VARIABLE SELECTION IN GENERALIZED ADDITIVE MIXED MODELS

Sezer, Ahmet
Anadolu University
a.sezer@anadolu.edu.tr

Toyganozu, Cuneyt
Suleyman Demirel University
cuneyttoyganozu@sdu.edu.tr

Abstract

Identifying the subset of the important variables is of special importance in multivariate regression. In this study we are interested in selecting significant covariates in semiparametric mixed modelling. Variable selection procedure considers both nonparametric and parametric component. We approximate nonparametric component by smoothing splines and minimize the sum of squared errors subject to an additive penalty of spline functions. We propose stepwise selection procedures for generalized additive models using penalized quasi-likelihood.

Keywords: Generalized linear mixed model, semi-parametric models, penalized quasi-likelihood.

Introduction

Generalized linear mixed models (GLMMs) (Breslow and Clayton, 1993) are widely used to analyse clustered data such as longitudinal and financial data. Lin and Zhang (1999) proposed generalized additive mixed models (GAMMs) that allow for flexible modeling of the covariate effects by replacing the linear predictor in GLMMs with an additive combination of nonparametric functions of covariates and random effects. Semiparametric models are good compromises and retain nice features of both the parametric and nonparametric models.

Clustered data arise frequently in epidemiology and clinical trials. Each subject in a longitudinal epidemiological study or each hospital in a multi-center clinical trial may be viewed as a cluster. The challenge in analyzing clustered data is that the data within a cluster tend to be correlated. A common way to account for this feature is to use cluster-specific random effects to model the correlation explicitly in a generalized linear mixed model (GLM). If the random effects are assumed to be normally distributed, likelihood inference procedure can be carried out using a Monte Carlo approach or numerical integration (Zeger and Karim (1991), Booth and Hobert (1999)). Likelihood inference may not be feasible when the random effects

structure is complex. However penalized quasi-likelihood can be used to overcome this diffuculty.

Cox and Kohn (1989) derived a test statistic for testing the adequacy of polynomial regression based on the smoothing spline formulation of the nonparametric function. Härdle *et al.* (1998) proposed a likelihood-ratio-based test using bootstrap to compare parametric generalized linear models with semiparametric generalized partial linear models. A common interest in many applications of nonparametric regression is to compare nonparametric covariate effects between two groups. Several tests were developed to test the equivalence of curves for longitudinal Gaussian data (Fan and Lin (1998) and Zhang *et al.* (2000)). Härdle, Liang and Gao (2000), Ruppert, Wand and Carroll (2003) and Yatchew (2003) presented diverse semiparametric regression models, and their inference procedures and applications. In order to select significant variables and estimate unknown regression coefficients together, Fan and Li (2001) proposed a family of variable selection procedures for parametric models via nonconcave penalized likelihood.

In this study, we are interested in how to select significant variables in the semiparametric mixed modeling. Variable selection for semiparametric regression models consists of nonparametric components and parametric portion. In practice, a number of variables are available to include in the model, but many of them may not be significant and should be excluded from the ideal model. It is common in practice to include only important variables in the model to enhance predictability and to give a parsimonious description between the response and the covariates. We extended stepwise regression to the semiparametric models by using the penalized quasi-likelihood. Nonparametric functions are estimated by using smoothing splines and jointly estimate the smoothing parameters and the variance components by using penalized quasi-likelihood.

Generalized Additive Mixed Models

Let $\mathbf{y}_i^T = (y_{i1},..., y_{iT_i})$ be response vector, where y_{it} denote observation t in cluster $i, i = 1,..., n, t = 1,..., T_i$. Let $\mathbf{x}_{it}^T = (1, x_{it1},..., x_{itp})$ be the covariate vector associated with fixed effects and $\mathbf{z}_{it}^T = (z_{it1},..., z_{itq})$ be the covariate vector associated with random effects. It is assumed that the observations y_{it} are conditionally independent with means $\mu_{it} = E(y_{it}|\mathbf{b}_i, \mathbf{x}_{it}, \mathbf{z}_{it})$ and variances $\text{var}(y_{it}|\mathbf{b}_i) = \phi v(\mu_{it})$, where $v(.)$ is a known variance function, ϕ is a scale parameter, and $\mathbf{b}_i$ is cluster-specific random effects.

