Communications for Statistical Applications and Methods
2014, Vol. 21, No. 5, 423–433

DOI: http://dx.doi.org/10.5351/CSAM.2014.21.5.423
Print ISSN 2287-7843 / Online ISSN 2383-4757

Cumulative Sums of Residuals in GLMM and Its Implementation

DoYeon Choi[a], KwangMo Jeong[1,a]

[a]Department of Statistics, Pusan National University, Korea

Abstract

Test statistics using cumulative sums of residuals have been widely used in various regression models including generalized linear models(GLM). Recently, Pan and Lin (2005) extended this testing procedure to the generalized linear mixed models(GLMM) having random effects, in which we encounter difficulties in computing the marginal likelihood that is expressed as an integral of random effects distribution. The Gaussian quadrature algorithm is commonly used to approximate the marginal likelihood. Many commercial statistical packages provide an option to apply this type of goodness-of-fit test in GLMs but available programs are very rare for GLMMs. We suggest a computational algorithm to implement the testing procedure in GLMMs by a freely accessible R package, and also illustrate through practical examples.

Keywords: Clustered data, generalized linear mixed model, cumulative sums of residuals, gaussian process, gradient, hessian matrix.

1. Introduction

The GLM is a representative method to model categorical responses such as binomial counts or Poisson counts data. A GLM is specified in terms of three components; the random distribution of responses, the link function, and the linear predictor of covariate variables. For example, for a dataset of patients having epileptic seizures we may fit a Poisson GLM with log link to predict the mean number of seizures by the related covariates such as the treatment and age.

The goodness-of-fit(GOF) test using residuals has been widely used for checking model specifications in GLMs. Some kinds of GOF tests using cumulative sums of residuals(CUSUM) are so popular that they can be performed by common statistical packages, for example, PROC GENMOD in SAS, or gof package in R. They provide the empirical p-value with the graphical plot of CUSUM statistic. The asymptotic processes of cumulative sums(CUSUM) of residuals in various models have been studied by many researchers among which we may refer to Pierce and Schafer (1986), Su and Wei (1991), Cook and Weisberg (1994), Stute (1997), Lin *et al.* (2002), Pan and Lin (2005).

A GLMM is an extension of GLM to be applied to the longitudinal or clustered data by incorporating the heterogeneity of subjects or clusters in terms of random effects. The GLMM can be specified in several respects; the distribution of random effects, the link function, the functional form of covariates, the overdispersion of response distribution. Under the correct model specification the residuals have a tendency centered around zero and we expect a simple plot of residuals should show any systematic departures from zero. But we have some difficulty to confirm the model misspecification or random fluctuation from a simple plot of residuals. After Pan and Lin (2005) proposed the objective

This work was supported by a 2-year Research Grant of Pusan National University.

[1] Corresponding author: Department of Statistics, Pusan National University, Jangjeon-Dong, Kumjung-Gu, Pusan 609-735, Korea. E-mail: kmjung@pusan.ac.kr

Published 30 September 2014 / journal homepage: http://csam.or.kr

and informative procedures using residuals, the GOF test based on CUSUM has a widespread use as a complementary method for the overall GOF tests such as Pearson's chi-squared statistic and deviance statistic.

In contrast to the CUSUM test statistic in GLMM we may list several recent papers such as Tang (2010), Chen (2011) and Hansen (2012) who studied GOF statistics having various forms. Tang (2010) suggested a chi-squared type statistic for the logistic-normal GLMM based on the residuals by partitioning the covariates space into non-overlapping cells; however, Chen (2011) considered a score type testing procedure for the normality assumption of random intercept by using the semi-nonparametric density representation technique. A parametric bootstrap that involves heavy computational burden has been applied to find the significance value. Lin and Chen (2012) proposed a GOF statistic having quadratic form based on the nonparametric smoothing of residuals to check the functional form of covariates. Finally, Hansen (2012) also studied a Cramer-Von-Mises type GOF test for the overall fit of GLMM with special emphasis on the autoregressive logistic regression models. As we reviewed briefly there are so many papers for the GOF testing procedures in various respects.

In this paper, we are concerned with the misspecification of random effects distribution that are related with the overdispersion and the intracluster correlation. We reviewed some recent papers and found that the available packages to perform CUSUM test in GLMM are rare. We suggest the implementation algorithms for the CUSUM testing procedure in GLMM that can be programmed in R package. In Section 2, we briefly review the CUSUM process of residuals. We discuss the implementation algorithm via R package in Section 3. In Section 4 we illustrate the application of the program through examples. Finally, we summarize and comment on further research areas.

