# A Score Test of Homogeneity in Generalized Additive Models for Zero-Inflated Count Data

by

GAOWEI NIAN

B.S., Henan University, China, 2012

---

A REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2014

Approved by:

Major Professor
Wei-Wen Hsu

# Copyright

# Abstract

Zero-Inflated Poisson (ZIP) models are often used to analyze the count data with excess zeros. In the ZIP model, the Poisson mean and the mixing weight are often assumed to depend on covariates through regression technique. In other words, the effect of covariates on Poisson mean or the mixing weight is specified using a proper link function coupled with a linear predictor which is simply a linear combination of unknown regression coefficients and covariates. However, in practice, this predictor may not be linear in regression parameters but curvilinear or nonlinear. Under such situation, a more general and flexible approach should be considered. One popular method in the literature is Zero-Inflated Generalized Additive Models (ZIGAM) which extends the zero-inflated models to incorporate the use of Generalized Additive Models (GAM). These models can accommodate the nonlinear predictor in the link function. For ZIGAM, it is also of interest to conduct inferences for the mixing weight, particularly evaluating whether the mixing weight equals to zero. Many methodologies have been proposed to examine this question, but all of them are developed under classical zero-inflated models rather than ZIGAM. In this report, we propose a generalized score test to evaluate whether the mixing weight is equal to zero under the framework of ZIGAM with Poisson model. Technically, the proposed score test is developed based on a novel transformation for the mixing weight coupled with proportional constraints on ZIGAM, where it assumes that the smooth components of covariates in both the Poisson mean and the mixing weight have proportional relationships. An intensive simulation study indicates that the proposed score test outperforms the other existing tests when the mixing weight and the Poisson mean truly involve a nonlinear predictor. The recreational fisheries data from the Marine Recreational Information Program (MRIP) survey conducted

by National Oceanic and Atmospheric Administration (NOAA) are used to illustrate the proposed methodology.

# Table of Contents

# List of Figures

# List of Tables

# Acknowledgments

First of all, I would like to thank my major professor, Dr. Wei-Wen Hsu, for all his guidance, suggestions and encouragement.

I would also like to thank Dr. Weixin Yao and Dr. Christopher Vahl for their willingness to serve on my committee and for their valuable insight.

Finally, I would like to thank my parents, my brother and my sister for their support and endless love and I would also like to thank my wife for her encouragement during my graduate education. In addition, I will thank everyone who helped me during the completion of the report.

# Chapter 1

# Introduction

The Zero-Inflated Poisson (ZIP) regression model is a simple two-component mixture model that is often used for count data containing many zeros. In the ZIP model, one component occurring with the probability $\omega$ is a degenerate distribution with mass one at zero, while the other component occurring with the probability $(1-\omega)$ is a standard Poisson distribution with the mean $\mu$ (see, for example, Lambert, 1992).

Under this classical ZIP model, the effect of covariates on the Poisson mean and the mixing weight is specified by a proper link function (such as log link; logit link function) coupled with a linear predictor which is simply a linear combination of unknown regression coefficients and covariates. However, in practice, this predictor may not be linear in regression parameters but curvilinear or nonlinear. In other words, the observed features of the data may not be consistent with the ZIP model. For example, in the paper of Lam et al (2006), they found that age had a nonlinear effect on the outcome variable, the number of days of missed primary activities in a given period. In the paper of Liu and Chan (2011), they found that sampling (Julian) day had a nonlinear effect on the outcome variable, jellyfish catch per unit. For such a problem where the predictor is not linear, one popular method called Zero-Inflated Generalized Additive Models (ZIGAM) which extended the zero-inflated models to incorporate the use of Generalized Additive Models (GAM) has been discussed widely (See,

for example, Barry and Welsh, 2002; Ma et al., 2010). These models can accommodate the nonlinear predictor in the link function.

As a goodness-of-fit test, it is also of interest to evaluate whether the mixing weight in the ZIGAM equals to zero. But the revelent methodologies are all developed under classical ZIP models rather than ZIGAM (see, for example, Jansakul and Hinde, 2002; Todem et al., 2012). To our knowledge, there is no test for homogeneity under the framework of ZIGAM.

In this report, we propose a generalized score test to evaluate whether the mixing weight equals to zero under the frame work of ZIGAM, focusing on the Poisson model. Technically, the proposed approach is developed based on the novel transformation proposed by Todem et al. (2012) and an assumption used by Ma et al. (2010). Their assumption assumes that the smooth components of covariates in the Poisson mean and the mixing weight have proportional relationships. In fact, ZIGAM coupled with this assumption is called Constrained Zero-Inflated Generalized Additive Models (COZIGAM) (Liu and Chan, 2011).

In sum, the proposed test is developed under the framework of COZIGAM. A resampling approach proposed by Lin et al. (1994) is adopted to characterize the null limiting distribution of our test statistic.

This report is organized as follows. In chapter 2, we briefly introduce the ZIP model and the COZIGAM. In chapter 3, the proposed score test based on the approach of Todem et al. (2012) is discussed here as well as the resampling skill. In chapter 4, the performances of the proposed score test are compared to those of the existing score tests (Jansakul and Hinde, 2002 and Todem et al. 2012). Also, a recreational fisheries data set is used to illustrate the proposed methodology. Finally, some conclusions are provided in chapter 5.

# Chapter 2

# Costrained Zero-Inflated Generalized Additive Models

## 2.1 Zero-inflated Poisson Model

Assume that $y_i$, $i = 1, \cdots, n$, are counts from a ZIP model and $\boldsymbol{x_i} = (x_{i1}, x_{i2}, \cdots, x_{ip})'$ is the corresponding $p \times 1$ vector of covariates. The probability mass function of the mixture model is

$$Pr(Y_i = y_i) = \begin{cases} \omega_i + (1-\omega_i)\exp\{-\mu_i\}, & \text{if } y_i{=}0, \\ (1-\omega_i)\frac{\exp\{-\mu_i\}\mu_i^{y_i}}{y_i!}, & \text{if } y_i > 0. \end{cases}, \tag{2.1}$$

where $\mu_i$ is the mean of the standard Poisson distribution and $\omega_i$ is known as the mixing weight. In the ZIP model, the zeros are generated from two different components: a degenerate distribution with mass one at zero and a standard Poisson distribution with mean $\mu_i$. The first component occurs with the probability $\omega_i$ and produces only zeros, while the second component occurs with the probability $(1-\omega_i)$ (Jansakul and Hinde, 2002). Lambert (1992) used two link functions for the Poisson mean $\mu_i$ and the mixing weight $\omega_i$. The link

functions are, respectively,

$$\log \mu_i = x_i^{'}\boldsymbol{\beta} \text{ and } \log \frac{\omega_i}{1-\omega_i} = g_i^{'}\boldsymbol{\gamma},$$

where $x_i$ and $g_i$ are covariate vectors and $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$ are $t \times 1$ and $r \times 1$ vectors of unknown parameters. In equation(2.1), generally the mixing weights $\omega_i$ are constrained in an interval,

$$-\exp\{-\mu_i\}/(1-\exp\{-\mu_i\}) \leqslant \omega_i \leqslant 1, \ i=1,...,n. \tag{2.2}$$

Since the mixing weight $\omega_i$ can be either negative, zero or positive, the corresponding models are Zero-Deflected Poisson, standard Poisson and Zero-Inflated Poisson, respectively (Dietz and Böhning, 2000).

## 2.2 Constrained Zero-inflated Generalized Additive Models

Generalized Additive Models (Hastie and Tibshirani, 1990; Wood, 2006) have been used widely in the literature to incorporate nonlinear predictors in Zero-Inflated models (See, for example, Barry and Welsh, 2002; Ma et al., 2010). It is more flexible to use GAM in a formal analysis due to the smooth terms.