The generalized semiparametric mixed model, including an additive term that depends on covariates $\mathbf{u}_{it}^T = (u_{it1},..., u_{itm})$ is given by

$$\begin{aligned} g(\mu_{it}) &= \mathbf{x}_{it}^T\boldsymbol{\beta} + \sum_{j=1}^{m} \alpha_{(j)}(u_{itj}) + \mathbf{z}_{it}^T\mathbf{b}_i \\ &= par\theta_{it} + add\theta_{it} + ran\theta_{it} \end{aligned} \quad (1)$$

where g is monotonic link function, $par\theta_{it} = \mathbf{x}_{it}^T\boldsymbol{\beta}$ is a linear parametric term, with parameter vector $\boldsymbol{\beta}^T = (\beta_0, \beta_1, ..., \beta_p)$, including the intercept, $add\theta_{it} = \sum_{j=1}^{m} \alpha_{(j)}(u_{itj})$ is an additive term with unspecified influence functions $\alpha_{(1)}, ..., \alpha_{(m)}$ and $ran\theta_{it} = \mathbf{z}_{it}^T\mathbf{b}_i$ contains the cluster-specific random effects $\mathbf{b}_i \sim N(0, \mathbf{Q})$, where $\mathbf{Q}$ is a qxq dimensional known or unknown covariance matrix.

In regression spline methodology the unknown functions $\alpha_{(j)}(.)$ are approximated by basis functions. A simple basis is known as the B-spline basis of degree d, yielding

$$\alpha_{(j)}(u) = \sum_{i=1}^{k} \alpha_i^{(j)} B_i^{(j)}(u; d),$$

where $B_i^{(j)}(u; d)$ denotes the $i-$th basis function for variable j. If the functions $\alpha_{(j)}(.)$ are strictly linear, the model reduces to the common generalized linear mixed model (GLMM). Versions of the additive model (1) have been considered by Zeger and Diggle (1994), Lin and Zhang (1999) and Zhang et al. (1998). While Lin and Zhang (1999) used natural cubic smoothing splines for the estimation of the unknown functions $\alpha_{(j)}(.)$, in this study cubic splines are used. In recent years regression splines have been widely used for the estimation of additive structures, see, for example, Marx and Eilers (1998), Wood (2004, 2006) and Wand (2000).

Let $\boldsymbol{\alpha}_j^T = (\alpha_1^{(j)}, ..., \alpha_k^{(j)})$ denote the unknown parameter vector of the $j-$ th smooth function and let $\mathbf{B}_j^T(u) = (B_1^{(j)}(u; d), ..., B_k^{(j)}(u; d))$ represent the vector-valued evaluations of the k basis functions. Then the parameterized model for (1) has the form

$$g(\mu_{it}) = \mathbf{x}_{it}^T\boldsymbol{\beta} + \mathbf{B}_1^T(u_{it1})\boldsymbol{\alpha}_1 + \cdots + \mathbf{B}_m^T(u_{itm})\boldsymbol{\alpha}_m + \mathbf{z}_{it}^T\mathbf{b}.$$

By collecting observations within one cluster, the design matrix would be $\mathbf{X}_i^T = (\mathbf{x}_{i1}, ..., \mathbf{x}_{iT_i})$ for the $i-$th covariate, and analogously it is set $\mathbf{Z}_i^T = (\mathbf{z}_{i1}, ..., \mathbf{z}_{iT_i})$, so that the model has the simpler form

$$g(\boldsymbol{\mu}_i) = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{B}_{i1}\boldsymbol{\alpha}_1 + \cdots + \mathbf{B}_{im}\boldsymbol{\alpha}_m + \mathbf{Z}_i\mathbf{b}_i,$$

where $\mathbf{B}_{ij}^T = [\mathbf{B}_j(u_{i1j}), ..., \mathbf{B}_j(u_{iT_ij})]$ denotes the transposed B-spline design matrix of the $i-$ th cluster and variable j.