2. Cumulative Sums of Residuals in GLMM

2.1. Gaussian quadrature approximation to the likelihood

Let y_{ij} be an outcome variable having $p \times 1$ vector of covariates $\mathrm{X_{ij}} = (x_{1ij}, x_{2ij}, \ldots, x_{pij})'$, where $i = 1, \ldots, n$ and $j = 1, \ldots, t_i$. A GLM can be written as

$$g(\mu_{ij}) = \mathbf{X_{ij}'}\boldsymbol{\beta}, \tag{2.1}$$

where $g(\cdot)$ is a link function, $\mu_{ij} = E(y_{ij}|\mathrm{X_{ij}})$ and β is a $p\times 1$ vector of unknown regression coefficients.

For the clustered data we frequently encounter the intra cluster correlation within each cluster. For example, the observations on the fetuses in a litter of female rats are usually positively correlated. Similarly, the observations from longitudinal studies can be considered as a clustered data. The GLMM is one of the most common methods for these kinds of clustered data by incorporating the heterogeneity between clusters in terms of random effects.

Let $\mathbf{u_i}$ be a $q\times 1$ vector of cluster-specific random effects, which are usually assumed to be independently and identically distributed with normal density function $h(\mathbf{u_i})$, that is, $\mathbf{u_i} \sim N(\mathbf{0}, \mathbf{D})$, where $\mathbf{D}$ is a covariance matrix that depends on a vector $\boldsymbol{\gamma}$. With a little abuse of notation we let $\mu_{ij} = E(y_{ij}|\mathbf{u_i})$ be the conditional mean of response variable given $\mathbf{u_i}$. A GLMM is of the form

$$g(\mu_{ij}) = \mathbf{X_{ij}'}\boldsymbol{\beta} + \mathbf{Z_{ij}'}\mathbf{u_i}, \tag{2.2}$$

where $\mathbf{Z_{ij}}$ denoting a $q \times 1$ vector of covariates associated with the i^{th} cluster. The y_{ij}'s are assumed to be conditionally independent given $\mathbf{u_i}$ within each cluster having a density $f_{y|u}(y_{ij}|\mathbf{u_i})$, and also are independent across all clusters. The maximum likelihood estimator(MLE) $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta} = (\boldsymbol{\beta}', \boldsymbol{\gamma}')'$ can

routinely be found by maximizing the loglikelihood function

$$L(\boldsymbol{\theta}) = \sum_{i=1}^{n} L_i(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left\{ \int \prod_{j=1}^{t_i} f_{y|u}(y_{ij}|\mathbf{u_i}) h(\mathbf{u_i}) d\mathbf{u_i} \right\}, \tag{2.3}$$

where $L_i(\boldsymbol{\theta})$ is the i^{th} marginal loglikelihood that is obtained by integrating out $\mathbf{u_i}$. We refer to McCullagh and Nelder (1989) for a general discussion on the GLMM. The integral above cannot explicitly be computed in general. Therefore we need to approximate it by a numerical method such as the Gaussian quadrature method by Pinheiro and Bates (1995). Given the Gauss-Hermite abscissas and weights (z_j, w_j), $j = 1, \ldots, r$, the adaptive Gaussian quadrature centered at $\hat{\mathbf{u}}_{\boldsymbol{i}}$ approximates the integral as

$$\int \prod_{j=1}^{t_i} f(y_{ij}|\mathbf{u_i}) h(\mathbf{u_i}) d\mathbf{u_i} \approx 2^{\frac{q}{2}} \left|\hat{\mathbf{L}}_{\boldsymbol{i}}\right|^{-\frac{1}{2}} \sum_{j_1}^{r} \cdots \sum_{j_q}^{r} \left[\prod_{j=1}^{t_i} f(y_{ij}|\mathbf{a}_{j_1,\ldots,j_q}) h(\mathbf{a}_{j_1,\ldots,j_q}) \prod_{k=1}^{q} w_{j_k} \exp\left(z_{j_k}^2\right) \right]. \tag{2.4}$$

The empirical Bayes predictor $\hat{\mathbf{u}}_{\boldsymbol{i}}$ is obtained by minimizing the quantity

$$-\log \left(\prod_{j=1}^{t_i} f(y_{ij}|\mathbf{u_i}) h(\mathbf{u_i}) \right). \tag{2.5}$$