In general, the Poisson means and the mixing weights in the ZIP model have the following structures,

$$g_\mu(\mu_i) = x_i^{'}\boldsymbol{\beta} \text{ and } g_\omega(\omega_i) = g_i^{'}\boldsymbol{\gamma},$$

where $\mu_i$ is the Poisson mean; $\omega_i$ is the proportion of the extra zeros; the $g_\mu(\cdot)$ and $g_\omega(\cdot)$ are link functions for the Poisson mean and mixing weight, which are often assumed to be log and logit link functions, respectively, under ZIP.

Under the framework of Zero-Inflated Generalized Additive Models, the more general structures for the Poisson means and the mixing weights can be assumed as,

$$g_\mu(\mu_i) = \beta_0 + \sum_{j=1}^{p} s_j(x_{ij}), \text{ and } g_\omega(\omega_i) = \alpha_0 + \sum_{j=1}^{p} h_j(x_{ij}),$$

where $\beta_0$ and $\alpha_0$ are unknown parameters; $s_j(\cdot)$ and $h_j(\cdot)$ are smooth functions which can be estimated by the penalized likelihood approach (See, for example, Green Peter J., 1987; Liu et al. 2012). Actually, the penalized likelihood estimator of $s_j$ generally equals to $Q$ linear combination of certain basis functions. In other words, the smooth function evaluated at $x_i$ could be expressed as $\boldsymbol{D_i\xi}$, where $\boldsymbol{D_i}$ is the $i^{th}$ row of the design matrix $\boldsymbol{D}$ of the basis functions, and $\boldsymbol{\xi}$ is the parameter vector to be estimated.

Here we also assume that some covariates affect the mixing weight and the nondegenerate distribution mean proportionally on the link scales (Liu and Chan, 2011). Under this assumption, the models have fewer unknown parameters and thus can be more accurately estimated (Ma et al., 2010). Specially, we assume

$$h_j = \delta s_j.$$

ZIGAM coupled with this assumption is called Constrained Zero-Inflated Generalized Additive Model (COZIGAM) (Liu and Chan, 2011). Under COZIGAM, the structures for the means of the nondegenerate distribution and the mixing weights become:

$$g_\mu(\mu_i) = \beta_0 + \sum_{j=1}^{p} s_j(x_{ij}), \text{ and } g_\omega(\omega_i) = \alpha_0 + \delta \sum_{j=1}^{p} s_j(x_{ij}),$$

where the unknown parameters of the model are consist of $\boldsymbol{\Theta}$=$(\beta_0, \alpha_0, \boldsymbol{\xi})$.

# Chapter 3

# Main framework

## 3.1 Testing Hypotheses

As a goodness-of-fit test, one is often interested in the two-sided hypotheses,

$$H_0 : \omega_i = 0, \text{ for all i vs. } H_a : \omega_i \neq 0, \text{ for some i,} \tag{3.1}$$

where $\omega_i$ satisfies the constraints in equation (2.2). To test these hypotheses, a suitable natural transformation (Todem et al., 2012) of $\omega_i$ that incorporates covariates should be considered. The natural transformation is then given by,

$$\omega_i = \frac{\pi_i - \exp\{-\mu_i\}}{1 - \exp\{-\mu_i\}}, \ 0 \leq \pi_i \leq 1. \tag{3.2}$$

where,

$$\pi_i = \exp(-\exp(x_i^{'}\boldsymbol{\gamma})) \ and \ \mu_i = \exp(x_i^{'}\boldsymbol{\beta}).$$

Based on the transformation in equation (3.2), the hypotheses (3.1) are formally represented as,

$$H_0 : \pi_i = \exp\{-\mu_i\}, \text{ for all i vs. } H_a : \pi_i \neq \exp\{-\mu_i\}, \text{ for some i.}$$

If a suitable parameterization of $\pi_i$ is considered, the homogeneity hypothesis above is reduced to a problem involving a small number of parameters (See, Todem et al., 2012). We already know that the Poisson means under GAM have the following form,

$$\mu_i = \exp\{\beta_0 + \sum_{j=1}^{p} s_j(x_{ij})\}.$$

Given the natural transformation and the proportional constraints on the GAM with zero-inflated data, the quantity $\pi_i$ is assumed to be,

$$\pi_i = \exp\{-\exp\{\alpha_0 + \delta \sum_{j=1}^{p} s_j(x_{ij})\}\}.$$

We assume the following reparameterization, $\gamma$=$\beta_0 - \alpha_0$. Then $\pi_i = \exp\{-\exp\{\beta_0 - \gamma + \delta \sum_{j=1}^{p} s_j(x_{ij})\}\}$. After the reparameterization, the new hypotheses are given,

$$H_0 : \gamma = 0 \text{ and } \delta = 1 \text{ vs. } H_a : \gamma \neq 0 \text{ or } \delta \neq 1. \tag{3.3}$$

## 3.2 Score test under COZIGAM

In classical parametric estimation, the unknown parameters are commonly estimated by maximum likelihood. However, for estimating GAMs, penalized likelihood method provides more powerful tools (Wood, 2000). For observations $y_1$,...,$y_n$, the penalized likelihood

function is given by

$$\begin{aligned}\ell\ell(\theta(\beta_0,\xi),\gamma,\delta,\lambda) =& \ell(\theta(\beta_0,\xi),\gamma,\delta) - \frac{1}{2}\lambda\boldsymbol{\xi}'\boldsymbol{K}\boldsymbol{\xi} \\ =& \sum_{i=1}^{n}\left\{ I(y_i=0)\log(\pi_i) + I(y_i>0)\log\left[\frac{(1-\pi_i)\exp\{-\mu_i\}\mu_i^{y_i}}{(1-\exp\{-\mu_i\})y_i!}\right]\right\} - \frac{1}{2}\lambda\boldsymbol{\xi}'\boldsymbol{K}\boldsymbol{\xi},\end{aligned}$$

where $\boldsymbol{K} = (\boldsymbol{K_1}, \boldsymbol{K_2}, \ldots, \boldsymbol{K_q}, \ldots, \boldsymbol{K_Q})'$ is the penalty matrix in which $\boldsymbol{K_q}$ is a $1 \times Q$ vector; $\lambda$ is the smoothing parameter corresponding to the penalty term, which controls the trade-off between the smoothness of the function and goodness-of-fit. The smoothing parameter can be selected by generalized cross-validation (GCV). Since the score test only requires the penalized likelihood estimates of the parameters under the null hypothesis, the general score test only involves fitting the standard Poisson model with GAM for the mean. Based on the above penalized likelihood function and the link function for Poisson mean $\mu_i$ and $\omega_i$, the score vector is

$$S(\theta(\beta_0,\xi),\gamma,\delta) = \begin{bmatrix} S_\theta(\theta(\beta_0,\xi),\gamma,\delta) \\ S_\gamma(\theta(\beta_0,\xi),\gamma,\delta) \\ S_\delta(\theta(\beta_0,\xi),\gamma,\delta)\end{bmatrix} = \begin{bmatrix} \frac{\partial\ell\ell(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\theta} \\ \frac{\partial\ell\ell(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\gamma} \\ \frac{\partial\ell\ell(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\delta}\end{bmatrix},$$

where

$$\begin{aligned}\frac{\partial\ell\ell}{\partial\beta_0} =& \sum_{i=1}^{n}\left\{ I_{(y_i=0)}\log(\pi_i) + I_{(y_i>0)}\left(\frac{-\pi_i\log(\pi_i)}{1-\pi_i}\right)\right. \\ & \left. + I_{(y_i>0)}\left(y_i - \mu_i - \frac{\mu_i\exp(-\mu_i)}{1-\exp(-\mu_i)}\right)\right\},\end{aligned}$$

