EUROPEAN JOURNAL OF PURE AND APPLIED MATHEMATICS
Vol. 1, No. 1, 2008, (4-37)
ISSN 1307-5543 – www.ejpam.com

Honorary Invited Paper

Multivariate Regression Models with Power Exponential Random Errors and Subset Selection Using Genetic Algorithms With Information Complexity

Minhui Liu[1], Hamparsum Bozdogan[2,*]

[1] *Department of Statistics, Operations and Management Science, University of Tennessee,*
[2] *Stokley Management Center, Knoxville Tennessee, 37996-0562 U.S.A.*

Abstract. In this paper we introduce and develop two different novel multivariate regression models with Power Exponential (PE) random errors for the first time. Our first model assumes that the observations are independent and the second model assumes that the observations are dependent. These two models coincide only when the shape parameter of the multivariate Power Exponential (MPE) distribution is equal to one which corresponds to the multivariate normal distribution. We develop method of moments (MOM) and the maximum likelihood (ML) methods to estimate the model parameters. The model selection criteria such as AIC and ICOMP(IFIM) for both models are derived. Two simulation examples and a real example on a benchmark data set are given to show the applications of these two models in subset selection of the best predictors. A genetic algorithm (GA) approach is used to obtain the estimates of the model parameters and to carry out the subset selection of the best predictors under these two different model types.

Key words: Multivariate Power Exponential Distribution, Multivariate Regression, AIC, ICOMP, Model Selection, Genetic Algorithm.

1. Introduction and Objectives

During the past fifty years, multivariate normal distribution has enjoyed a significant role in the development of many important multivariate modeling techniques including the multivariate regression models. In many practical applications such as in behavioral and social sciences, biometrics, chemometrics, econometrics, environmental sciences, and financial modeling to name a few, we cannot any longer assume the multinormality on the set of dependent variables or the random error term of the model. Real data often show significant departures from normality and normality may not be tenable, especially when the tails are thicker or thinner than those of normal distributions. For this reason, to achieve more flexibility in statistical modeling and model selection, and to robustify many multivariate statistical procedures, the purpose of this paper is to develop novel techniques in multivariate regression models for nonnormal data under the general class of: Multivariate Power Exponential (PE) distributions by broadening the usual multivariate normal assumption on the random errors under various assumptions. In regression models, the

*Corresponding author. *Email address:* bozdogan@utk.edu (H. Bozdogan)

random error terms are generally assumed to be normally distributed. However, since data often are non-normal, the normality assumption is not always tenable especially when the tails are thicker or thinner than those of normal distributions. The *Power Exponential* (*PE*) distribution family which is introduced by Subbotin ([32]) and popularized by Box and Tiao ([6]), has been used in modeling economic and financial data as a generalization of normal distribution in recent years (e.g. [28, 33, 34]). Gómez-Villegas and Sá nchez-Manzano et al. ([20, 31]) proposed multivariate and matrix generalizations of the PE family of distributions and studied their properties in relation to multivariate *Elliptically Contoured* (*EC*) distributions.

Zeckhauser and Thompson ([36]) were probably the first who attempted to study the simple multiple linear regression model with PE error terms in a short but an incomplete paper. Liu and Bozdogan ([23]) developed a GA on GA (or GA engineering) approach for PE multiple regression and subset selection of variables with information-theoretic complexity (ICOMP) criterion. The closed form expressions of the inverse Fisher Information Matrix (IFIM) and ICOMP(IFIM) for PE multiple linear regression models were also given. Finding proper criterion or measure for the comparison of competing models is important to select correct regression models. AIC-type criteria ([2–5]) are the most widely used information-based criteria for model selection in recent years. However, the penalty term used in AIC-type criteria is insufficient to measure the model complexity which has been cited by many authors (see, e.g. [29]). Bozdogan's ICOMP criteria ([7,9,10,12–14]) improve AIC-type criteria by using an information-theoretic measure of *"overall"* model complexity based on the generalized covariance complexity index of van Emden ([16]). ICOMP(IFIM) is the most general form of ICOMP.