The Hessian matrix $\tilde{\mathbf{H}}_i$ comes from the empirical Bayes minimization, and $\mathbf{a}_{j_1,\ldots,j_q} = \hat{\mathbf{u}}_i + 2^{1/2} \tilde{\mathbf{H}}_i^{-1/2} z_{j_1,\ldots,j_q}$, where $z_{j1,\ldots,jq} = (z_{j1}, \ldots, z_{jq})'$ with $q = \dim(\mathbf{u_i})$. The gradient vector that is defined by

$$\mathbf{U}(\boldsymbol{\theta}) = \frac{\partial \log L(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{i=1}^{n} \mathbf{U_i}(\boldsymbol{\theta}),$$

where $\mathbf{U_i}(\boldsymbol{\theta}) = \partial \log L_i(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$, plays an important role in the limiting process of CUSUM process. Under some regularity conditions the MLE $\hat{\theta}$ satisfies the relationship

$$n^{\frac{1}{2}} \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta} \right) = n^{-\frac{1}{2}} \boldsymbol{\Omega^{-1}} \mathbf{U}(\boldsymbol{\theta}) + o_p(1), \tag{2.6}$$

where $\boldsymbol{\Omega} = \lim_{n->\infty} \mathbf{I}_n(\boldsymbol{\theta})$ with $\mathbf{I}_n(\theta) = -n^{-1} \times \partial^2 \log L(\boldsymbol{\theta}) / \partial \boldsymbol{\theta} \partial \boldsymbol{\theta}'$. We also note that $n^{1/2}(\hat{\theta} - \theta)$ is asymptotically normal with mean zero and covariance matrix $\boldsymbol{\Omega^{-1}}$.

2.2. A Goodness-of-fit test using CUSUM

Under a correct model specification the residuals have a tendency centered around zero. The systematic departure from this tendency indicates some sort of lacks of fit, that is, misspecified functional form of covariates, incorrect link function, and other violation of assumptions related with the fitted model. The CUSUM of residuals with respects to a certain indexes provides more objective information compared to a simple plot of residuals. The marginal residual is a difference between y_{ij} and the marginal mean $E(y_{ij})$ obtained by

$$m_{ij}(\boldsymbol{\theta}) = E(y_{ij}) = \int g^{-1} \left(\mathbf{X}_{\boldsymbol{ij}}^{'} \boldsymbol{\beta} + \mathbf{Z}_{\boldsymbol{ij}}^{'} \mathbf{u_i} \right) h(\mathbf{u_i}) d\mathbf{u_i}. \tag{2.7}$$

Let $e_{ij} = y_{ij} - \hat{m}_{ij}$ be the marginal residual with $\hat{m}_{ij}$ denoting the predicted value $m_{ij}(\hat{\boldsymbol{\theta}})$. The CUSUM statistic with respect to $\hat{m}_{ij}$ would be an informative measure to check the distributional assumptions of random effects $\mathbf{u}_i$ related with responses. We consider a CUSUM process

$$W(t) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \sum_{j=1}^{t_i} I\left(\hat{m}_{ij} \leq t\right) e_{ij}, \tag{2.8}$$

where t is a real number. This type of test statistic was firstly studied in GLMs by Lin *et al.* (2002), and later extended to GLMMs by Pan and Lin (2005). We have a target to implement this testing procedure in a freely accessible package. According to the theoretical discussion by Pan and Lin (2005), the CUSUM process $W(t)$ converges in distribution to a zero-mean Gaussian process under a correct model specification. Furthermore, the significance of a CUSUM test can be determined numerically from the limiting process of $W(t)$.

To discuss the limiting process we let $\hat{W}(t)$ as follows

$$\widehat{W}(t) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \left\{ \sum_{j=1}^{t_i} I\left(\hat{m}_{ij} \leq t\right) e_{ij} + \boldsymbol{\eta}'\left(t; \hat{\boldsymbol{\theta}}\right) \mathbf{I}_n^{-1}\left(\hat{\boldsymbol{\theta}}\right) \mathbf{U}_i\left(\hat{\boldsymbol{\theta}}\right) \right\} G_i, \tag{2.9}$$

where

$$\boldsymbol{\eta}'(t; \boldsymbol{\theta}) = -\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{t_i} I\left(\hat{m}_{ij} \leq t\right) \frac{\partial m_{ij}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}.$$