$$\frac{\partial \ell\ell}{\partial \xi_q} = \sum_{i=1}^{n} \Bigg\{ \left[ I_{(y_i=0)} \log(\pi_i) + I_{(y_i>0)} \left( \frac{-\pi_i \log(\pi_i)}{1-\pi_i} \right) \right] \delta \sum_{j=1}^{p} s_{jq}(x_{ij}) + \left[ I_{(y_i>0)} \left( y_i - \mu_i - \frac{\mu_i \exp(-\mu_i)}{1-\exp(-\mu_i)} \right) \right] \sum_{j=1}^{p} s_{jq}(x_{ij}) \Bigg\} - \lambda \boldsymbol{K_q} \boldsymbol{\xi}',$$

$$q = 1, \ldots, Q,$$

$$\frac{\partial \ell\ell}{\partial \gamma} = \frac{\partial l}{\partial \pi_i} \frac{\partial \pi_i}{\partial \gamma} = \sum_{i=1}^{n} \Bigg\{ - I_{(y_i=0)} \log(\pi_i) + I_{(y_i>0)} \left( \frac{\pi_i \log(\pi_i)}{1-\pi_i} \right) \Bigg\},$$

$$\frac{\partial \ell\ell}{\partial \delta} = \frac{\partial l}{\partial \mu_i} \frac{\partial \mu_i}{\partial \delta} = \sum_{i=1}^{n} \Bigg\{ \left[ I_{(y_i=0)} \log(\pi_i) - I_{(y_i>0)} \left( \frac{\pi_i \log(\pi_i)}{1-\pi_i} \right) \right] \sum_{i=1}^{p} s_j(x_{ij}) \Bigg\}.$$

The expected information matrix $I(\theta(\beta_0, \xi), \gamma, \delta)$ can be partitioned as

$$I(\theta(\beta_0, \xi), \gamma, \delta) = \begin{bmatrix} I_\theta(\theta(\beta_0, \xi), \gamma, \delta) & I_{\theta\gamma}(\theta(\beta_0, \xi), \gamma, \delta) & I_{\theta\delta}(\theta(\beta_0, \xi), \gamma, \delta) \\ I_{\gamma\theta}(\theta(\beta_0, \xi), \gamma, \delta) & I_\gamma(\theta(\beta_0, \xi), \gamma, \delta) & I_{\gamma\delta}(\theta(\beta_0, \xi), \gamma, \delta) \\ I_{\delta\theta}(\theta(\beta_0, \xi), \gamma, \delta) & I_{\delta\gamma}(\theta(\beta_0, \xi), \gamma, \delta) & I_\delta(\theta(\beta_0, \xi), \gamma, \delta) \end{bmatrix},$$

where the elements $I_\theta$, $I_{\theta\gamma}$=$I'_{\gamma\theta}$, $I_{\theta\delta}$=$I'_{\delta\theta}$, $I_\gamma$, $I_{\gamma\delta}$=$I_{\delta\gamma}$ and $I_\delta$ are, respectively,
$-E\left[\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\theta\partial\theta'}\right]$, $-E\left[\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\theta\partial\gamma'}\right]$, $-E\left[\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\theta\partial\delta'}\right]$, $-E\left[\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\gamma^2}\right]$,
$-E\left[\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\gamma\partial\delta'}\right]$, $-E\left[\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\delta^2}\right]$.

Under the null hypothesis, the general score statistic is then

$$S_{\omega} = S'_{\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1)\Lambda^{-1}S_{\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1),$$

where $\hat{\theta}(\beta_0, \boldsymbol{\xi})$ is the maximum penalized likelihood estimator (MPLE) under the null model and the MPLE has asymptotically normality (see, Liu and Chan, 2011); and

$$S_{\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) = \begin{bmatrix} \sum_{i=1}^{n} \left\{ I_{(y_i=0)}(\hat{\mu}_i) - I_{(y_i>0)} \frac{\hat{\mu}_i \exp(-\hat{\mu}_i)}{1-\exp(-\hat{\mu}_i)} \right\} \\ \sum_{i=1}^{n} \left\{ \left[ -I_{(y_i=0)}(\hat{\mu}_i) + I_{(y_i>0)} \frac{\hat{\mu}_i \exp(-\hat{\mu}_i)}{1-\exp(-\hat{\mu}_i)} \right] \sum_{j=1}^{p} s_j(x_{ij}) \right\} \end{bmatrix},$$

$$\Lambda = I^*_{\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) - I^*_{\gamma,\delta,\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) I^{-1}_{\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) I^{*'}_{\theta,\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1),$$

where

$$I^*_{\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) = \begin{bmatrix} I_{\gamma}(\hat{\theta}(\beta_0, \xi), 0, 1) & I_{\gamma\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) \\ I_{\delta\gamma}(\hat{\theta}(\beta_0, \xi), 0, 1) & I_{\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) \end{bmatrix},$$

$$I^*_{\gamma,\delta,\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) = \begin{bmatrix} I_{\gamma\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) \\ I_{\delta\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) \end{bmatrix},$$

$$I^*_{\theta,\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) = \begin{bmatrix} I_{\theta\gamma}(\hat{\theta}(\beta_0, \xi), 0, 1) & I_{\theta\delta}(\hat{\theta}(\beta_0, \xi), 0, 1) \end{bmatrix}.$$

As we mentional in previous chapter, we assume that $\mu_i$=exp$\{\beta_0 + \sum_{j=1}^{p} s_j(x_{ij})\}$ and $\pi_i$= exp$\{-\exp\{\beta_0 - \gamma + \delta \sum_{j=1}^{p} s_j(x_{ij})\}\}$. Under the null model, given that $\gamma$=0, $\delta$=1 coupled with $\hat{\theta}(\beta_0, \boldsymbol{\xi})$ which is the estimate of $\theta(\beta_0, \boldsymbol{\xi})$, estimates of entries of the information matrix are given by,

$$I_{\theta\gamma}(\hat{\theta}(\beta_0,\xi),0,1) = \begin{bmatrix} \sum_{i=1}^{n}\left\{\frac{-\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\right\} \\ \sum_{i=1}^{n}\left\{\frac{-\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{j1}(x_{ij})\right\} \\ \vdots \\ \sum_{i=1}^{n}\left\{\frac{-\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{jQ}(x_{ij})\right\} \end{bmatrix},$$

$$I_{\theta\delta}(\hat{\theta}(\beta_0,\xi),0,1) = \begin{bmatrix} \sum_{i=1}^{n}\left\{\frac{\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{j}(x_{ij})\right\} \\ \sum_{i=1}^{n}\left\{\frac{\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{j}(x_{ij})\sum_{j=1}^{p}s_{j1}(x_{ij})\right\} \\ \vdots \\ \sum_{i=1}^{n}\left\{\frac{\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{j}(x_{ij})\sum_{j=1}^{p}s_{jQ}(x_{ij})\right\} \end{bmatrix},$$

$$I_{\gamma}(\hat{\theta}(\beta_0,\xi),0,1) = \sum_{i=1}^{n}\left\{\frac{\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\right\},$$

$$I_{\gamma\delta}(\hat{\theta}(\beta_0,\xi),0,1) = \sum_{i=1}^{n}\left\{-\frac{\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{j}(x_{ij})\right\},$$

$$I_{\delta}(\hat{\theta}(\beta_0,\xi),0,1) = \sum_{i=1}^{n}\left\{\frac{\hat{\mu_i}^2\exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})}\sum_{j=1}^{p}s_{j}(x_{ij})\sum_{j=1}^{p}s_{j}(x_{ij})\right\}.$$

This term $I_{\theta}(\hat{\theta}(\beta_0,\xi),0,1)$ can be obtained by an R program (See detailed information in Appendix B)

## 3.3 Resampling method

We'd like to use a resampling approach which applies the idea of Lin et al. (1994) to approximate the empirical distribution of the proposed score statistic. This resampling technique has been used widely in the literature (for example, Zhu and Zhang, 2006). In addition, this resampling approach can save a lot of time, compared with a simple nonpara-

metric bootstrap (Efron and Tibshirani, 1993).

