Zero-inflated negative binomial mixed model: An application to two microbial organisms important in Esophagitis

Online material

*Microbiome identification*

DNA from all samples was extracted using Qiagen DNAeasy Extraction Kits for blood and tissue according to manufacturer’s specifications (Qiagen, Valencia, CA). DNA was amplified in triplicate with barcoded PCR primers that include adaptors for the Roche 454 sequencing platform (Hamady et al. 2008) . Negative PCR controls were performed for each barcode, and PCR was repeated for any sample where the negative control was positive. Amplicons were pooled after normalization of DNA concentration (Harris et al. 2010), and sequenced using the Roche 454 FLX platform according to manufacturer’s specifications (Roche, Branford, CT). Sequence data were assigned to samples of origin using bar code sequences added during PCR, and screened for basic quality defects (short sequences < 200 nucleotides in length, >1 sequence ambiguity, best read with quality ≥20 over a 10-nucleotide moving window) by the program BARTAB (Frank 2009). Non-bacterial sequences were removed from datasets by requiring a close match with a bacterial rRNA secondary structure model within Infernal (Nawrocki 2009). Sequences identified as potential chimeras by ChimeraSlayer (Haas et al. 2011) were also removed from datasets. The Ribosomal Database Project Classifier software was used to make taxonomic assignments (Wang et al. 2007). Taxonomic information was used to construct sequence groups with identical taxonomic rank.

Hamady M, Walker JJ, Harris JK, Gold NJ, Knight R (2008) Error-correcting barcoded primers for pyrosequencing hundreds of samples in multiplex. Nat Methods 5: 235–237.

Harris JK, Sahl JW, Castoe TA, Wagner BD, Pollock DD, et al. (2010) Comparison of Normalization Methods for Construction of Large Multiplex

Amplicon Pools for Next-Generation Sequencing. Appl Environ Microbiol.

Frank DN (2009) BARCRAWL and BARTAB: software tools for the design and implementation of barcoded primers for highly multiplexed DNA sequencing. BMC Bioinformatics 10: 362.

Nawrocki EP, Kolbe DL, Eddy SR (2009) Infernal 1.0: inference of RNA alignments. Bioinformatics 25: 1335–1337.

Haas BJ, Gevers D, Earl AM, Feldgarden M, Ward DV, et al. (2011) Chimeric 16S rRNA sequence formation and detection in Sanger and 454-pyrosequenced PCR amplicons. Genome Res 21: 494–504.

Wang Q, Garrity GM, Tiedje JM, Cole JR (2007) Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. Appl Environ Microbiol 73: 5261–5267.

*SAS code*

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* \*/

/\* \*/

/\* HAEMOPHILUS \*/

/\* \*/

/\* \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Run genmod without random effects to obtain initial parameter estimates for NLMIXED\*/

**proc** **genmod** data=in.EOE\_2015\_06\_25;

 where last='Haemophilus';

model r=string nasal oral ppi steroid eoe gerd eos\_15 eoe\*ppi eoe\*string eoe\*nasal eoe\*oral denver/dist=zinb offset=ltotal;

**run**;

/\*ZINB\*/

/\* independent random effects \*/

**proc** **sort** data=in.EOE\_2015\_06\_25;

 by Subject;

**run**;

**proc** **nlmixed** data=in.EOE\_2015\_06\_25;

 where last='Haemophilus';

 /\* Specify initial parameter estimates \*/

parms b0=-**3** b1=**0** b2=-**2** b3=**0.1** b4=**.5** b5=-**0.2** b6=**.2** b7=-**.6** b8=**.1** b9=-**.8** b10=**.4** b11=**.5** b12=**.4**

 c0=-**12** c1=**3** c4=**1** k=**1** su=**.2** sv=**2**;

 bounds su > **0**;

 vu=su\*su;

 vv=sv\*sv;