In (2.9) the random variates $G_1, \ldots, G_n$ are *iid* from $N(0, 1)$ that are independent of the given data $(y_{ij}, \mathbf{X}_{ij}, \mathbf{Z}_{ij})$. It is known that the limiting distribution of $W(t)$ is the same as the distribution of $\widehat{W}(t)$ given the data $(y_{ij}, \mathbf{X}_{ij}, \mathbf{Z}_{ij})$. This means that the significance of the CUSUM process $W(t)$ can be determined numerically using the values of $\hat{W}(t)$ that are obtained repeatedly by generating $G_1, \ldots, G_n$ from $N(0, 1)$. The CUSUM process $W(t)$ has a tendency to fluctuate randomly around zero under the correct model specification. The Kolmogorov-Smirnov type statistic over t values, that is,

$$W = \sup_t |W(t)|, \tag{2.10}$$

is a natural choice as a goodness-of-fit test for the model misspecification related with the distributional assumption of GLMM. The unusually large value of W compared with the realizations $\hat{W} = \sup_t |\widehat{W}(t)|$ indicates a certain kind of lacks of fit for the fitted model. The empirical p-value is defined as the proportion of $\hat{W}$ values greater than the observed W value. The exceptionally small p-value may denote a violation of normality assumption on random effects. We need to modify the assumed model by refitting the GLMM by taking other distributions on random effects. We may refer to Liu and Yu (2008) for a detailed discussion on the fitting of nonnormal random effects GLMM.

2.3. Algorithms and R-library functions

To perform the CUSUM process $W(t)$ and its limiting process $\widehat{W}(t)$ we suggest the algorithms to be implemented via R functions listed in the Appendix. We assume a Poisson-normal GLMM having simple random effects with $Z_{ij} = 1$ but they can be easily extended to include general GLMMs such as binomial-logistic model. Firstly, in algorithms A1 and A2 we compute the MLEs and residuals, and the auxiliary statistics that are necessary to $\widehat{W}(t)$ using R functions *adaptivegq*, *densegh*, and *conddens*.

We comment that the MLEs can also be obtained by the package *glmmML* that is a routine one in R to fit GLMM but it does not provide the gradient vector and the Hessian matrix that are necessary in A2. To obtain the gradient $\mathbf{U}_i(\hat{\boldsymbol{\theta}})$ and the Hessian $\mathbf{H}(\hat{\boldsymbol{\theta}})$ the ***numDeriv*** of R is also additively linked with the R functions mentioned above. Finally, the CUSUM test statistic W and its realization $\hat{W}$ are obtained in A3. Many values of $\hat{W}$ are repeatedly computed to find a p-value of the statistic W. The CUSUM statistic $W(t)$ and its realizations $\widehat{W}(t)$ are computed by the functions $\max W$ and $\max \hat{W}$, respectively. Main R functions are listed in the Appendix. We summarize the algorithms.

A1. Find MLEs of $\boldsymbol{\theta}$ in GLMM by Gaussian Quadrature Approximation

A1-1. Compute marginal means

A1-2. Obtain Residuals

A2. Find gradient vector $U_i(\hat{\boldsymbol{\theta}})$ and Information matrix $I_n(\hat{\boldsymbol{\theta}})$

A2-1. Find $I_n^{-1}(\hat{\boldsymbol{\theta}})$

A2-2. Compute the $\boldsymbol{\eta}'(t;\hat{\boldsymbol{\theta}})$

A3. Compute CUSUM Statistic and its Realizations

A3-1. Find a test statistic $W = \sup|W(t)|$

A3-2. Simulate realized values $\widehat{W} = \sup|\widehat{W}(t)|$ repeatedly

A3-3. Print out p-value with graphs of $W(t)$ and some realizations of $\widehat{W}(t)$

3. Illustrating Examples

Example 1. We firstly illustrate the application of CUSUM statistic implemented by R package for the epileptic seizure data from Thall and Vail (1990). For each subject of 59 patients having epileptic seizures we observe repeatedly the counts of epileptic seizures during 2-week periods prior to the four visits of clinic. The related covariate variables are the baseline counts(x_1), treatment(x_2), age(x_3), and the time of visit(x_4).

We fit a Poisson-normal GLMM with random intercept given by

$$\log(\mu_{ij}) = \beta_0 + \beta_1 x_{1ij} + \beta_2 x_{2ij} + \beta_3 x_{3ij} + \beta_4 x_{4ij} + u_i,$$

where $\mathrm{u}_i \sim N(0, \sigma^2)$ and $\mathbf{X_{ij}} = (x_{1ij}, x_{2ij}, x_{3ij}, x_{4ij})'$, $i = 1, 2, \ldots, 59$; $j = 1, \ldots, 4$. We note that covariates x_1 and x_3 are log-transformed. The x_4 is included as an indicator variable having 1 only for the fourth visit. The log scaled is not significant with p-value 0.333, and also the $\hat{\sigma} = 0.516$ with $se(\hat{\sigma}) = 0.0579$. The homogeneity test for $\sigma = 0$ is very significant with $p < 0.000$.