Before applying the resampling approach, we need to make some basic preparation. Under the null model, we define $c_i$,

$$c_i = b_i(\hat{\theta}(\beta_0, \xi), 0, 1) - I^*_{\theta\gamma\delta} * I_\theta^{-1} * a_i(\hat{\theta}(\beta_0, \xi), 0, 1).$$

The function $c_i$ can be obtained from a Taylor expansion of $b_i(\hat{\theta}(\beta_0, \xi), 0, 1)$.
Actually, $b_i(\theta(\beta_o, \xi), \gamma, \delta)$ and $a_i(\theta(\beta_o, \xi), \gamma, \delta)$ are the score functions under the null model, they are, respectively,

$$b_i(\hat{\theta}(\beta_o, \xi), 0, 1) = S_{\gamma,\delta}(\hat{\theta}(\beta_0, \xi), 0, 1),$$
$$a_i(\hat{\theta}(\beta_o, \xi), 0, 1) = \frac{\partial \ell\ell(\hat{\theta}(\beta_0, \xi), 0, 1)}{\partial \theta}.$$

And $I^*_{\theta\gamma\delta}$ and $I_\theta$ can be acquired can be obtained from the fisher information matrix,

$$I^*_{\gamma\delta\theta} = \begin{bmatrix} I_{\gamma\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) \\ I_{\delta\theta}(\hat{\theta}(\beta_0, \xi), 0, 1) \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^n \left\{ \frac{-\hat{\mu_i}^2 \exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})} \right\} \\ \sum_{i=1}^n \left\{ \frac{\hat{\mu_i}^2 \exp(-\hat{\mu_i})}{1-\exp(-\hat{\mu_i})} \sum_{i=1}^p s_j(x_{ij}) \right\} \end{bmatrix},$$

$$I_\theta = I_\theta(\hat{\theta}(\beta_0, \xi), 0, 1).$$

Then we randomly generate $\{\varepsilon_1^{(b)}, \cdots, \varepsilon_n^{(b)}\}$ independently from standard normal distribution, where superscript $(b)$ stands for replications, $b$=1, $\cdots$ , $B$. Given the realizations of the data, $\{y_i, x_i\}_{i=1}^n$, and values of $\gamma = 0$, $\delta = 1$, we calculate the statistic $U_n^{(b)}(\hat{\theta}(\beta_0, \xi), 0, 1) = \sum_{i=1}^n c_i * \varepsilon_i^{(b)}$, where $\hat{\boldsymbol{\theta}}$ is the maximum penalized likelihood estimator of $\theta$ under the null model. Then we calculate the proposed score statistic for artificial observations

$$S^{(b)}_{\boldsymbol{\omega}_n} = (U_n^{(b)}(\hat{\theta}(\beta_0, \xi), 0, 1))' * \Lambda^{-1} * U_n^{(b)}(\hat{\theta}(\beta_0, \xi), 0, 1), \tag{3.4}$$

By repeatedly generating the normal variates $\{\varepsilon_1, \cdots, \varepsilon_n\}$ for $B$ times, and repeating the above procedure for each generated sample, we obtain the empirical distribution of $S_{\boldsymbol{\omega}}^{(b)}$, $b$=$1, \cdots, B$. The asymptotical $p$-value of the test is the proportion of times the artificial score statistics which are greater than or equal to the observed test statistic $S_{\boldsymbol{O}}$ given the generated data $\{y_i, x_i\}_{i=1}^n$. Then $p$-value=$B^{-1} \sum_{b=1}^{B} \mathbf{1}\{S_{\boldsymbol{\omega}}^{(b)} \geqslant S_{\boldsymbol{O}}\}$.

# Chapter 4

# Numerical study

## 4.1 Simulation

The simulation study is aimed to evaluate the empirical performance of the score test under COZIGAM. We assess the performances of our proposed score test to those of the score tests proposed by Jansakul and Hinde (2002) and Todem et al. (2012). In our simulations, data are generated from a mixture model with true mixing weights $\omega_i^*$ and a Poisson distribution with two different forms of true mean: one depends on covariates through regression technique, $\mu_i^*$=exp$(0.5 - 0.25x_i)$, where $x_i$ is a covariate generated from a uniform distribution on the interval(0,1); the other one depends on smooth functions of covariates, $\mu_i^*$=exp$(0.5 - 0.3m(x_i))$, where $m(x_i)$=$(0.2x_i^{11}(10(1 - x_i))^6 + 10(10x_i)^3(1 - x_i)^{10})/8$. The score test of Jansakul and Hinde (2002) assumed that $\omega_i$=$\gamma_0 + \gamma_1 x_i$, and that of Todem et al. (2012) assumed that $\omega_i$=$(\pi_i - \exp(-\mu_i))(1 - \exp(-\mu_i))^{-1}$ and $\pi_i$=exp$(-\exp(\gamma_0 + \gamma_1 x_i))$ under the alternative hypothesis. For our proposed score test under COZIGAM, we assume that $\omega_i$=$(\pi_i - \exp(-\mu_i))(1 - \exp(\mu_i))^{-1}$ under the alternative, where $\pi_i$=exp$\{-\exp\{\beta_0 - \gamma + \delta \sum_{j=1}^{p} s_j(x_{ij})\}\}$ and $\mu_i$=exp$(\beta_0 + \sum_{j=1}^{p} s_j(x_{ij}))$. With the assumption and parameterizations, the null hypotheses to be evaluated are given by: $H_0$: $\alpha_0$=$\beta_0$ and $\delta$=1 for our test;

$H_0$: $\gamma_j$=0, $j$=0, 1, for the test of Jansakul and Hinde (2002); and $H_0$: $\gamma_j$=$\beta_j$, $j$=0, 1, for the test of Todem et al. (2012). For each simulation, we have 1000 replicates for sample size 50, 100, 200 and 400.

**Table 4.1**: *Empirical sizes and powers of score test statistics for different forms of* $\omega_i^*$ *with Poisson mean* $\mu_i^* = \exp(0.5 - 0.25x_i)$*, at 5 % significant level.*

| | $n$=50 | | | $n$=100 | | | $n$=200 | | | $n$=400 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\omega^*$ | JH | TH | GAM | JH | TH | GAM | JH | TH | GAM | JH | TH | GAM |
| $\omega^*$=0 | 0.056 | 0.056 | 0.051 | 0.049 | 0.050 | 0.043 | 0.050 | 0.054 | 0.044 | 0.049 | 0.051 | 0.044 |
| $\omega^*$=0.15 | 0.150 | 0.152 | 0.108 | 0.265 | 0.273 | 0.200 | 0.601 | 0.559 | 0.448 | 0.890 | 0.889 | 0.698 |
| $\omega^*$=$-0.1 + 0.15x_i$ | 0.069 | 0.071 | 0.061 | 0.134 | 0.123 | 0.101 | 0.158 | 0.152 | 0.119 | 0.287 | 0.273 | 0.158 |
| $\omega^*$=$\frac{\exp(-2+x_i)}{1+\exp(-2+x_i)}$ | 0.185 | 0.185 | 0.135 | 0.406 | 0.416 | 0.302 | 0.753 | 0.754 | 0.585 | 0.982 | 0.981 | 0.822 |
| $\omega^*$=$0.2 - 0.25m(x_i)$ | 0.079 | 0.088 | 0.072 | 0.151 | 0.163 | 0.121 | 0.276 | 0.291 | 0.246 | 0.526 | 0.555 | 0.488 |
| $\omega^*$=$\frac{\exp(-2+1.5m(x_i))}{1+\exp(-2+1.5m(x_i))}$ | 0.265 | 0.264 | 0.204 | 0.568 | 0.580 | 0.476 | 0.898 | 0.895 | 0.741 | 0.999 | 0.999 | 0.905 |

Note: 1. $x_i$ a covariate taking on $n$ uniformly distributed values on (0,1), $m(x_i)$=$(0.2x_i^{11}(10(1-x_i))^6 + 10(10x_i)^3(1-x_i)^{10})/8$; 2. JH stands for the score test of Jansakul and Hinde, TH stands for the score test of Todem et al., GAM stands for the score test under COZIGAM; 3. For each simulation, we have 1000 replicates.