In this paper, we extend the work of Liu and Bozdogan ([23]) and study the multivariate regression models with PE random errors under various assumptions. As a short hand notation, we abbreviate these models as MVPER models. We develop and use the genetic algorithms (GAs) for both the parameter estimation of the models and for subset selection of best predictors with information criteria as our fitness function. We note that, in the literature, this work seems to be the first to attempt to study MVPER models utilizing modern optimization techniques such the GAs and information criteria as our fitness function. More specifically, we consider the multivariate regression model:

$$Y_{n\times p} = X_{n\times q}B_{q\times p} + E_{n\times p}, p+q \leq n, rank(X) = q \tag{1.1}$$

where $Y = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_p] = [\mathbf{y}_{(1)}, \mathbf{y}_{(2)}, \cdots, \mathbf{y}_{(n)}]'$ is the matrix of n observations on p response variables. $X = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_q] = [\mathbf{x}_{(1)}, \mathbf{x}_{(2)}, \cdots, \mathbf{x}_{(n)}]'$ is the matrix of $(n \times q)$ constant terms on non-stochastic predictor variables, $B = [\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_p] = [\mathbf{b}_{(1)}, \mathbf{b}_{(2)}, \cdots, \mathbf{b}_{(q)}]'$ is the coefficient matrix and $E = [\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_p] = [\varepsilon_{(1)}, \varepsilon_{(2)}, \cdots, \varepsilon_{(n)}]'$ is the unobservable random error term matrix. The error terms are assumed to be multivariate PE rather than normal. Applying the *vector operator* $Vec(\cdot)$ ([30, p.21]) to (1.1), the multivariate regression model can be transformed to a univariate regression problem given by

$$Vec(Y') = Vec(B'X') + Vec(E') = (X \otimes I_p)Vec(B') + Vec(E') \tag{1.2}$$

or

$$\mathbf{y}_{np\times 1} = (X \otimes I_p)\mathbf{b}_{pq\times 1} + \varepsilon_{np\times 1} \tag{1.3}$$

where $\mathbf{y}_{np\times 1} = [\mathbf{y}'_{(1)}, \mathbf{y}'_{(2)}, \cdots, \mathbf{y}'_{(n)}]'$, $\mathbf{b}_{pq\times 1} = [\mathbf{b}'_{(1)}, \mathbf{b}'_{(2)}, \cdots, \mathbf{b}'_{(q)}]'$ and $\varepsilon_{np\times 1} = [\varepsilon'_{(1)}, \varepsilon'_{(2)}, \cdots, \varepsilon'_{(n)}]'$. I_p is the $(p \times p)$ dimensional identity matrix. $\otimes$ is *Kronecker product* operator ([30, p.12]). In this paper, we let $Vec'(\cdot)$ denote $(Vec(\cdot))'$.

The rest of the paper is organized as follows. In Section 2, we introduce the multivariate PE distribution. In Section 3, we study Type I MVPER model and in Section 4 we study Type II MVPER model which takes the dependency structure of the data into account. Section 5 gives the derivation of AIC and ICOMP(IFIM) for these two types of MVPER models. Section 6 outlines and presents the general background of the Genetic Algorithms (GAs). In Section 7, we give two simulation examples and a real model selection example on a benchmark macro-economic data in subset selection of best predictors under both the Type I and Type II MVPER models to illustrate the versatility of our new approach. Section 8 concludes the paper.