Figure 1 shows the plots of $W(t)$ depicted with a bold line and 50 realizations of $\widehat{W}(t)$ overlaid with gray lines. The p-value = 0.542 from 1000 replications of $\hat{W}(t)$ denotes that the assumed model specification is adequate as already had been analyzed by Waagepetersen (2006) to test the random distribution using other test statistic. The p-values are subject to random variation but according to the independent repetitions of 1000 times they are in the range of about 0.42 through 0.58.

Example 2. As a second example of overdispersed count responses Figure 2 shows a time series of 534 monthly counts of mumps cases in New York City, 1928–1972 (Waagepetersen, 2006). In Figure 2 we see a pronounced seasonal variation varying from the smallest count 20 to the maximum count of 1956. The graph of autocorrelation function(omitted for space) also denotes strong autocorrelations

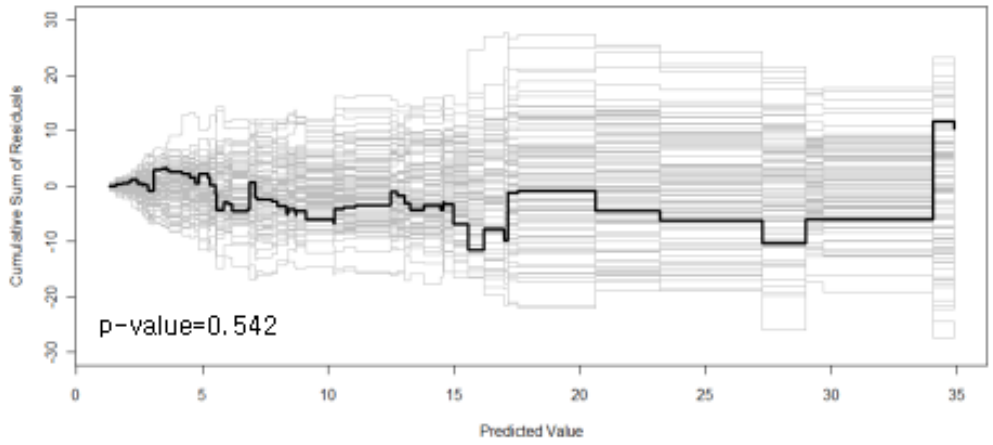

Figure 1: *Plot of CUSUM process against predicted values*

with significant periodic variation of mumps. The mean and variance of mumps is 487.7 and 147721.5, respectively, thus there exists a large extra variation expected than the variance of Poisson distribution. This dataset was also analyzed *e.g* Jeong (2012). We consider a Poisson-normal GLMM having several covariates such as month and time variable measured in unit of month given by

$$\log(\mu_i) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 (x_{2i})^2 + u_i, \quad i = 1, 2, \ldots, 534,$$

where x_{1i} denotes the categorical covariate month($1 \sim 12$) and X_{2i} is the log transformed time variable. Month itself is included to explain the seasonal variation, and the time variable is the elapsed time measured in unit of month. The signs of $\hat{\beta}_1$ denote seasonal variation that is positive until June but negative from July. Figure 2 shows a strong correlation between observations with $\hat{\sigma} = 0.43$. Figure 3 shows the CUSUM test for this data set shows a p-value of 0.04; consequently, we doubt model misspecification, in particular there seems to be a violation of the independent normal random effects. Additionally, there also exists an overdispersion as had been analyzed by Waagepetersen (2006). We now comment on the computational burden in performing the CUSUM test for the previous two examples. In the case of Example 1 the System CPU time takes about 0.54 seconds but it is 1.36 seconds for the Example 2 as measured by CPU Intel(R) E8400 with duo memories 3.00 GHz.

4. Conclusion and Further Research

The cumulative sums of residuals are useful to diagnose the misspecification of general linear models but the available package to perform this type of test is very rare in GLMM. In this paper we are purposed to implement the testing procedure via a easily accessible package such as R. Even though the well-known packages provide a routine result on the MLEs and assessment measures they can not be used to do check model misspecification using the CUSUM of residuals.