**Table 4.2**: *Empirical sizes and powers of score test statistics for different forms of* $\omega_i^*$ *with Poisson mean* $\mu_i^* = \exp(0.5 - 0.3m(x_i))$*, at 5 % significant level.*

| | $n$=50 | | | $n$=100 | | | $n$=200 | | | $n$=400 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\omega^*$ | JH | TH | GAM | JH | TH | GAM | JH | TH | GAM | JH | TH | GAM |
| $\omega^*$=0 | 0.048 | 0.053 | 0.047 | 0.047 | 0.043 | 0.045 | 0.048 | 0.049 | 0.047 | 0.048 | 0.046 | 0.047 |
| $\omega^*$=0.15 | 0.121 | 0.125 | 0.089 | 0.295 | 0.302 | 0.214 | 0.586 | 0.586 | 0.434 | 0.897 | 0.902 | 0.736 |
| $\omega^*$=$0.25 - 0.1x_i$ | 0.229 | 0.230 | 0.179 | 0.475 | 0.470 | 0.358 | 0.816 | 0.822 | 0.678 | 0.987 | 0.989 | 0.887 |
| $\omega^*$=$\frac{\exp(-1.5+0.5x_i)}{1+\exp(-1.5+0.5x_i)}$ | 0.265 | 0.259 | 0.176 | 0.527 | 0.534 | 0.410 | 0.896 | 0.897 | 0.720 | 0.999 | 0.998 | 0.866 |
| $\omega^*$=$-0.15 + 0.25m(x_i)$ | 0.128 | 0.124 | 0.137 | 0.199 | 0.177 | 0.222 | 0.356 | 0.306 | 0.467 | 0.622 | 0.522 | 0.810 |
| $\omega^*$=$\frac{\pi_i^* - \exp(-\mu_i^*)}{1-\exp(-\mu_i^*)}$, $\pi_i^*$=$\exp(-\exp(1.2 - 2m(x_i))$ | 0.300 | 0.264 | 0.402 | 0.590 | 0.471 | 0.766 | 0.886 | 0.784 | 0.966 | 0.995 | 0.985 | 1.000 |

Note: 1. $x_i$ a covariate taking on $n$ uniformly distributed values on (0,1), $m(x_i)$=$(0.2x_i^{11}(10(1-x_i))^6 + 10(10x_i)^3(1-x_i)^{10})/8$; 2. JH stands for the score test of Jansakul and Hinde, TH stands for the score test of Todem et al., GAM stands for the score test under COZIGAM; 3. For each simulation, we have 1000 replicates.

## 4.2 Findings from simulation

Firstly, the three tests have controlled type I error rates well (Table 4.1 and Table 4.2). In Table 4.1, the true Poisson means depend on covariates through regression technique. The results demonstrate that no matter whether the true mixing weight is constant, a linear form of covariate, or smooth function of covariate, our proposed score test loses some efficiency, compared to the other two tests.

However, in Table 4.2, the true Poisson mean depends on smooth functions of covariate. It is clear that our proposed test outperforms the other two tests when the true mixing weights are, $\omega_i^*$=$-0.15 + 0.25m(x_i)$ and $\omega_i^*$=$\frac{\pi_i^* - \exp(-\mu_i^*)}{1-\exp(-\mu_i^*)}$, where $\pi_i^*$=exp(exp$(1.2 - 2m(x_i))$. This is expected as data were generated under the situation where the ture mixing weights and the true Poisson means involve smooth functions of covariate.

Finally, incorporating smooth functions can improve the performances of the score test. Our proposed approach is indeed more powerful in detecting heterogeneity in the population when nonlinear covariates effects exist in both the Poisson mean and the mixing weight. Besides, our proposed approach loses some efficiency when the Poisson mean or the mixing weight truly depends on a linear function of covariates, on the link scales. However, the true model is always unknown to the analyst, it is a more conservative strategy to use our proposed score test to conduct inference for the mixing weight.

## 4.3 Application to Recreational Fisheries data

National Oceanic and Atmospheric Administration (NOAA) have conducted several fishing surveys since 2004. The main goal of these surveys is working with both commercial and recreational fishermen to count what's being caught, when, where, and how. They mainly use the collected information to decide how many fish can be taken recreationally and commercially without having negative effect on the sustainability of individual fisheries.

**Table 4.3**: *Observed score test statistics and the associated p-values for heterogeneity in recreational fisheries data*

| | Response | |
|---|---|---|
| Methodology | test statistic | p-vlaue |
| Test of Jansakul and Hinde | 28.6527 | 0.016 |
| Test of Todem et al. | 28.7641 | 0.017 |
| Test under COZIGAM | 22.2899 | 0.009 |

The information also ensures appropriate measures are taken to recover fisheries in trouble.

To illustrate our methodology, we used fisheries data collected during July and August of 2013. The primary count outcome is the number of fish caught per hour per individual (NFPHPI). Age of the angler is considered as the covariate. After looking at the original data, we can observe many zeros in the data (see Figure 4.1). This implies that there may exist extra zeros. We evaluated the homogeneity hypothesis using the proposed score test under ZIGAM with Poisson, given the evidence from Figure 4.2 that there is nonlinear relationship between the age of angler (year) and the predictor in the Poisson mean on the log link scale. The nondegenerate distribution is a standard Poisson regression model with mean $\mu_i$=exp($\beta_0 + s(Age)$) and the mixing weight $\omega_i$ is given by equation (3.2) with the quantity $\pi_i$=exp($-\exp(\beta_0 - \gamma + \delta s(Age))$). The score test of Jansakul and Hinde (2002) and that proposed by Todem et al. (2012) were also conducted. With the above parameterizations, the null hypotheses to be evaluated become: $H_0$: $\gamma$=0, and $\delta$=1, for our score test; $H_0$: $\gamma_j$=0, $j$=0, 1, for Jansakul and Hinde's test; $H_0$: $\gamma_j$=$\beta_j$, $j$=0, 1, for test of Todem et al. The first two tests were conducted by replacing the nuisance parameter $\beta$ by its maximum likelihood estimate under the null distribution, while our proposed test was conducted by replacing the nuisance parameter $\theta$ by its maximum penalized likelihood estimate under the null distribution. Results of this analysis are given in Table 4.3.

The results in Table 4.3 show that all the three tests reject the homogeneity hypothesis at 5 % significance level. But our proposed test is more powerful to detect the heterogeneity

in the data than the other two tests.