Let $\mathbf{X}^T = [\mathbf{X}_1^T, ..., \mathbf{X}_n^T]$, $\mathbf{Z} = diag(\mathbf{Z}_1, ..., \mathbf{Z}_n)$ be a block-design matrix and $\mathbf{b}^T = (\mathbf{b}_1^T, ..., \mathbf{b}_n^T)$ be the vector collecting all random effects. Then the model in the matrix form would be

$$g(\boldsymbol{\mu}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{B}_1\boldsymbol{\alpha}_1 + \cdots + \mathbf{B}_m\boldsymbol{\alpha}_m + \mathbf{Z}\mathbf{b} \qquad (2)$$

with $\mathbf{B}_j^T = [\mathbf{B}_{1j}^T, ..., \mathbf{B}_{nj}^T]$ representing the transposed B-spline design matrix of the $j-$ th smooth function. The model can be written in matrix form as

$$g(\boldsymbol{\mu}) = \mathbf{X}\boldsymbol{\beta} + \mathbf{B}\boldsymbol{\alpha} + \mathbf{Z}\mathbf{b},$$

where $\boldsymbol{\alpha}^T = (\boldsymbol{\alpha}_1^T, ..., \boldsymbol{\alpha}_m^T)$ and $\mathbf{B} = [\mathbf{B}_1, ..., \mathbf{B}_m]$ (Groll and Tutz, 2012).

The Penalized Likelihood Approach

It is assumed that the conditional density of y_{it}, given explanatory variables and the random effect $\mathbf{b}_i$, is of exponential family type

$$f(y_{it}|\mathbf{x}_{it},\mathbf{u}_{it},\mathbf{b}_i) = \exp\left\{\frac{(y_{it}\eta_{it} - \kappa(\eta_{it}))}{\varphi} + c(y_{it},\varphi)\right\}, \qquad (3)$$

where $\eta_{it} = \eta(\mu_{it})$ denotes the natural parameter, $\kappa(\eta_{it})$ is a specific function corresponding to the type of exponential family, $c(.)$ the log normalization constant and ϕ the dispersion parameter.

A popular method to maximize generalized mixed models penalized quasi-likelihood (PQL), which has been suggested by Breslow and Clayton (1993), Lin and Breslow (1996) and Breslow and Lin (1995). In mixed models, it is assumed that the covariance matrix $\mathbf{Q}(\rho)$ of the random effects $\mathbf{b}_i$ may depend on an unknown parameter vector ρ which specifies the correlation. It is specified that the joint likelihood function by the parameters of the covariance structure ρ together with the dispersion parameter ϕ, which are collected in $\boldsymbol{\nu}^T = (\varphi, \boldsymbol{\rho}^T)$ and is defined the parameter vector $\boldsymbol{\delta}^T = (\boldsymbol{\beta}^T, \boldsymbol{\alpha}^T, \mathbf{b}^T)$. The corresponding log-likelihood is

$$l(\boldsymbol{\delta},\boldsymbol{\nu}) = \sum_{i=1}^{n} \log \int f(\mathbf{y}_i|\boldsymbol{\delta},\boldsymbol{\nu})\, p(\mathbf{b}_i,\boldsymbol{\nu}) d\mathbf{b}_i$$

Then the penalized log-likelihood is

$$l^{pen}(\delta,\nu) = \sum_{i=1}^{n} \log\left(\int f(\mathbf{y}_i|\delta,\nu)\, p(\mathbf{b}_i,\nu) d\mathbf{b}_i\right) - \frac{1}{2}\sum_{j=1}^{m} \lambda_j \alpha_j^T \mathbf{K}_j \alpha_j \qquad (4)$$

where $\mathbf{K}_j$ penalizes the parameters $\boldsymbol{\alpha}_j$ and λ_j are smoothing parameters which control the effect of the $j-$th penalty term. The log-likelihood (4) has also been considered by Lin and Zhang (1999) but with $\mathbf{K}_j$ referring to smooth splines. (Groll and Tutz, 2012).