2. Multivariate PE Distributions

A random variable z is PE distributed if the density function of z is

$$f(z;\mu,\sigma,\beta) = \frac{1}{\sigma\,\Gamma\left(1+\frac{1}{2\beta}\right)2^{1+\frac{1}{2\beta}}}\exp\left(-\frac{1}{2}\left|\frac{z-\mu}{\sigma}\right|^{2\beta}\right), \tag{2.1}$$

where the parameters $-\infty < \mu < \infty$ and $\sigma > 0$ are location and scale parameters, respectively, and $\beta > 0$ is the shape parameter, which is related to the *kurtosis parameter*. A family of unimodel symmetric curves with different shapes for different values of β can be represented by the above density. When $\beta = 0.5$, the Laplace distribution, and when $\beta \to \infty$ the Uniform distribution arises. Particularly when $\beta = 1$, the density becomes normal distribution. So PE distribution can be seen as a generalized normal distribution. Standard PE distribution is the PE distribution with mean zero and $\sigma = 1$.

According to Gómez et al. ([20]), a multivariate generalization of the PE family of distributions, denoted by $PE_p(\boldsymbol{\mu}, \Sigma, \beta)$, is defined as

$$f(\mathbf{z};\boldsymbol{\mu},\Sigma,\beta) = \frac{p\Gamma(p/2)}{\pi^{p/2}\Gamma(1+\frac{p}{2\beta})2^{1+p/2\beta}}|\Sigma|^{-1/2}\exp(-\frac{1}{2}((\mathbf{z}-\boldsymbol{\mu})'^{-1}(\mathbf{z}-\boldsymbol{\mu}))^{\beta}) \tag{2.2}$$

where $\mathbf{z} = [z_1, z_2, \cdots, z_p]'$ is a p dimension random vector, $\boldsymbol{\mu} \in \mathcal{R}^p$, Σ is a $(p \times p)$ positive definite symmetric matrix, and $\beta \in (0, \infty)$ is the shape parameter. If β is given, this multivariate generalized PE distribution is actually a multivariate *Elliptically Contoured* (EC) distribution $EC_p(\boldsymbol{\mu}, \Sigma, g)$ with the probability density generator $g(t) = \exp(-\frac{1}{2}t^{\beta})$ ([18, p.46]). Further, it is a special case of symmetric *Kotz* type distribution ([18, p.76]) with $N = 1$. When $p = 1$, (2.2) reduces to (2.1).

Sánchez-Manzano et al. ([31]) gave the definition of matrix variate PE distribution. A random $(n \times p)$ matrix Z has a $(n \times p)$-variate PE distribution, denoted as $Z \sim MPE_{p\times n}(M, \Phi, \Sigma, \beta)$ with parameters M, a $(n \times p)$ matrix; Φ, a $(n \times n)$ positive definite matrix; Σ, a $(p \times p)$ positive definite matrix and $\beta \in (0, \infty)$ if

$$Vec(Z') \sim PE_{np}(Vec(M'), \Phi \otimes \Sigma, \beta). \tag{2.3}$$

The matrix form of the density function of Z is then given by

$$f(Z;M,\Phi,\Sigma,\beta)=k|\Phi|^{-p/2}|\Sigma|^{-n/2}\exp(-\frac{1}{2}(tr((Z-M)'^{-1}(Z-M)\Sigma^{-1}))^{\beta}) \tag{2.4}$$

where

$$k=\frac{np\Gamma(np/2)}{\pi^{np/2}\Gamma(1+np/2\beta)2^{1+np/2\beta}}.$$

If $Z\sim MPE_{p\times n}(M,\Phi,\Sigma,\beta)$, then $Z'\sim MPE_{n\times p}(M',\Sigma,\Phi,\beta)$. Some probabilistic characteristics of Z are given by:

$$\begin{aligned}
&E[Z]=M,\\
&Var[Vec(Z')]=\frac{2^{1/\beta}\Gamma(\frac{np+2}{2\beta})}{np\Gamma(\frac{np}{2\beta})}(\Phi\otimes\Sigma),\\
&\gamma_1[Z]=0,\\
&\gamma_2[Z]=\frac{(np)^2\Gamma(\frac{np}{2\beta})\Gamma(\frac{np+4}{2\beta})}{\Gamma^2(\frac{np+2}{2\beta})}-np(np+2),\\
&E[(tr((Z-M)'^{-1}(Z-M)\Sigma^{-1}))^s]=\frac{2^{s/\beta}\Gamma(\frac{np+2s}{2\beta})}{\Gamma(\frac{np}{2\beta})},
\end{aligned}$$