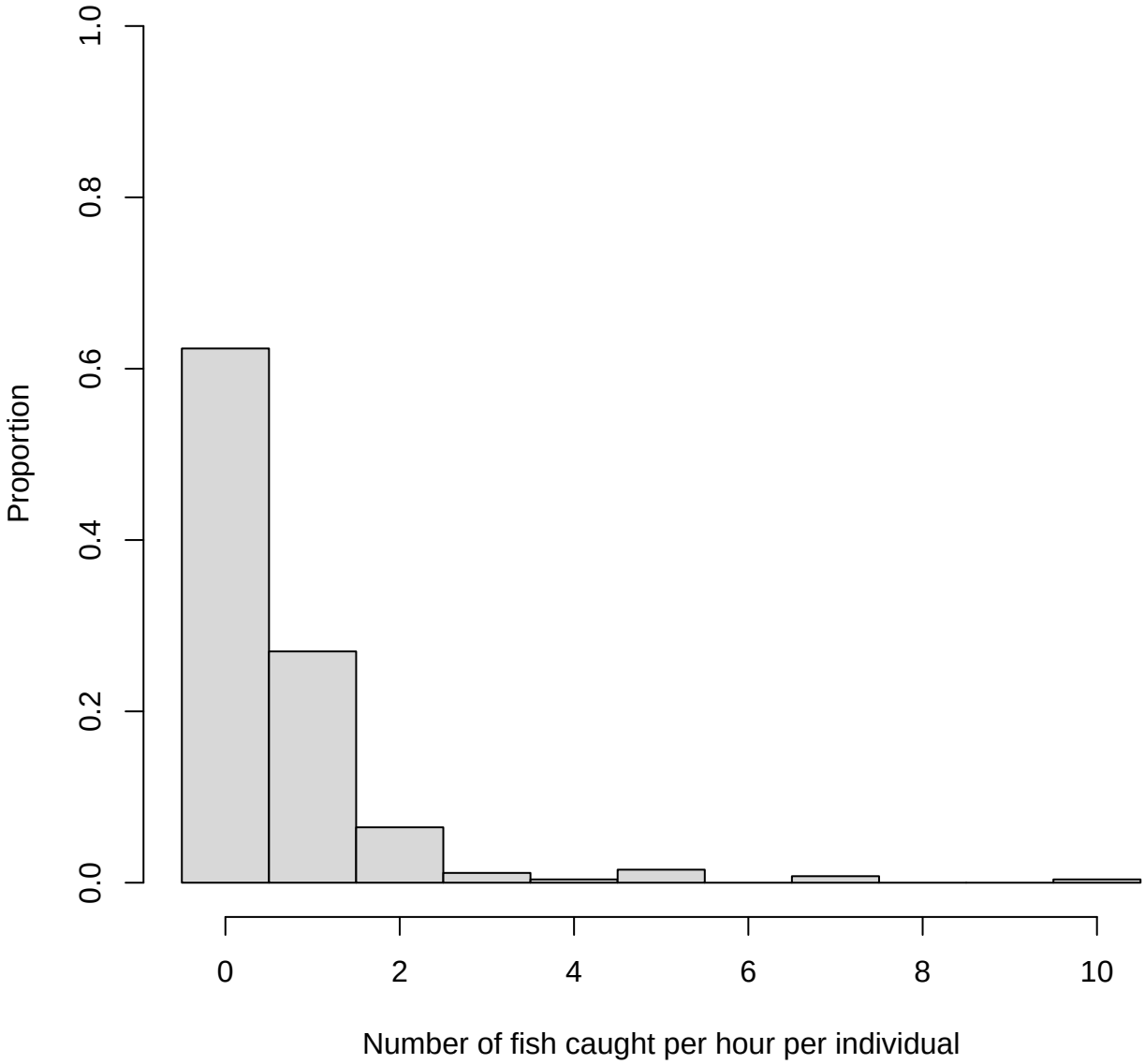


**Figure 4.1**: *Observed proportion of the number of fish caught per hour per individual.*

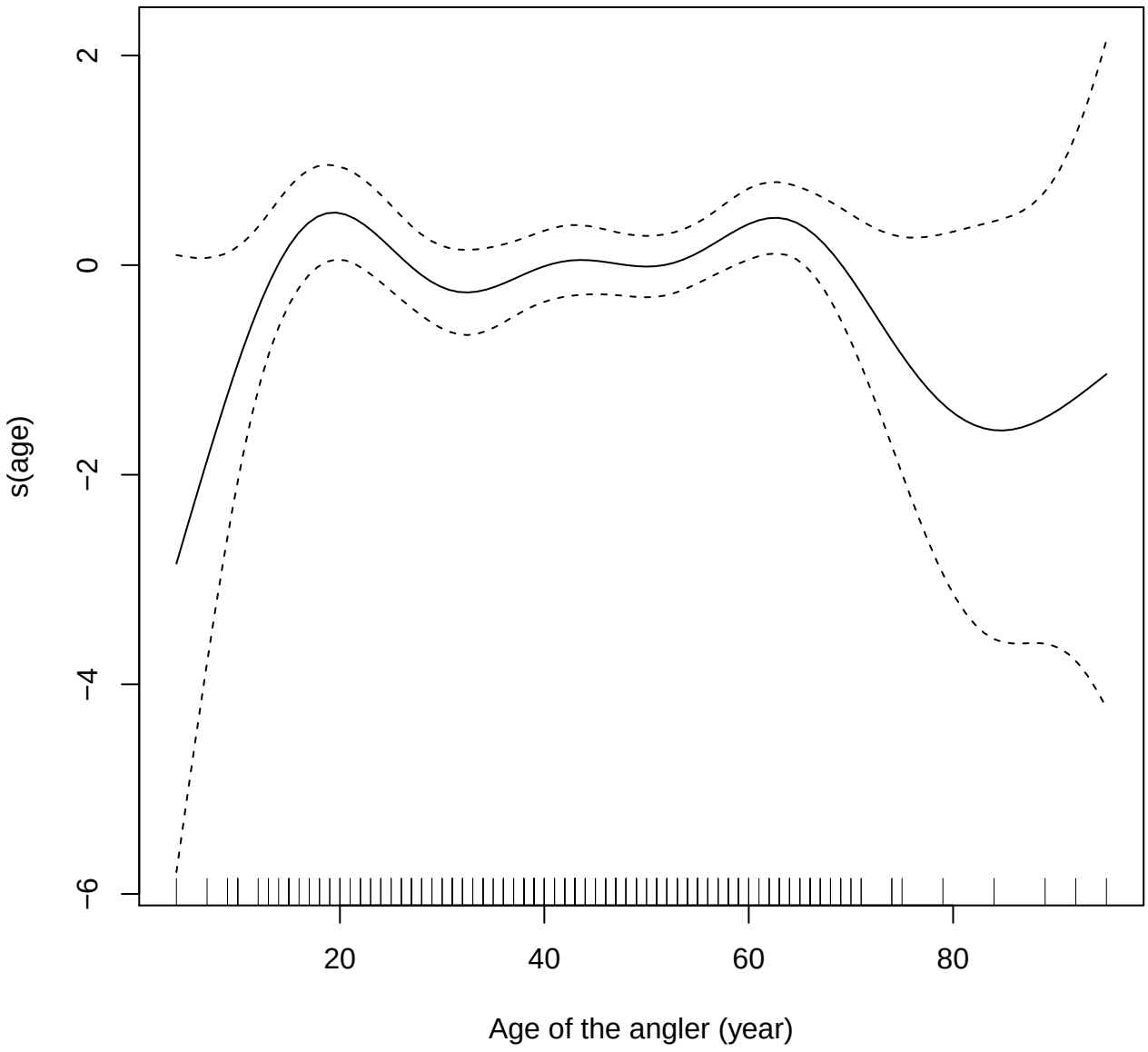


**Figure 4.2**: *Plot of the smooth function components of the Poisson mean, on the log scale, of the fitted ZIGAM with the recreational fisheries data.*

# Chapter 5

# Discussion

In this report, we proposed a generalized score test to evaluate the mixing weight under zero-inflated generalized additive models. Simulation studies indicate that our proposed test loses some efficiency compared with the tests of Jansakul (2002) and Todem et al. (2012) when the true Poisson mean depends on a linear form of covariates. However, if both the Poisson mean and the mixing weight truly involve smooth functions of covariates, our proposed test outperforms the other tests. Because the true model is always unknown to the analyst, we suggest that it is a conservative strategy to evaluate the mixing weight with our proposed score test.

It is worth nothing that, Wald test will be a good candidate to evaluate whether the mixing weight equals to zero under COZIGAM if the alternative model can be fitted in routine. In the literature, the R package "COZIGAM" was developed to fit the alternative model, but it has been removed from the CRAN list in R software due to its non-stability.

Furthermore, it is also worthwhile to extend our approach to analyze the longitudinal/-correlated data using random effects models or generalized estimating equations approach. These are actually the subjects of future research.

# Bibliography


[1] Lambert D. Zero-inflated poisson regression, with an application to defects in manufacturing. *Technometrics*, 34(1), 1992.

[2] Lam K. F., Xue H. Q., and Cheung Y. B. Semiparametric analysis of zero-inflated count data. *Biometrics*, 62:996–1003, 2006.