PQL works within the profile likelihood concept. It is distinguished between the estimation of $\boldsymbol{\delta}$, given the plug-in estimate $\hat{\boldsymbol{\nu}}$, resulting in the profile-likelihood $l^{pen}(\boldsymbol{\delta},\hat{\boldsymbol{\nu}})$, and the estimation of $\boldsymbol{\nu}$. The PQL method for generalized additive mixed models is implemented in the *gamm* function of the R-package *mgcv* (Wood, 2006).

Algorithm For Stepwise Regression

To select significant variables, following algorithm is constructed for the stepwise regression. Begin by performing a multiple regression. If all covariates are shown as significant (P-values < α), then stop. All the variables should be in the model. If one or more of the p-values for the t-tests are low, forward stepwise regression can be used to develop the best model that contains some of the variables as follows.

STEP 1. Do simple regressions of response vs. each covariate variable individually. Select the covariate with the lowest p-value. (Suppose it is X_4.)

STEP 2. Do all possible 2-variable regressions in which one of the two variables is X_4. If none of the 2-variable regressions gives low p-values for both X_4 and the other variable -STOP - Use the model utilizing only X_4.

If one or more of the 2-variable models gives low p-values for both X_4 and the second variable, select the model with the lowest p-values. (Suppose it is the one with X_4 and X_6.) . Go to STEP 3.

STEP 3. Do all possible 3-variable regressions in which two of the three variables are X_4 and X_6. If none of the 3-variable regressions gives low p-values for each of X_4, X_6, and the other variable -STOP - Use the model utilizing only X_3 and X_5.

If one or more of the 3-variable models gives low p-values for X_4, X_6 and the third variable, select the model with the lowest p-values.

GO TO STEP 4 and continue this process.

Application

To show the stepwise regression procedure in generalized additive models we used the data from Wood (2006) produced by the *gamSim* function (see appendix). This function produced covariates that are candidate to be defined as smooth and linear function in generalized addive models. Figure 1 gives relationships betweeen response and each covariates. From this figure we can predict that x_0,x_1,x_2 are the covariates to be in the model as nonparametric form where as x_3 is the candidate to be in the lineer form. Table 1 reveals the result from stepwise regression algorithm of generalized adittive models obtained by the penalized quasi-likelihood.