 /\* define linear function for count part of model \*/

eta = b0 + b1\*string + b2\*nasal + b3\*oral + b4\*ppi + b5\*steroid + b6\*EoE + b7\*GERD + b8\*eos\_15 + b9\*eoe\*ppi + b10\*eoe\*string + b11\*eoe\*nasal + b12\*eoe\*oral + ltotal + ui;

 lambda = exp(eta);

/\* define linear function for ZI part of model \*/

 eta\_p = c0 + c1\*GERD + c4\*ltotal + vi;

 pi = **1**/(**1**+exp(-eta\_p));

 /\* define ZINB log likelihood \*/

loglike=lgamma(r + **1**/k) -lgamma(r + **1**) -lgamma(**1**/k) + r\*log(k\*lambda) - (r + **1**/k)\*log(**1** + k\*lambda);

 if r=**0** then pdf= pi + (**1**-pi)\*exp(loglike);

 else pdf = (**1**-pi)\*exp(loglike);

 if pdf > **1e-8** then loglike2=log(pdf);

 else loglike2 = -**1e20**;

 model r ~ general (loglike2);

 random ui vi ~ normal ([**0**,**0**], [vu, **0**, vv]) subject=Subject;

 /\* estimate LSmeans and perform comparisons\*/

 estimate 'untrt EoE string' exp (b0 + b1 + b6 + b10 + **0.3**\*b8);

estimate 'ppi EoE string' exp (b0 + b1 + b4 + **.22**\*b5 + b6 + b9 + b10 + **0.3**\*b8);

 estimate 'normal string' exp(b0 + b1 + **0.3**\*b8 +**.44**\*b4 +**.22**\*b5);

 estimate 'untrt EoE biopsy' exp(b0 + b6 + **0.3**\*b8);

 estimate 'normal biopsy' exp(b0 + **0.3**\*b8 +**.44**\*b4 +**.22**\*b5);

 estimate 'untrt EoE oral' exp(b0 + b3 + b6 + b12 + **0.3**\*b8);

 estimate 'normal oral' exp(b0 + b3 + **0.3**\*b8 +**.44**\*b4 +**.22**\*b5);

 estimate 'untrt EoE nasal' exp(b0 + b2 + b6 + b11 + **0.3**\*b8);

 estimate 'normal nasal' exp(b0 + b2 + **0.3**\*b8 +**.44**\*b4 +**.22**\*b5);

estimate 'untrt EoE string2' exp ((**1**-(**1**/(**1**+exp(-c0 + c4))))\*(b0 + b1 + b6 + b10 + **0.3**\*b8));

estimate 'normal string2' exp((**1**-(**1**/(**1**+exp(-c0 + c4))))\*(b0 + b1 + **0.3**\*b8 +**.44**\*b4 +**.22**\*b5));

estimate 'untrt EoE biopsy2' exp ((**1**-(**1**/(**1**+exp(-c0 + c4))))\*(b0 + b6 + **0.3**\*b8));

estimate 'normal biopsy2' exp((**1**-(**1**/(**1**+exp(-c0 + c4))))\*(b0 + **0.3**\*b8 +**.44**\*b4 +**.22**\*b5));

 contrast 'eoe vs normal string0' b6 + b10 -**.44**\*b4 -**.22**\*b5;

 contrast 'eoe vs normal string1' b6 + b10;

 contrast 'eoe vs normal biopsy' b6;

 contrast 'eoe vs normal oral' b6 + b12;

 contrast 'eoe vs normal nasal' b6 + b11;

 ods output AdditionalEstimates=fig2;

**run**;

/\*NB\*/

**proc** **nlmixed** data=in.EOE\_2015\_06\_25;

 where last='Haemophilus';

parms b0=-**3** b1=**0** b2=-**2** b3=**0.2** b4=**.8** b5=-**0.4** b6=**.8** b7=-**.6** b8=**.1** b9=-**.8** b10=**.2** b11=**.1** b12=**.1**

 k=**1** su=**.2**;

 bounds su > **0**;

 vu=su\*su;