[3] Liu H. and Chan K. S. Generalized additive models for zero-inflated data with partial constraints. *Scandinavian Journal of Statistics*, 38:650–665, 2011.

[4] Barryand S. C. and Welsh A. H. Generalized additive modelling and zero inflated count data. *Ecological Modelling*, 157:179–188, 2002.

[5] Ma S., Liu A., Carr J., Post W., and Kronmal R. Statistical modeling of agatston score in multi-ethnic study of atherosclerosis (mesa). *PLoS ONE*, 5:e12036, 2010.

[6] Jansakul N. and Hinde J. P. Score tests for zero-inflated poisson models. *Computational Statistics & Data Analysis*, 40:75–96, 2002.

[7] Todem D., Hsu W-W., and Kim K. M. On the efficiency of score tests for homogeneity in two-component parametric models for discrete data. *Biometrics*, 68:975–982, 2012.

[8] Lin D. Y., Fleming T. R., and Wei L. J. Confidence bands for sruvival curves under the proportional hazards model. *Biometrika*, 81:73–81, 1994.

[9] Dietz E. and Böhning D. On estimation of the poisson parameter in zero-modified poisson models. *Computational Statistics and Data Analysis*, 34:441–459, 2000.

[10] Hastie T. and Tibshirani R. Generalized additive models. *Statistical Science*, 1:297–318, 1986.

[11] Wood S. N. *Generalized Additive Models, An Introduction with R.*, volume 1st Ed. London: Chapman and Hall, 2006.

[12] Green P. J. Penalized likelihood for general semi-parametric regression models. *International Statistical Review*, 55:245–259, 1987.

[13] Liu H., Ma S., Kronmal R., and Chan K. S. Semiparametric zero-inflated modeling in multi-ethnic study of atherosclerosis (mesa). *The Annals of Applied Statistics*, 6(3): 1236–1255, 2012.

[14] Wood S. N. Modelling and smoothing parameter estimation with multiple quadratic penalties. *J. R. Statist. Soc. B*, 62:413–428, 2000.

[15] Zhu Z. Y. and Zhang H. Spatial sampling design under the in?ll asymptotic framework. *Enrironmetrics*, 17:323–337, 2006.

[16] Efron B. and Tibshirani R. J. *An Introduction to the Bootstrap*. New York: Chapman and Hall, 1993.

# Appendix A

# Second derivative of the Penalized likelihood function

$$\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\theta\partial\gamma'} = \begin{bmatrix} \sum_{i=1}^{n}\left\{ -I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right\} \\ \sum_{i=1}^{n}\left\{ \left[-I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\sum_{j=1}^{p} s_{j1}(x_{ij})\right\} \\ \vdots \\ \sum_{i=1}^{n}\left\{ \left[-I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\sum_{j=1}^{p} s_{jQ}(x_{ij})\right\} \end{bmatrix},$$

$$\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\theta\partial\delta'} = \begin{bmatrix} \sum_{i=1}^{n}\left\{ \left[-I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\sum_{j=1}^{p} s_{j}(x_{ij})\right\} \\ \sum_{i=1}^{n}\left\{ \left[-I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\sum_{j=1}^{p} s_{j}(x_{ij})\sum_{j=1}^{p} s_{j1}(x_{ij})\right\} \\ \vdots \\ \sum_{i=1}^{n}\left\{ \left[-I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\sum_{j=1}^{p} s_{j}(x_{ij})\sum_{j=1}^{p} s_{jQ}(x_{ij})\right\} \end{bmatrix}$$

$$\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\gamma\partial\gamma'} = \sum_{i=1}^{n}\left\{ I_{(y_i=0)}\log\pi_i - I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2+\pi_i\log\pi_i-{\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right\},$$

$$\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\gamma\partial\delta'} = \sum_{i=1}^{n}\left\{\left[-I_{(y_i=0)}\log\pi_i + I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2 + \pi_i\log\pi_i - {\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\sum_{j=1}^{p} s_j(x_{ij})\right\},$$

$$\frac{\partial^2 l(\theta(\beta_0,\xi),\gamma,\delta)}{\partial\delta\partial\delta'} = \sum_{i=1}^{n}\left\{\left[I_{(y_i=0)}\log\pi_i - I_{(y_i>0)}\left(\frac{\pi_i(\log\pi_i)^2 + \pi_i\log\pi_i - {\pi_i}^2\log\pi_i}{(1-\pi_i)^2}\right)\right]\left(\sum_{j=1}^{p} s_j(x_{ij})\right)^2\right\}.$$