We suggest an algorithm to implement the CUSUM process via R functions focusing on the Poisson-normal GLMM. The algorithm have been applied to the practical examples by diagnosing the distributional assumption related with overdispersion and the serial correlation. As a further research it would be desirable to generalize the functions to include other GLMMs such as binomial-logistic GLMM. We haven't performed a Monte Carlo study to check the performance of the CUSUM test due to a computational burden of the program. We need to improve the implementation program to

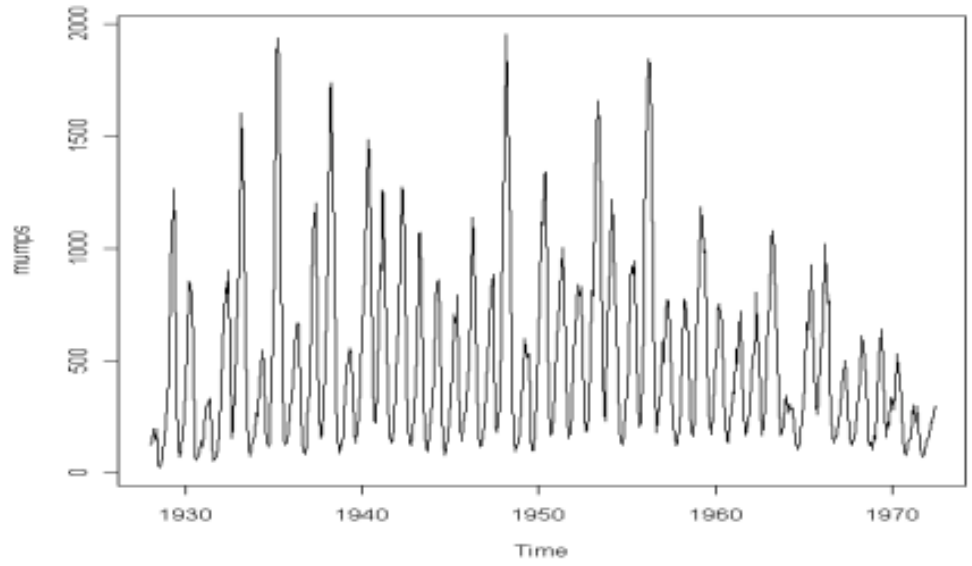

Figure 2: *Plot of CUSUM Process against Predicted values*

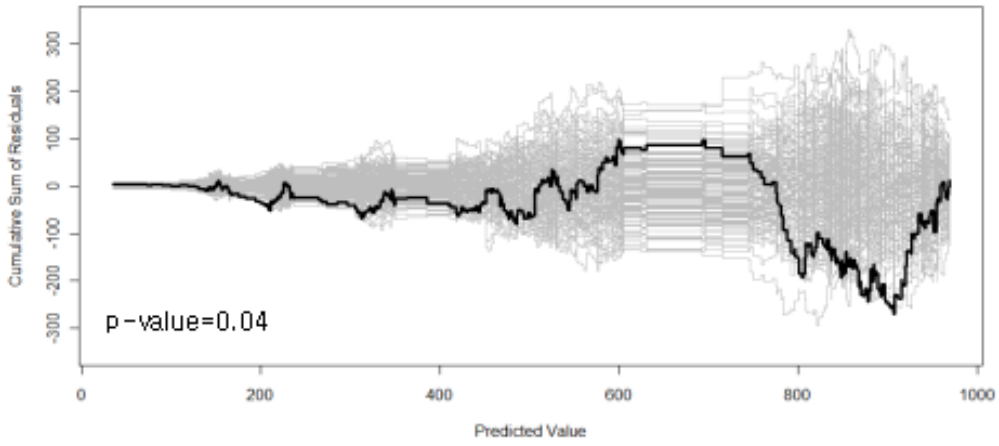

Figure 3: *Plot of CUSUM Process against Predicted values*

speed up computational time. The whole program is omitted due to its lengthy pages and finally we hope that the generalized source program to be opened in R environment.