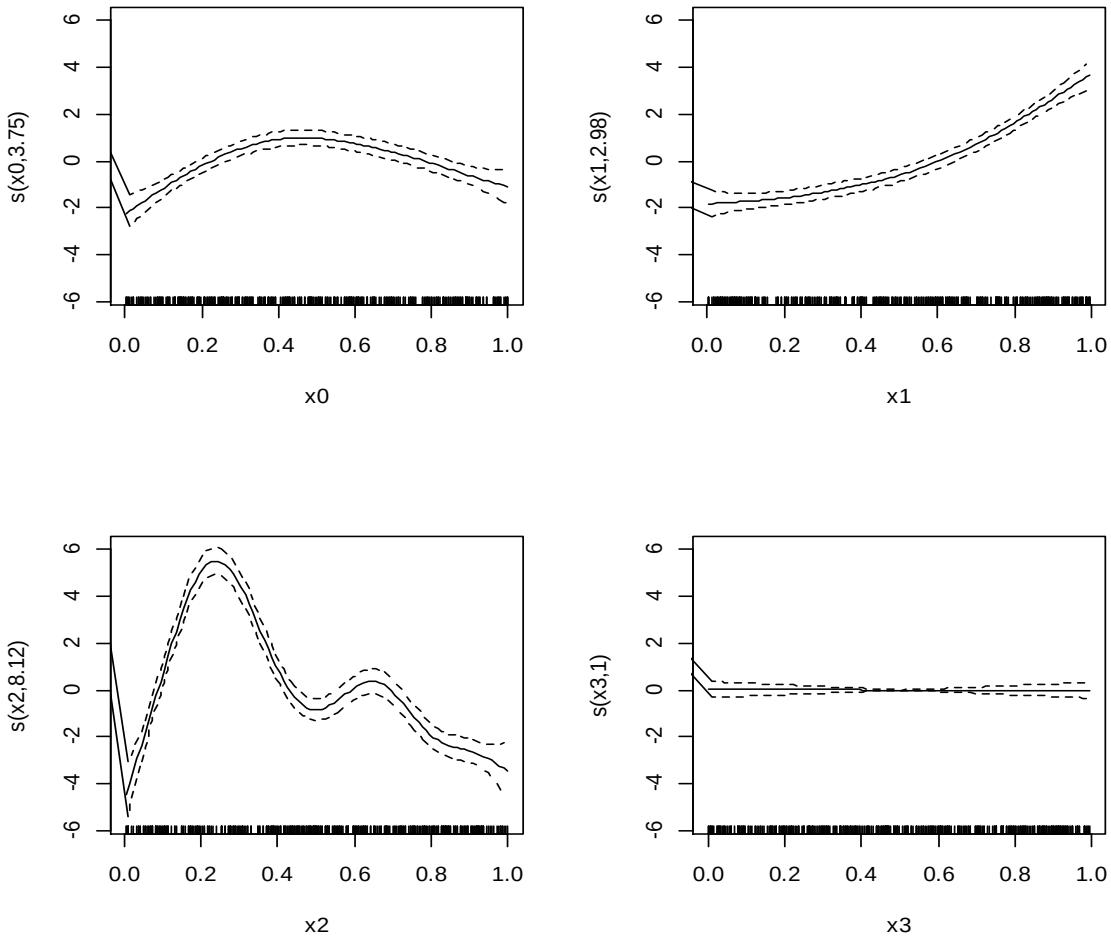

Figure 1.

Table 1. Stepwise Variable selection for Generalized Additive Models

One Variable in the model	Two Variables in the model	Three Variables in the model
X_0 → p= 0.01 X_1 → p= 0.0059 X_2 → p= 0.0009*** X_3→ p= 0.27 Note: X_2 is chosen	(X_0 / X_2)→ p= 0.54 (X_1 / X_2)→ p= 0.00 (X_3 / X_2)→ p= 0.74 Note: X_1 is chosen when X_2 is already in the model.	(X_0 / X_1, X_2)→ p= 0.56 (X_3/ X_1, X_2)→ p= 0.15 Note: X_0 and X_3 are not significant when X_1 and X_2 are already in the model.

When we enter each covariate individually, X_2 provides the lowest p value so that X_2 should be chosen at the first step. When X_2 is already in the model we add X_0, X_1, X_3 as second variabile. Since X_1 has the smallest p-value, it should join the model at the second step. At the third step, none of the covariate provides significant p-value when X_1 and X_2 are already in the model. So our best model should consist of smooth function of X_1 and X_2. Clearly, one big advantage of using Penalized Quasi likelihood is that we do not have to know the distribution of the response variable. We believe that this flexibility provides us to have many real data application in many fields.

REFERENCES

BRESLOW, N. E. AND CLAYTON, D. G. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* **88**, 9–25.

BRESLOW, N. E. AND LIN, X. (1995). Bias correction in generalized linear mixed models with a single component of dispersion. *Biometrika* **82**, 81-84

BOOTH, J. G. AND HOBERT, P. (1999) Maximizing Generalized Linear Mixed Model Likelihoods with an Automated Monte Carlo EM Algorithm. *Journal of the Royal Statistical Society* B **61**(1), 265-285.

COX, D. D. AND KOHN, E. (1989). A smoothing spline based test of model adequacy in polynomial regression. *Annals of Ins. of Stat. Math.* **41**, 383–400.

FAN, J. AND LI, R. (2001) Variable Selection Via Nonconcave Penalized Likelihood And Its Oracle Properties. *Journal of the American Statistical Association* **96**, 1348–1360.