 /\* define linear function for count part of model \*/

eta = b0 + b1\*string + b2\*nasal + b3\*oral + b4\*ppi + b5\*steroid + b6\*EoE + b7\*GERD + b8\*eos\_15 + b9\*eoe\*ppi + b10\*eoe\*string + b11\*eoe\*nasal + b12\*eoe\*oral + ltotal + ui;

 lambda = exp(eta);

 /\* define ZINB log likelihood \*/

loglike=lgamma(r + **1**/k) -lgamma(r + **1**) -lgamma(**1**/k) + r\*log(k\*lambda) - (r + **1**/k)\*log(**1** + k\*lambda);

 model r ~ general (loglike);

 random ui ~ normal (**0**, vu) subject=Subject;

 predict lambda out=nb\_r ;

**run**;

/\* Correlated random effects \*/

**proc** **nlmixed** data=in.EOE\_2015\_06\_25;

 where last='Haemophilus';

parms b0=-**3** b1=**0** b2=-**2** b3=**0.1** b4=**.5** b5=-**0.2** b6=**.2** b7=-**.6** b8=**.1** b9=-**.8** b10=**.4** b11=**.5** b12=**.4**

 c0=-**4** k=**1** su=**.2** sv=**1.8** cov=**.1**;

 bounds su cov > **0**;

 vu=su\*su;

 vv=sv\*sv;

 /\* define linear function for count part of model \*/

eta = b0 + b1\*string + b2\*nasal + b3\*oral + b4\*ppi + b5\*steroid + b6\*EoE + b7\*GERD + b8\*eos\_15 + b9\*eoe\*ppi + b10\*eoe\*string + b11\*eoe\*nasal + b12\*eoe\*oral + ltotal + ui;

 lambda = exp(eta);

/\* define linear function for ZI part of model \*/

 eta\_p = c0 + vi;

 pi = **1**/(**1**+exp(-eta\_p));

 /\* define ZINB log likelihood \*/

loglike=lgamma(r + **1**/k) -lgamma(r + **1**) -lgamma(**1**/k) + r\*log(k\*lambda) - (r + **1**/k)\*log(**1** + k\*lambda);

 if r=**0** then pdf= pi + (**1**-pi)\*exp(loglike);

 else pdf = (**1**-pi)\*exp(loglike);

 if pdf > **1e-8** then loglike2=log(pdf);

 else loglike2 = -**1e20**;

 model r ~ general (loglike2);

random ui vi ~ normal ([**0**,**0**], [vu, cov, vv]) subject=Subject;

**run**;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* \*/

/\* \*/

/\* Fusobacterium \*/

/\* \*/

/\* \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Run genmod without random effects to obtain initial parameter estimates for NLMIXED\*/

**proc** **genmod** data=in.EOE\_2015\_06\_25;

 where last='Fusobacterium' and type ne 'B';

model r= eoe string nasal steroid denver denver\*steroid/type3 dist=zinb offset=ltotal;

 zeromodel eoe ppi denver ;

**run**;

/\*ZINB\*/

/\* independent random effects \*/

**proc** **sort** data=in.EOE\_2015\_06\_25;

 by Subject;

**run**;

**proc** **nlmixed** data=in.EOE\_2015\_06\_25;

 where last='Fusobacterium' and type in ('O' 'S' 'N');

 /\* Specify initial parameter estimates \*/

 parms b0=-**4** b1=**.1** b2=-**.2** b3=-**2** b4=-**1** b5=**.5** b6=**1**

 c0=**.3** c1=-**6** c2=-**22** c3=-**3** k=**1.2** su=**.2** sv=**.0001**;

 bounds su sv >= **0**;

 vu=su\*su;

 vv=sv\*sv;

 /\* define linear function for count part of model \*/

eta = b0 + b1\*eoe + b2\*string + b3\*nasal + b4\*steroid + b5\*denver + b6\*steroid\*denver + ltotal + ui;

 lambda = exp(eta);

/\* define linear function for ZI part of model \*/

 eta\_p = c0 + c1\*eoe + c2\*ppi + c3\*denver +vi;

 pi = **1**/(**1**+exp(-eta\_p));