Appendix: I. Explanation for Example 3.1

```
#seize data in R
seizedat=read.table("seize.dat",header=T)
attach(seizedat)
y=seize # y: response
logbase=log(base/4); logage=log(age); V4=as.numeric(visit==4)
```

```
X=cbind(rep(1,length(logbase)),logbase,trt,logage,V4) # X: design matirx
group=subject # group: subject id
ssubject=unique(subject)
for (i in 1:length(ssubject)){
group[subject==ssubject[i]]=i
}

# beta, sigma denote the estimated coefficient of beta and sigma, respectivel
y.
library(glmmML)
model=glmmML(y ~ logbase+trt+logage+V4,family=Poisson ,cluster=group)
sigma=model$sigma; beta=coef(model)
# n: sample size, p: cluster size
main(n=59,p=4,y,X,beta,sigma,group,ssubject,offset=c(),POISSON=1,re=1000)
```

Appendix: II. R functions

```
(1) Implementing algorithms A1 and A2 via adaptivegq, densegh, conddens and n
eglogdens

adaptivegq=function(n,y,X,beta,sigma,adaptive=T){
gh=.Fortran("ghq",as.integer(n),as.double(rep(0,n)),as.double(rep(0,n)))
if (adaptive){
object=function(a){
temp=0.5*a^2-conddens(y,X,beta,sigma,a)[1]
return(temp)
}
minimize=optimize(object,c(-5,5)); ahat=minimize$minimum
hess=1+sigma^2*sum(exp(sigma*ahat+X%*%beta))
grad=ahat-sigma*sum(y)+sigma*sum(exp(sigma*ahat+X%*%beta))
temp=gh[[2]]; gh[[2]]=gh[[2]]/sqrt(hess)+ahat
gh[[3]]=gh[[3]]*exp(temp^2/2-gh[[2]]^2/2)/sqrt(hess)
}
return(gh)
}

densgh=function(y,X,beta,sigma,gh){
a=gh[[2]]; w=gh[[3]]; temp=rep(0,length(a)); cd=rep(0,length(a))
for (i in 1:length(a)){
temp=conddens(y,X,beta,sigma,a[i]); cd[i]=temp[1]+temp[2]
}
maxcd=max(cd); cd=cd-maxcd; temp=w*exp(cd)
return(c(sum(temp),maxcd))
}

conddens=function(y,X,beta,sigma,a){
```

```
etadet=X%*%beta; eta=etadet+sigma*a
#exp of sum of two components times prod(y!)^{-1} is condensity
return(c(sum(y*sigma*a-exp(eta)),sum(etadet*y)))
}

neglogdens=function(par,arg){
beta=par[1:(length(par)-1)]; sigma=par[length(par)]; nobet=length(beta); n=20
logdens=0
for (l in 1:length(ssubject)){
first=group==ssubject[l]; Xtemp=X[first,]
if (!is.matrix(Xtemp)) Xtemp=t(as.matrix(Xtemp))
if (dim(Xtemp)[2]!=length(beta)) Xtemp=t(Xtemp)
gh=ghadap(n,y[first],Xtemp,beta,sigma,adaptive)
temp=densgh(y[first],Xtemp,beta,sigma,gh)
logdens=logdens+log(temp[1])+temp[2]
}
return(-logdens)
}

(2) Implementing A3 via maxW and maxhatW

maxW=function(par,arg){
beta=par[1:(length(par)-1)]; sigma=par[length(par)]
## marginal residuals
Xbeta=X%*%beta; mhat=exp(Xbeta+((sigma^2)/2)); residuals=y-mhat
rmax=0; maxWr=0; rlength=length(unique(mhat))
r=sort(unique(mhat)); pltdata=array(0,rlength)
iter=1
while(iter<=rlength){
if(iter==1){
pltdata[1]=sum(residuals[mhat <= r[iter]])/sqrt(n)
maxWr=abs(pltdata[1]); rmax=r[iter]
}
else{
pltdata[iter]=sum(residuals[mhat <= r[iter]])/sqrt(n)
tempWr=abs(pltdata[iter])
if(tempWr>maxWr){
maxWr=tempWr; rmax=r[iter]
}
}
iter=iter+1
}
return(list(maxWr,rmax))  ## sup W(r) and r in sup W(r)
}

maxhatW=function(par,arg,maxWr,re){
```

```
# Assign the no. of repetitions : re
# Initial Seed values : seed1
beta=par[1:(length(par)-1)]; sigma=par[length(par)]
## marginal residuals
Xbeta=X%*%beta; mhat=exp(Xbeta+((sigma^2)/2)); residuals=y-mhat
## eta
diff=matrix(0,n*p,length(beta)+1)
for(i in 1:length(beta)) diff[,i]=mhat*X[,i]
diff[,(length(beta)+1)]=mhat*sigma; info=-hessian/n; infosolve=solve(info)
rep_What=array(0,re); hpltdata=matrix(0,rlength,re)
for(rep in 1:re){
rmax=0; seed=seed1+rep; set.seed(seed); G=rnorm(n,0,1)
What=array(0,rlength); wh.maxtemp=0; iter=1
while(iter<=rlength){
eta=matrix(0,1,length(beta)+1)
for(i in 1:(length(beta)+1)){
temp=diff[,i]; eta[1,i]=sum(temp[mhat <= r[iter]])
}
eta=-eta/n; res_ti=array(0,n)
for (i in 1:n){
first=(group==ssubject[i]); mhat_ti=mhat[first]
temp=residuals[first]
res_ti[i]=sum(temp[mhat_ti<=r[iter]])
temp=(res_ti[i]+eta%*%infosolve%*%(grad[i,]))*G[i]
What[iter]=What[iter]+temp
}
hpltdata[iter,rep]=What[iter]/sqrt(n)
What[iter]=abs(hpltdata[iter,rep])
if(What[iter]>wh.maxtemp) wh.maxtemp=What[iter]
iter=iter+1
}
rep_What[rep]=wh.maxtemp # sup What(r)
}
return(sum(rep_What>maxWr)/re)  ## Compute p-values
}

(3) Main step combining the functions

library(numDeriv)
main=function(n,p,y,X,beta,sigma,group,ssubject,offset,POISSON,re){
grad=matrix(0,length(ssubject),length(beta)+1)
hessian=hessian(neglogdensall,par=c(beta,sigma),arg=c(length(y),y,group,X,off
set))
for (l in 1:n){
first=(group==ssubject[l])
temp=grad(neglogdensall,par=c(beta,sigma),arg=c(length(y[first]),
```

```
y[first],group[first],X[first,],offset))
grad[l,]=temp
}
maxWr=maxW(par=c(beta,sigma),arg=c(length(y),y,group,X,offset))[[1]]
p.value=maxhatW(par=c(beta,sigma),arg=c(length(y),y,group,X,offset),maxWr,re)
}
return(p.value)
}
```