where s is a positive integer, γ_1 and γ_2 are *multidimensional asymmetry (skewness)* and *kurtosis* coefficients ([26]) defined as

$$\begin{aligned}
\gamma_1[Z]= \;& E[((Vec(Z')-Vec(M'))'(Var[Vec(Z'^{-1}(Vec(Z')-Vec(M'^3],\\
\gamma_2[Z]= \;& E[((Vec(Z')-Vec(M'))'(Var[Vec(Z'^{-1}(Vec(Z')-Vec(M'^2]\\
& -np(np+2).
\end{aligned}$$

From the above, it is easy to show that

$$E[(Vec(Z')-Vec(M'))'(Var[Vec(Z'^{-1}(Vec(Z')-Vec(M'))]=np.$$

For a given $\beta>0$, Z actually is a matrix variate EC distribution denoted as

$$Z\sim E_{n,p}(M,\Phi\otimes\Sigma,\Psi)$$

by Gupta and Varga ([21, p.20, p.26]) with $h(t)=k\exp(-t^{\beta}/2)$. For $n=1$ and $\Phi=I_n$, Z' and Z have the same distribution. The parameters in the definition of a matrix multivariate PE distribution are not uniquely defined as shown in the following theorem.

Theorem 2.1. *Let $Z\sim MPE_{p\times n}(M_1,\Phi_1,\Sigma_1,\beta_1)$ and at the same time $Z\sim MPE_{p\times n}(M_2,\Phi_2,\Sigma_2,\beta_2)$. If Z is non-degenerate, then there exist positive constant c such that $M_2=M_1$, $\Sigma_2=c\Sigma_1$, $\Phi_2=\Phi_1/c$ and $\beta_2=\beta_1$.*

Proof: The proof follows along the lines given in Gupta and Varga ([21, p.23]). Since Z is symmetric about M_1 as well as about M_2, then $M_2=M_1$. Let $M=M_1$ and

$$\begin{aligned}
&\Sigma_l=[{}_l\sigma_{ij}],\ i,j=1,\cdots,p;\ l=1,2,\\
&\Phi_l=[{}_l\phi_{ij}],\ i,j=1,\cdots,n;\ l=1,2.
\end{aligned}$$

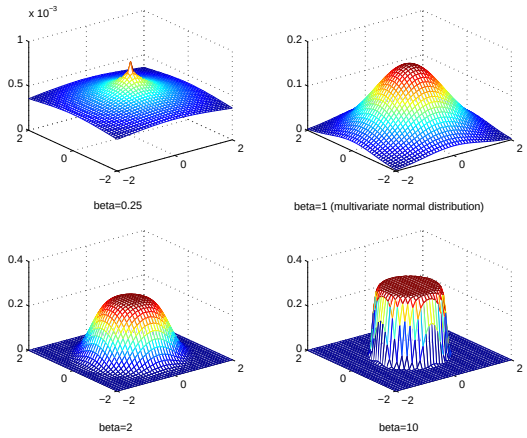

Figure 1: Matrix PE density function with $n = 1$, $p = 2$, $\Phi = I_1$, $\Sigma = I_2$ and variate β.