FAN, J. AND LIN, S. (1998). Test of significance when data are curves. *Journal of the American Statistical Association* **93**, 1007–1021.

GROLL, A. AND TUTZ, G. (2012). Regularization for generalized additive mixed models by likelihood-based boosting. *Methods Inf. Med.* **51**(2), 168-77.

HÄRDLE, W., LIANG, H., AND GAO, J. T. (2000). *Partially Linear Models*. Heidelberg: Springer Physica.

HÄRDLE, W., MAMMEN, E. AND MÜLLER, M. (1998). Testing parametric versus semiparametric modeling in generalized linear models. *Journal of the American Statistical Association* **93**, 1461–1474.

LIN, X. AND BRESLOW, N. E. (1996). Bias correction in generalized linear mixed models with multiple component of dispersion. *Journal of the American Statistical Association* **91**, 1007-1016.

LIN, X. AND ZHANG, D. (1999). Inference in generalized additive mixed models by using smoothing splines. *Journal of the Royal Statistical Society* B **61**, 381–400.

MARX, D. B. AND EILERS, P. H. C. (1998). Direct generalized additive modelling with penalized likelihood. *Comp. Stat.& Data Analysis* **28**, 193-209.

YATCHEW, A. (2003). *Semiparametric Regression for the Applied Econometrician*. Cambridge University Press, Cambridge.

RUPPERT, D., WAND, M., AND CARROLL, R. (2003). *Semiparametric Regression*. Cambridge University Press, Cambridge.

WAND, M. P. (2000). A comparison of regression spline smoothing procedures. *Computational Statistics* **15**, 443-462.

WOOD, S. N. (2004). Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association* **99**, 673-686.

WOOD, S. N. (2006). *Generalized Additive Models: An Introduction with R*. London: Chapman & Hall.

ZEGER, S. L. AND DIGGLE, P. J. (1994). Semi-parametric models for longitudinal data with application to CD4 cell numbers HIV seroconverters. *Biometrics* **50**, 689-699.

ZEGER, S.L. AND KARIM, M.R (1991). Generalized linear models with random effects: A Gibbs sampling approach. *Journal of American Statistical Association* **86,** 79-86.

ZHANG, D., LIN, X., RAZ, J. AND SOWERS, M. (1998). Semi-parametric stochastic mixed models for longitudinal data. *Journal of the American Statistical Association* **93**, 710-719.

