i]]/MGAMMA[JVINDEX[i], KVTRTN[i]])))
        log(LAMBDA[i]) <- LINK[i] + OFFSET[i]
    }
    for (i in 1:172) {
        SALPHA[1, i] <- SDELTA[i, 1]
        SBETA1[1, i] <- SDELTA[i, 2]
        SBETA2[1, i] <- SDELTA[i, 3]
        SDELTA[i, 1:3] ~ dmnorm(MDELTA[PATKVT[i], 1:3], SGINV[PATKVT[i], 1:3, 1:3])
    }
    for (i in 1:3) {
        MALPHA[1, i] <- MDELTA[i, 1]
        MBETA1[1, i] <- MDELTA[i, 2]
        MBETA2[1, i] <- MDELTA[i, 3]
        OMEGA[i, 1, 1] ~ dgamma(0.5, 0.0004)
        for (x in 2:3) {
            OMEGA[i, x, x] ~ dgamma(0.5, 0.0004)
            for (y in 1:(x - 1)) {
                OMEGA[i, x, y] <- 0
```

```
                    OMEGA[i, y, x] <- OMEGA[i, x, y]
                }
            }
            SGINV[i, 1:3, 1:3] ~ dwish(2*2*OMEGA[i, 1:3, 1:3], 4)
            SIGMA[i, 1:3, 1:3] <- inverse(SGINV[i, 1:3, 1:3])
            ALPSIGSQ[i] <- SIGMA[i, 1, 1]
            BT1SIGSQ[i] <- SIGMA[i, 2, 2]
            BT2SIGSQ[i] <- SIGMA[i, 3, 3]
            ALPBT1SIGSQ[i] <- SIGMA[i, 1, 2]
            ALPBT2SIGSQ[i] <- SIGMA[i, 1, 3]
            BT1BT2SIGSQ[i] <- SIGMA[i, 2, 3]
            MDELTA[i, 1:3] ~ dmnorm(ZEROVEC[1:3], IDENMAT[1:3, 1:3])
            for (k in 1:1) {
                MKAPPA[k, i] ~ dunif(0.4285714286, 1.5714285714)
                MGAMMA[k, i] ~ dunif(0.05, 2)
                PHI[k, i] ~ dgamma(0.5, 0.5)
                for (j in 1:10) {
                    PI[k, i, j] ~ dunif(0, 0.5)
                }
            }
        }
    }'
    OMEGA <- array(NA, dim = c(3, 3, 3))
    for (k in 1:3) {
        OMEGA[k, 1, 1] <- 1
        for (i in 2:3) {
            OMEGA[k, i, i] <- 1
        }
    }
    ORIGPARMS <- c('SALPHA', 'SBETA1', 'SBETA2', 'MALPHA', 'MBETA1', 'MBETA2', 'MKAPPA',
    'MGAMMA', 'PHI', 'PI', 'ALPSIGSQ', 'BT1SIGSQ', 'BT2SIGSQ', 'ALPBT1SIGSQ',
    'ALPBT2SIGSQ', 'BT1BT2SIGSQ')
    ORIGPARMS
    NEWPARAMS <- c('LAMBDA')
    NEWPARAMS
    ALLPARAMS <- c(ORIGPARMS, NEWPARAMS)
    ALLPARAMS
    SDELTA =
    as.matrix(read.csv('C:/Progra~1/R/R-3.5.1/bin/FIVE_CFU_O_MZIDW_Initial_SDELTA.csv'))
    #Dataset containing initial values.