Let $\mathbf{k}(i)$ denote the p-dimensional vector whose i^{th} entry is 1 and all the others are 0. Let $\mathbf{l}(i)$ denote the n-dimensional vector whose i^{th} entry is 1 and all the others are 0. Since Z is non-degenerate, it must have an element $z_{i_0j_0}$ which is non-degenerate. Since $z_{i_0j_0} = \mathbf{l}'(i_0)Z\mathbf{k}(j_0)$, from Theorem 4 of Sánchez-Manzano et al. ([31]), there are $z_{i_0j_0} \sim PE(m_{i_0j_0}, {}_1\phi_{i_0i_0 1}\sigma_{j_0j_0}, \beta_1)$ and $z_{i_0j_0} \sim PE(m_{i_0j_0}, {}_2\phi_{i_0i_0 2}\sigma_{j_0j_0}, \beta_2)$. So we have $\beta_2 = \beta_1$ and ${}_2\phi_{i_0i_0 2}\sigma_{j_0j_0} = {}_1\phi_{i_0i_0 1}\sigma_{j_0j_0}$. From Gupta and Varga ([21, p.24]), there must be $\Phi_2 \otimes \Sigma_2 = \Phi_1 \otimes \Sigma_1$. By the Theorem 1.3.16 of Gupta and Varga ([21, p.13]), there exists a nonzero real number c such that $\Sigma_2 = c\Sigma_1$ and $\Phi_2 = \Phi_1/c$. If $\Phi_2 = \Phi_1 = \Phi$, then $c = 1$ and $\Sigma_2 = \Sigma_1$. □

From the proof of Theorem 2.1, the only case that a matrix multivariate PE distribution is not uniquely defined is that there exist positive constant c such that $\Sigma_2 = c\Sigma_1$ and $\Phi_2 = \Phi_1/c$. So if the parameter Σ or Φ is known, then the distribution is uniquely defined by the other three parameters. Therefore, in the rest of this paper, we only consider the case that parameter Φ is known to handle the uniqueness issue.

An advantage of *PE* distribution is that it is adaptive to both *peakedness* and *flatness* in the data by varying the values of β. When β increases, the sharpness diminishes. Figure 1 represents the plot of (2.2) with $n = 1$, $p = 2$, $\Phi = I_1$, $\Sigma = I_2$ and the shape parameter β. The relationship between β and γ_2 for the multivariate PE distribution is shown in Figure 2. For $\beta = 1$, (2.2) is a multivariate normal distribution and for $\beta \to \infty$, (2.2) is a multivariate uniform distribution. We note that as the dimension of the distribution gets larger, the kurtosis gets larger for both $\beta < 1$ and $\beta > 1$. What this means is that for large dimensional data sets, we expect heavy fat tails. Therefore, we should use flexible distributional models to capture the heavy fat tail behavior of large dimensional data sets.

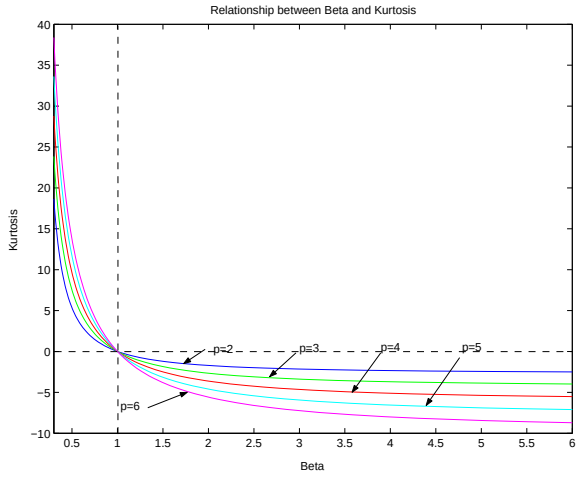

Figure 2: The relationship between β and kurtosis.

3. Type I Multivariate PE Regression Model

In model (1.1), if we assume that the rows of the random error matrix E are drawn independently from $PE_p(\mathbf{0}, \Sigma, \beta)$ distribution, i.e., $\varepsilon_{(1)}, \varepsilon_{(2)}, \cdots, \varepsilon_{(n)}$ are *i.i.d.* and $\varepsilon_{(1)} \sim PE_p(\mathbf{0}, \Sigma, \beta)$, then $\mathbf{y}_{(i)} \sim PE_p(B'\mathbf{x}_{(i)}, \Sigma, \beta)$. We denote the multivariate linear regression model under this assumption as Type I MVPER model. In this case, the likelihood function is

$$\mathcal{L}(B, \Sigma, \beta | Y, X) = k_1 \exp(-\frac{1}{2}\sum_{i=1}^{n}((\mathbf{y}_{(i)} - B'\mathbf{x}_{(i)})'\Sigma^{-1}(\mathbf{y}_{(i)} - B'\mathbf{x}_{(i)}))^{\beta}), \tag{3.1}$$

where

$$k_1 = \frac{p^n \Gamma^n(\frac{p}{2})}{\pi^{np/2}\Gamma^n(1+\frac{p}{2\beta})2^{n+\frac{np}{2\beta}}} |\Sigma|^{-n/2}.$$