 /\* define ZINB log likelihood \*/

loglike=lgamma(r + **1**/k) -lgamma(r + **1**) -lgamma(**1**/k) + r\*log(k\*lambda) - (r + **1**/k)\*log(**1** + k\*lambda);

 if r=**0** then pdf= pi + (**1**-pi)\*exp(loglike);

 else pdf = (**1**-pi)\*exp(loglike);

 if pdf > **1e-8** then loglike2=log(pdf);

 else loglike2 = -**1e20**;

 model r ~ general (loglike2);

 random ui vi ~ normal ([**0**,**0**], [vu, **0**, vv]) subject=Subject;

**run**;

/\*ZINB\*/

/\* remove ZI random effect \*/

**proc** **nlmixed** data=in.EOE\_2015\_06\_25;

 where last='Fusobacterium' and type in ('O' 'S' 'N');

 parms b0=-**4** b1=**.1** b2=-**.2** b3=-**2** b4=-**1** b5=**.5** b6=**1**

 c0=**.3** c1=-**6** c2=-**22** c3=-**3** k=**1.2** su=**.2** ;

 bounds su >= **0**;

 vu=su\*su;

 /\* define linear function for count part of model \*/

eta = b0 + b1\*eoe + b2\*string + b3\*nasal + b4\*steroid + b5\*denver + b6\*steroid\*denver + ltotal + ui;

 lambda = exp(eta);

/\* define linear function for ZI part of model \*/

 eta\_p = c0 + c1\*eoe + c2\*ppi + c3\*denver ;

 pi = **1**/(**1**+exp(-eta\_p));

 /\* define ZINB log likelihood \*/

loglike=lgamma(r + **1**/k) -lgamma(r + **1**) -lgamma(**1**/k) + r\*log(k\*lambda) - (r + **1**/k)\*log(**1** + k\*lambda);

 if r=**0** then pdf= pi + (**1**-pi)\*exp(loglike);

 else pdf = (**1**-pi)\*exp(loglike);

 if pdf > **1e-8** then loglike2=log(pdf);

 else loglike2 = -**1e20**;

 model r ~ general (loglike2);

 random ui ~ normal (**0**, vu) subject=Subject;

/\* estimate LSmeans and perform comparisons\*/

 estimate 'S chicago nosteroid' exp(b0 + **0.5**\*b1 + b2);

 estimate 'S chicago steroid' exp(b0 + **0.5**\*b1 + b2 + b4);

 estimate 'S Denver nosteroid' exp(b0 + **0.5**\*b1 + b2 + b5);

 estimate 'S Denver steroid' exp(b0 + **0.5**\*b1 + b2 + b4 + b5 + b6);

 contrast 'S denver steroid vs nonsteroid' b4 + b6;

 contrast 'S chicago steroid vs nonsetroid' b4;

 contrast 'S nonsteroid denver vs chicago' b5;

 contrast 'S steroid denver vs chicago' b5 + b6;

 contrast 'string vs nasal' b2 -b3;

 ods output AdditionalEstimates=fig3;

**run**;

/\*NB\*/

**proc** **nlmixed** data=in.EOE\_2015\_06\_25;

 where last='Fusobacterium' and type in ('O' 'S' 'N');

 parms b0=-**4** b1=**.1** b2=-**.2** b3=-**2** b4=-**1** b5=**.5** b6=**1**

 k=**.8** su=**.2** ;

 bounds su > **0**;

 vu=su\*su;

 /\* define linear function for count part of model \*/

eta = b0 + b1\*eoe + b2\*string + b3\*nasal + b4\*steroid + b5\*denver + b6\*steroid\*denver + ltotal + ui;

 lambda = exp(eta);

 /\* define ZINB log likelihood \*/

loglike=lgamma(r + **1**/k) -lgamma(r + **1**) -lgamma(**1**/k) + r\*log(k\*lambda) - (r + **1**/k)\*log(**1** + k\*lambda);

 model r ~ general (loglike);

 random ui ~ normal (**0**, vu) subject=Subject;

**run**;