    MDELTA = t(matrix(c(colmeans(SDELTA), colmeans(SDELTA), colmeans(SDELTA)), ncol = 3))
    INITIAL <- replicate(NCORES, list(list(PHI = array(rep(0.1, 3), dim = c(1, 3)), OMEGA =
    OMEGA, SDELTA = SDELTA, .RNG.name = 'base::Mersenne-Twister', .RNG.seed = sample.int(n
    = 100000, size = 1))))
    TIME <- proc.time()
    SAMPLE <- run.jags(model = BAYESMODEL, data = DATA, inits = INITIAL, monitor =
    ALLPARAMS, n.chains = NCORES, burnin = 15000, thin = 450, sample = 150, summarise =
    FALSE, method = 'parallel', modules = 'glm', factories = 'bugs::MNormal sampler off')
    proc.time() - TIME
    READ1 <- as.matrix(as.mcmc(SAMPLE, vars = c(ORIGPARMS)))
    READ2 <- as.matrix(as.mcmc(SAMPLE, vars = c(NEWPARAMS)))
    write.csv(READ1, 'FIVE_CFU_O_MZIDW_CODA1.csv')
    write.csv(t(READ2), 'FIVE_CFU_O_MZIDW_CODA2.csv')
    SUMMARY <- summary(SAMPLE, confidence = c(0.95))
    write.csv(SUMMARY, 'FIVE_CFU_O_MZIDW_LOG.csv')
    pdf('FIVE_CFU_O_MZIDW_DIAGN.pdf')
    plot(SAMPLE, vars = c(ORIGPARMS), plot.type = c('trace'), new.window = FALSE)
    plot(SAMPLE, vars = c(ORIGPARMS), plot.type = c('autocorr'), new.window = FALSE)
    dev.off()
    extract(SAMPLE, what = 'samplers')
```

COMPOUND LAPLACE-METROPOLIS MARGINAL LIKELIHOOD

The Laplace-Metropolis approximation of $\log(f[\boldsymbol{y}|M])$ (that is, CLMML) under Model M can be written as:

$$\log(f[\boldsymbol{y}|M]) = \frac{1}{2}\log(2\pi)\,pJ + \frac{1}{2}\log\left|R_{(\boldsymbol{\delta}_j,\kappa_j,\gamma_j,\phi_j,\pi_{jk},j=1,\ldots,J,k=1,\ldots,T)}\right| + s_{(\boldsymbol{\delta}_j,\kappa_j,\gamma_j,\phi_j,\pi_{jk},j=1,\ldots,J,k=1,\ldots,T)} + \sum_{j=1}^{J}\sum_{i=1}^{N_j}\left(\log\left[\int\left(\prod_{k=1}^{T} f\left(y_{ijk}|\boldsymbol{\delta}_{ij},\hat{\kappa}_j,\hat{\gamma}_j,\hat{\phi}_j,\hat{\pi}_{jk}\right)\right)P\left(\boldsymbol{\delta}_{ij}|\hat{\boldsymbol{\delta}}_j,\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\delta}_j}\right)\mathrm{d}\boldsymbol{\delta}_{ij}\right]\right) + \sum_{j=1}^{J}\log\left[P\left(\hat{\boldsymbol{\delta}}_j\right)P\left(\hat{\kappa}_j\right)P\left(\hat{\gamma}_j\right)P\left(\hat{\phi}_j\right)P\left(\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\delta}_j}\right)\right] + \sum_{j=1}^{J}\sum_{k=1}^{T}\log\left[P\left(\hat{\pi}_{jk}\right)\right]$$

where p is the number of parameters among $\boldsymbol{\delta}_j$, κ_j, γ_j, ϕ_j, $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}$, π_{jk} of treatment group j. Here, $\hat{\boldsymbol{\delta}}_j$, $\hat{\kappa}_j$, $\hat{\gamma}_j$, $\hat{\phi}_j$, $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{\delta}_j}$, $\hat{\pi}_{jk}$ are respectively the mean of the posterior distribution of $\boldsymbol{\delta}_j$, κ_j, γ_j, ϕ_j, $\boldsymbol{\Sigma}_{\boldsymbol{\delta}_j}$, π_{jk}. $\left|R_{(\boldsymbol{\delta}_j,\kappa_j,\gamma_j,\phi_j,\pi_{jk},j=1,\ldots,J,k=1,\ldots,T)}\right|$ and $s_{(\boldsymbol{\delta}_j,\kappa_j,\gamma_j,\phi_j,\pi_{jk},j=1,\ldots,J,k=1,\ldots,T)}$ respectively denote the determinant of the correlation matrix and the sum of the logarithm of the standard deviations of the posterior distributions of $\boldsymbol{\delta}_j$, κ_j, γ_j, ϕ_j, π_{jk}. The model with the largest CLMML is favored.