The log likelihood function is

$$\begin{array}{l} l(B, \Sigma, \beta | Y, X) \equiv \log \mathcal{L}(B, \Sigma, \beta | Y, X) \\ = n\log(p\Gamma(\frac{p}{2})) - \frac{np}{2}\log(\pi) - n\log\Gamma(1+\frac{p}{2\beta}) - n(1+p/2\beta)\log 2 \\ -\frac{n}{2}\log|\Sigma| - \frac{1}{2}\sum_{i=1}^{n}((\mathbf{y}_{(i)} - B'\mathbf{x}_{(i)})'\Sigma^{-1}(\mathbf{y}_{(i)} - B'\mathbf{x}_{(i)}))^{\beta}. \end{array} \tag{3.2}$$

Let $\boldsymbol{\theta} = (\mathbf{b}', Vec'(\Sigma), \beta)'$, $\varepsilon_{(i)} = \mathbf{y}_{(i)} - B'\mathbf{x}_{(i)}$ and $t_i = \varepsilon_{(i)}'^{-1}\varepsilon_{(i)}$. Differentiating (3.2) with respect to $\mathbf{b}$, $Vec(\Sigma)$ and β respectively, we have

$$\frac{\partial l(\theta)}{\partial \mathbf{b}} = \beta\sum_{i=1}^{n} t_i^{\beta-1} Vec(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}'), \tag{3.3}$$

$$\frac{\partial l(\theta)}{\partial Vec(\Sigma)} = -\frac{n}{2}Vec(\Sigma^{-1}) + \frac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1} Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon_{(i)}'^{-1}), \tag{3.4}$$

and

$$\frac{\partial l(\theta)}{\partial \beta} = \frac{np}{2\beta^2}\psi(1+\frac{p}{2\beta}) + \frac{np}{2\beta^2}\log 2 - \frac{1}{2}\sum_{i=1}^{n} t_i^{\beta}\log t_i, \tag{3.5}$$

where $\psi(\cdot) = \mathrm{d}\log\Gamma(x)/\mathrm{d}x$ is called digamma function ([1]), or psi function. To derive and construct the Fisher information matrix (FIM) and its inverse IFIM, we have

$$\frac{\partial^2(\theta)}{\partial \mathbf{b}'\partial \mathbf{b}} = \beta\sum_{i=1}^{n}(I_{pq}\otimes Vec'(I_p))(I_q\otimes M_i\otimes I_p)(Vec(I_q)\otimes I_{pq}), \tag{3.6}$$

where

$$M_i = -t_i^{\beta-1}Vec(\Sigma^{-1})Vec'(\mathbf{x}_{(i)}\mathbf{x}'_{(i)}) - 2(\beta-1)t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)})\otimes(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)}).$$

$$\frac{\partial^2 l(\theta)}{\partial Vec'(\Sigma)\partial Vec(\Sigma)} = (I_{p^2}\otimes Vec'(I_p))(I_p\otimes\frac{\partial^2 l(\theta)}{\partial\Sigma\partial\Sigma}\otimes I_p)(Vec(I_p)\otimes I_{p^2}), \tag{3.7}$$