APPENDIX

The *gamSim* function;

```
function (eg = 1, n = 400, dist = "normal", scale = 2)
{
   if (eg == 1 || eg == 7) {
      if (eg == 1)
         cat("Gu & Wahba 4 term additive model\n")
      else cat("Gu & Wahba 4 term additive model, correlated predictors\n")
      x0 <- runif(n, 0, 1)
      if (eg == 7)
         x1 <- x0 * 0.7 + runif(n, 0, 0.3)
      else x1 <- runif(n, 0, 1)
      x2 <- runif(n, 0, 1)
      if (eg == 7)
         x3 <- x2 * 0.9 + runif(n, 0, 0.1)
      else x3 <- runif(n, 0, 1)
      f0 <- function(x) 2 * sin(pi * x)
      f1 <- function(x) exp(2 * x)
      f2 <- function(x) 0.2 * x^11 * (10 * (1 - x))^6 + 10 *
         (10 * x)^3 * (1 - x)^10
      f3 <- function(x) 0 * x
```

```
    f <- f0(x0) + f1(x1) + f2(x2)
    if (dist == "normal") {
        e <- rnorm(n, 0, scale)
        y <- f + e
    }
    else if (dist == "poisson") {
        g <- exp(f * scale)
        f <- log(g)
        y <- rpois(rep(1, n), g)
    }
    else if (dist == "binary") {
        f <- (f - 5) * scale
        g <- binomial()$linkinv(f)
        y <- rbinom(g, 1, g)
    }
    else stop("dist not recognised")
    data <- data.frame(y = y, x0 = x0, x1 = x1, x2 = x2,
        x3 = x3, f = f, f0 = f0(x0), f1 = f1(x1), f2 = f2(x2),
        f3 = x3 * 0)
    return(data)
}
else if (eg == 2) {
    cat("Bivariate smoothing example\n")
    test1 <- function(x, z, sx = 0.3, sz = 0.4) {
        (pi^sx * sz) * (1.2 * exp(-(x - 0.2)^2/sx^2 - (z -
            0.3)^2/sz^2) + 0.8 * exp(-(x - 0.7)^2/sx^2 -
            (z - 0.8)^2/sz^2))
    }
    x <- runif(n)
    z <- runif(n)
    xs <- seq(0, 1, length = 40)
    zs <- seq(0, 1, length = 40)
    pr <- data.frame(x = rep(xs, 40), z = rep(zs, rep(40,
        40)))
    truth <- matrix(test1(pr$x, pr$z), 40, 40)
    f <- test1(x, z)
    y <- f + rnorm(n) * scale
    data <- data.frame(y = y, x = x, z = z, f = f)
    truth <- list(x = xs, z = zs, f = truth)
    return(list(data = data, truth = truth, pr = pr))
}
else if (eg == 3) {
    cat("Continuous `by' variable example\n")
    x1 <- runif(n, 0, 1)
    x2 <- sort(runif(n, 0, 1))
    f <- 0.2 * x2^11 * (10 * (1 - x2))^6 + 10 * (10 * x2)^3 *
        (1 - x2)^10
    e <- rnorm(n, 0, scale)
    y <- f * x1 + e
    return(data.frame(y = y, x1 = x1, x2 = x2, f = f))
}
else if (eg == 4) {
    cat("Factor `by' variable example\n")
    n <- 400
    x0 <- runif(n, 0, 1)
```

```
    x1 <- runif(n, 0, 1)
    x2 <- runif(n, 0, 1)
    f1 <- 2 * sin(pi * x2)
    f2 <- exp(2 * x2) - 3.75887
    f3 <- 0.2 * x2^11 * (10 * (1 - x2))^6 + 10 * (10 * x2)^3 *
       (1 - x2)^10
    e <- rnorm(n, 0, scale)
    fac <- as.factor(c(rep(1, 100), rep(2, 100), rep(3, 200)))
    fac.1 <- as.numeric(fac == 1)
    fac.2 <- as.numeric(fac == 2)
    fac.3 <- as.numeric(fac == 3)
    y <- f1 * fac.1 + f2 * fac.2 + f3 * fac.3 + e
    return(data.frame(y = y, x0 = x0, x1 = x1, x2 = x2, fac = fac,
      f1 = f1, f2 = f2, f3 = f3))
}
else if (eg == 5) {
    cat("Additive model + factor\n")
    x0 <- rep(1:4, 50)
    x1 <- runif(n, 0, 1)
    x2 <- runif(n, 0, 1)
    x3 <- runif(n, 0, 1)
    y <- 2 * x0
    y <- y + exp(2 * x1)
    y <- y + 0.2 * x2^11 * (10 * (1 - x2))^6 + 10 * (10 *
       x2)^3 * (1 - x2)^10
    e <- rnorm(n, 0, scale)
    y <- y + e
    x0 <- as.factor(x0)
    return(data.frame(y = y, x0 = x0, x1 = x1, x2 = x2, x3 = x3))
}
else if (eg == 6) {
    cat("4 term additive + random effect")
    dat <- gamSim(1, n = n, scale = 0)
    fac <- rep(1:4, n/4)
    dat$f <- dat$f + fac * 3
    dat$fac <- as.factor(fac)
    if (dist == "normal") {
       dat$y <- dat$f + rnorm(n) * scale
    }
    else if (dist == "poisson") {
       g <- exp(dat$f * scale)
       dat$y <- rpois(rep(1, n), g)
    }
    else if (dist == "binary") {
       g <- (dat$f - 5) * scale
       g <- binomial()$linkinv(g)
       dat$y <- rbinom(g, 1, g)
    }
    return(dat)
}
```