Table 2: Simulation study: performance of regression models MZINB and MZIDW (50 patients)

		MZINB[a]				MZIDW[b]			
Parameter[c]	**Value**	**Bias**	**RMSE**	**Coverage[d]**	**Length[e]**	**Bias**	**RMSE**	**Coverage[d]**	**Length[e]**
α_1	12.65	0.0107	0.1984	93.0	0.7690	0.0230	0.2045	95.0	0.8413
β_{11}	4.20	0.0493	0.4144	97.0	1.6706	0.0411	0.4832	95.0	1.8598
β_{21}	1.80	0.0136	0.2294	96.0	0.8880	0.0280	0.2116	97.0	0.9124
$\sigma^2_{\alpha_1}$	0.95	-0.0089	0.2952	97.0	1.2923	-0.0263	0.3447	96.0	1.4548
$\sigma^2_{\beta_{11}}$	0.85	-0.1258	0.7416	100.0	2.9449	-0.1335	0.6402	100.0	3.1990
$\sigma^2_{\beta_{21}}$	0.80	0.0579	0.3126	97.0	1.2932	0.0517	0.2686	99.0	1.3067
$\sigma_{\alpha_1\beta_{11}}$	0.25	-0.1490	0.2660	100.0	1.4378	-0.1680	0.2522	100.0	1.5842
$\sigma_{\alpha_1\beta_{21}}$	0.35	-0.0252	0.1552	98.0	0.8703	-0.0161	0.2074	98.0	0.9323
$\sigma_{\beta_{11}\beta_{21}}$	-0.15	0.0400	0.2506	100.0	1.3533	0.0365	0.2020	100.0	1.4497
ϕ_1	0.70	0.0349	0.0800	94.0	0.2846	0.0140	0.0445	96.0	0.1716
π_{11}	0.01	0.0071	0.0120	96.0	0.0388	0.0057	0.0102	96.0	0.0370
π_{12}	0.01	0.0422	0.0498	97.0	0.1352	0.0374	0.0439	99.0	0.1248

HPD: Highest posterior density. MZIDW: Marginal zero-inflated discrete Weibull. MZINB: Marginal zero-inflated negative binomial. RMSE: Root mean square error. [a]The log-link function is used to describe the biphasic relationship between the *mean* count and time. [b]The log-link function is used to describe the biphasic relationship between the *median* count and time. [c]For the prevention of numerical overflow, the models were fitted with time expressed in weeks. [d]95% HPD interval coverage (%). [e]95% HPD interval average length.

Table 3: Simulation study: robustness of regression models MZINB and MZIDW (15 patients)

			MZINB[a]				MZIDW[b]			
ψ_{α_1}	Parameter[c]	Value	Bias	RMSE	Coverage[d]	Length[e]	Bias	RMSE	Coverage[d]	Length[e]
-2	α_1	12.65	-0.0004	0.4823	95.6	2.0276	-0.0334	0.4896	95.8	2.1268
	β_{11}	3.95	-0.0306	0.6748	97.8	3.0098	0.0037	0.7497	96.2	3.3310
	β_{21}	1.35	0.0512	0.3521	96.2	1.4967	0.0184	0.3375	96.6	1.4838
-3	α_1	12.65	0.0033	0.5726	95.4	2.5633	0.0101	0.5838	95.0	2.6299
	β_{11}	3.95	-0.0587	0.6913	96.4	3.0861	0.0197	0.7044	97.4	3.3905
	β_{21}	1.35	0.0559	0.3668	96.8	1.5293	0.0339	0.3504	96.0	1.5318
-4	α_1	12.65	-0.0420	0.7152	95.4	3.1330	-0.0051	0.7316	95.4	3.1963
	β_{11}	3.95	-0.0316	0.6887	97.4	3.1463	0.0521	0.7473	97.4	3.4657
	β_{21}	1.35	0.0410	0.3665	97.0	1.5859	0.0437	0.3455	98.6	1.5946

HPD: Highest posterior density. MZIDW: Marginal zero-inflated discrete Weibull. MZINB: Marginal zero-inflated negative binomial. RMSE: Root mean square error. [a]The log-link function is used to describe the biphasic relationship between the *mean* count and time. [b]The log-link function is used to describe the biphasic relationship between the *median* count and time. [c]For the prevention of numerical overflow, the models were fitted with time expressed in weeks. [d]95% HPD interval coverage (%). [e]95% HPD interval average length.