References

Chen, N. W. (2011). *Goodness-of-Fit Test Issues In Generalized Linear Mixed Models*, Unpublished Ph.D. Thesis, Graduate Studies of Texas A&M University.

Cook, D. R. and Weisberg, S. (1994). *An Introduction to Regression Graphics*, Wiley.

Hansen, A. M. (2012). *Goodness-of-Fit Tests for Autoregressive Logistic Regression Models and Generalized Linear Mixed Models*, Unpublished Ph.D. Thesis, University of California Riverside.

Jeong, K. M. (2012). Modelling count responses with overdispersion, *Communications for Statistical Applications and Methods*, **19**, 761–770.

Lin, K. C. and Chen, Y. J. (2012). Assessing generalized linear mixed models using residual analysis, *International Journal of Innovative Computing, Information and Control*, **8**, 5693–5701.

Lin, D. Y., Wei, L. J. and Ying, Z. (2002). Model-checking techniques based on cumulative residuals, *Biometrics*, **58**, 1–12.

Liu, L. and Yu, Z. (2008). A likelihood reformulation method in non-normal random effects models, *Statistics in Medicine*, **27**, 3105–3124.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*, Second edition, London: Chapman and Hall.

Pan, Z. and Lin, D. Y. (2005). Goodness-of-fit methods for generalized linear mixed models, *Biometrics*, **61**, 1000–1009.

Pierce, D. A. and Schafer, D. W. (1986). Residuals in generalized linear models, *Journal of the American Statistical Association*, **81**, 977–986.

Pinheiro, J. C. and Bates, D. M. (1995). Approximations to the loglikelihood function in nonlinear mixed-effects models, *Journal of Computational and Graphical Statistics*, **4**, 12–35.

Stute, W. (1997). Nonparametric model checks for regression, *The Annals of Statistics*, **25**, 613–641.

Su, J. Q. and Wei, L. J. (1991). A lack-of-fit test for the mean function in a generalized linear model, *Journal of the American Statistical Association*, **86**, 420–426.

Tang, M. (2010). *Goodness of Fit Tests for Generalized Linear Mixed Models*, Unpublished Ph.D. Thesis, Graduate School of the University of Maryland.

Thall, P. F. and Vail, S. C. (1990). Some covariance models for longitudinal count data with overdispersion, *Biometrika*, **46**, 657–671.

Waagepetersen, R. (2006). A simulation based goodness-of-fit test for random effects in generalized linear mixed models, *Scandinavian Journal of Statistics*, **33**, 721–731.

Received June 9, 2014; Revised July 24, 2014; Accepted August 19, 2014