# Appendix B

# Code

```
rm(list=ls(all=TRUE));library(mgcv);library(boot);
#################################################################
N1=1000; ## MC samples
N2=1000; ## resampling samples
a0=0.5;a1=-0.25; ## true parameters of Poisson mean depending on x3
#a0=0.5;a1=-0.3; ## true parameters of Poisson mean depending on s1
iteration=100; ## show the progress every xxx interations.
#################################################################
for (n in c(400,200,100,50)){
ptm=Sys.time();
S1=numeric(0)
S2=numeric(0)
S3=numeric(0)
for(mc in 1:N1)
{
#################################################################
##Generating Poisson data(sample size n)
i=1
seq0=numeric(0)
while(i<=n)
{
j=0
seq01=numeric(0)
```

```
x3=runif(1,0,1)
s1=(0.2*x3^11*(10*(1-x3))^6+10*(10*x3)^3*(1-x3)^10)/8
u=exp(a0+a1*x3);#u=exp(a0+a1*s1);
p=0
x=runif(1,0,1)
cp=p+(1-p)*exp(-u)*(u^(0))/(factorial(0))
while(cp<=x)
{
py=(1-p)*exp(-u)*(u^(j+1))/(factorial(j+1))
cp=cp+py
j=j+1
}
seq01=c(seq01,j,u,x3,i)
z=c(seq01)
seq0=c(seq0,z)
i=i+1
mat=matrix(seq0,4)
mat=t(mat)
}##end of generating Poisson data
y=mat[,1]
x3=mat[,3]
mmat=data.frame(y,x3);
#########################################################
#score test for GAM
fit1=gam(y~s(x3),family="poisson")
x1=predict(fit1,type="terms")
ss=predict(fit1,type="lpmatrix")
lambdahat=(fit1$sp)^2;
S=fit1$sm[[1]][12][[1]][[1]]
coefnew1=matrix(c(fit1$coef[1],1),nrow=2,ncol=1)
X1=matrix(c(matrix(1,n,1),x1),nrow=n,ncol=2)
muhat1=exp(X1%*%coefnew1)
#score test statistic
I00=as.matrix(vcov(fit1))
I01=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1)))
I02=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*x1)
I11=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,2])
```

```
I12=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,2]*x1))
I21=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,3])
I22=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,3]*x1))
I31=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,4])
I32=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,4]*x1))
I41=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,5])
I42=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,5]*x1))
I51=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,6])
I52=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,6]*x1))
I61=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,7])
I62=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,7]*x1))
I71=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,8])
I72=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,8]*x1))
I81=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,9])
I82=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,9]*x1))
I91=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*ss[,10])
I92=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*(ss[,10]*x1))
I222=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1)))
I232=sum(-muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*x1)
I322=t(I232)
I332=sum(muhat1^2*exp(-muhat1)/(1-exp(-muhat1))*x1*x1)
sgama1=sum(muhat1*((y==0)*1)-((y>0)*1)*(muhat1*exp(-muhat1)/(1-exp(-muhat1))))
sdelta1=sum((-muhat1*((y==0)*1)+((y>0)*1)*(muhat1*exp(-muhat1)/(1-exp(-muhat1))))*x1)
Score1=matrix(c(sgama1,sdelta1),nrow=2,ncol=1)
C11=matrix(c(I222,I232,I322,I332),nrow=2,ncol=2)
C12=matrix(c(I01,I02,I11,I12,I21,I22,I31,I32,I41,I42,I51,I52,I61,I62,I71,I72,I81,I82,I91,I92),
nrow=2,ncol=10)
C13=t(C12)
C1=C11-C12%*%I00%*%C13
sts1=t(Score1)%*%solve(C1)%*%Score1
sts1=c(sts1)
##sts1 is the observed score statistic for Gam.
################################################################################
##Resampling for GAM
#resampling set up
scorea10=-muhat1+y
scorea11=(-muhat1+y)*ss[,2]-matrix(lambdahat*S[1,]%*%t(t(fit1$coef[2:10])),n,1)
```

```
scorea12=(-muhat1+y)*ss[,3]-matrix(lambdahat*S[2,]%*%t(t(fit1$coef[2:10])),n,1)
scorea13=(-muhat1+y)*ss[,4]-matrix(lambdahat*S[3,]%*%t(t(fit1$coef[2:10])),n,1)
scorea14=(-muhat1+y)*ss[,5]-matrix(lambdahat*S[4,]%*%t(t(fit1$coef[2:10])),n,1)
scorea15=(-muhat1+y)*ss[,6]-matrix(lambdahat*S[5,]%*%t(t(fit1$coef[2:10])),n,1)
scorea16=(-muhat1+y)*ss[,7]-matrix(lambdahat*S[6,]%*%t(t(fit1$coef[2:10])),n,1)
scorea17=(-muhat1+y)*ss[,8]-matrix(lambdahat*S[7,]%*%t(t(fit1$coef[2:10])),n,1)
scorea18=(-muhat1+y)*ss[,9]-matrix(lambdahat*S[8,]%*%t(t(fit1$coef[2:10])),n,1)
scorea19=(-muhat1+y)*ss[,10]-matrix(lambdahat*S[9,]%*%t(t(fit1$coef[2:10])),n,1)
scorea1=cbind(scorea10,scorea11,scorea12,scorea13,scorea14,scorea15,scorea16,scorea17,
scorea18,scorea19)
scorea1=t(scorea1)
scorew1=cbind(X1[,1]*(((y==0)*1)*muhat1-((y>0)*1)*((muhat1*exp(-muhat1))/(1-exp(-muhat1)))),
X1[,2]*(-((y==0)*1)*muhat1+((y>0)*1)*((muhat1*exp(-muhat1))/(1-exp(-muhat1)))))
scorew1=t(scorew1)
Iwa1=C12
Ia1=I00
ci1=scorew1-Iwa1%*%Ia1%*%t(scorea1)
##resampling
seq1=matrix(,1,N2)
for (k in 1:N2)
{
e1=matrix(rnorm(n,0,1),n,1)
u1=ci1%*%e1
sb1=t(u1)%*%solve(C1)%*%u1
sb1=c(sb1)
seq1[1,k]=sb1
}
p1=mean((seq1>sts1)*1)
E1=(p1<0.05)*1
S1=c(S1,E1)
##end of resampling for GAM and S1 are the resampling score statistcs for Gam
############################################################
#score test statistic of JH
#find the MLE of beta under standard poisson.
fit2=glm(y~x3, family="poisson");
coef2=fit2$coef;
X2=matrix(c(matrix(1,n,1),x3),nrow=n,ncol=2);
```

```
G2=matrix(c(matrix(1,n,1),x3),nrow=n,ncol=2);
muhat2=exp(X2%*%coef2);
D2=diag(c(muhat2));
#score test statistic
score2=t(G2)%*%(((y==0)*1-exp(-muhat2))/exp(-muhat2));
score21=t(X2)%*%(-((y==0)*1)*muhat2+((y>0)*1)*(y-muhat2));
f11=t(X2)%*%D2%*%X2;
f22=t(G2)%*%diag(c(((1-exp(-muhat2))/exp(-muhat2))))%*%G2;
f21=t(G2)%*%diag(c(-muhat2))%*%X2;
f12=t(f21);
C2=f22-f21%*%solve(f11)%*%f12;
sts2=t(score2)%*%solve(C2)%*%score2;
sts2=c(sts2);
##sts2 is the observed score statistic for JH
############################################################
#resampling for Jansakul
#resampling set up
scorea2=cbind(X2[,1]*(-muhat2+((y>0)*1)*y),X2[,2]*(-muhat2+((y>0)*1)*y))
scorea2=t(scorea2)
scorew2=cbind(G2[,1]*((y==0)*1-exp(-muhat2))/exp(-muhat2),
G2[,2]*((y==0)*1-exp(-muhat2))/exp(-muhat2))
scorew2=t(scorew2)
Iwa2=t(G2)%*%diag(c(-muhat2))%*%X2
Ia2=t(G2)%*%diag(c(muhat2))%*%X2
ci2=scorew2-Iwa2%*%solve(Ia2)%*%scorea2
##Resampling
seq2=matrix(,1,N2)
for(m in 1:N2)
{
e2=matrix(rnorm(n,0,1),n,1)
u2=ci2%*%e2
sb2=t(u2)%*%solve(C2)%*%u2
sb2=c(sb2)
seq2[1,m]=sb2
}
p2=mean((seq2>sts2)*1)
E2=(p2<0.05)*1
```

```
S2=c(S2,E2)
#############################################################
# TH's score test statistic
##find the MLE of beta under standard poisson.
fit3=glm(y~x3, family="poisson", data=mmat)
coef3=fit3$coef
X3=matrix(c(matrix(1,n,1),x3),nrow=n,ncol=2)
muhat3=exp(X3%*%coef3)
##score test statistic of TH
D3=diag(c(muhat3))
score3=t(X3)%*%(muhat3*((((y==0)*1)-exp(-muhat3))/(1-exp(-muhat3))));
score31=t(X3)%*%(-muhat3+((y>0)*1)*y)
H11=t(X3)%*%D3%*%X3;
H22=t(X3)%*%diag(c(((muhat3)^2)*exp(-muhat3)/(1-exp(-muhat3))))%*%X3;
H12=t(X3)%*%diag(c(-((muhat3)^2)*exp(-muhat3)/(1-exp(-muhat3))))%*%X3;
H21=t(H12);
C3=H22-H21%*%solve(H11)%*%H12;
sts3=t(score3)%*%solve(C3)%*%score3;
sts3=c(sts3)
##sts3 is the observed score statistic for TH.
#############################################################
#resampling for TH
scorea3=cbind(X3[,1]*(-muhat3+((y>0)*1)*y),X3[,2]*(-muhat3+((y>0)*1)*y))
scorea3=t(scorea3)
scorew3=cbind(X3[,1]*muhat3*((((y==0)*1)-exp(-muhat3))/(1-exp(-muhat3))),
X3[,2]*muhat3*((((y==0)*1)-exp(-muhat3))/(1-exp(-muhat3))))
scorew3=t(scorew3)
Iwa3=t(X3)%*%diag(c(-muhat3^2*exp(-muhat3)/(1-exp(-muhat3))))%*%X3
Ia3=t(X3)%*%D3%*%X3
ci3=scorew3-Iwa3%*%solve(Ia3)%*%scorea3
seq3=matrix(,1,N2)
for(t in 1:N2)
{
e3=matrix(rnorm(n,0,1),n,1)
u3=ci3%*%e3
sb3=t(u3)%*%solve(C3)%*%u3
sb3=c(sb3)
```

```
seq3[1,t]=sb3
}
p3=mean((seq3>sts3)*1)
E3=(p3<0.05)*1
S3=c(S3,E3)
if (m%%iteration==0){cat("iteration = ", mc,"of",N1, "\n")};
################################################
}##end of MC
gam_r=mean(S1);
jh_r=mean(S2);
th_r=mean(S3);
duration=(Sys.time()-ptm);
cat("########################################","\n");
cat("Sample size=",n,"\n");
cat("#######  Resampling  ######","\n");
cat("GAM=",gam_r,"\n");
cat("JH=",jh_r,"\n");
cat("TH=",th_r,"\n");
print(duration);
cat("########################################","\n");
}
```