where

$$\begin{aligned}\tfrac{\partial^2 l(\theta)}{\partial\Sigma\partial\Sigma} = \;& \tfrac{n}{2}Vec(\Sigma^{-1})Vec'^{-1}) \\ & -\tfrac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1}(Vec(\Sigma^{-1})Vec'^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}) \\ & +Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})Vec'^{-1})) \\ & -\tfrac{\beta(\beta-1)}{2}\sum_{i=1}^{n} t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})\otimes(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}).\end{aligned}$$

$$\tfrac{\partial^2 l(\theta)}{\partial\beta^2} = \; -\tfrac{np}{\beta^3}\psi(1+\tfrac{p}{2\beta}) - \tfrac{np^2}{4\beta^4}\psi'(1+\tfrac{p}{2\beta}) - \tfrac{np}{\beta^3}\log 2 - \tfrac{1}{2}\sum_{i=1}^{n} t_i^{\beta}\log^2 t_i, \tag{3.8}$$

where $\psi'(\cdot) = \mathrm{d}^2\log\Gamma(x)/\mathrm{d}x^2$ is called trigamma function.

$$\frac{\partial^2 l(\theta)}{\partial Vec'(\Sigma)\partial \mathbf{b}} = \beta\sum_{i=1}^{n}(I_{pq}\otimes Vec'(I_p))(I_q\otimes N_i\otimes I_p)(Vec(I_q)\otimes I_{p^2}), \tag{3.9}$$

where

$$\begin{aligned}N_i = \;& -t_i^{\beta-1}Vec(\Sigma^{-1})Vec'^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)}) \\ & -(\beta-1)t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'^{-1}_{(i)}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}).\end{aligned}$$

$$\tfrac{\partial^2 l(\theta)}{\partial\beta\partial\mathbf{b}} = \; \sum_{i=1}^{n} t_i^{\beta-1}Vec(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)}) + \beta\sum_{i=1}^{n} t_i^{\beta-1}\log(t_i)Vec(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)}). \tag{3.10}$$

$$\begin{aligned}\tfrac{\partial^2 l(\theta)}{\partial\beta\partial Vec(\Sigma)} = \;& \tfrac{1}{2}\sum_{i=1}^{n} t_i^{\beta-1}Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}) \\ & +\tfrac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1}\log(t_i)Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}).\end{aligned} \tag{3.11}$$

Some of the details of the above derivations are given in the Appendix. Since there are no closed form solutions to the likelihood equations, numerical methods such as *Genetic Algorithms* or *Newton-Raphson iterative method* can be used to obtain the maximum likelihood estimators (MLEs). Here we give a method to compute the method of moments (MOM) estimates which can be used as the starting values to calculate the MLEs. The steps of MOM are:

Step 1 : Compute $\hat{B} = (X'X)^{-1}X'Y$.

Step 2 : Let $d_i = \varepsilon'_{(i)}Var(\mathbf{y}_{(i)})^{-1}\varepsilon_{(i)}$. d_i actually is the squared *Mahalanobis distances*. By the probability characteristics of multivariate PE distribution, we have

$$E[t_i^2] = \frac{2^{2/\beta}\Gamma(\frac{p+4}{2\beta})}{\Gamma(\frac{p}{2\beta})}.$$

Then

$$E[d_i^2] = E[\frac{p^2\Gamma^2(\frac{p}{2\beta})}{2^{2/\beta}\Gamma^2(\frac{p+2}{2\beta})}t_i^2] = \frac{p^2\Gamma(\frac{p}{2\beta})\Gamma(\frac{p+4}{2\beta})}{\Gamma^2(\frac{p+2}{2\beta})}.$$

So we can compute $\hat{\beta}$ as the solution of

$$\frac{p^2\Gamma(\frac{p}{2\hat{\beta}})\Gamma(\frac{p+4}{2\hat{\beta}})}{\Gamma^2(\frac{p+2}{2\hat{\beta}})} = \frac{1}{n}\sum_{i=1}^{n}\hat{d}_i^2$$

where $\hat{d}_i = (\mathbf{y}_{(i)} - \hat{B}'\mathbf{x}_{(i)})'^{-1}(\mathbf{y}_{(i)} - \hat{B}'\mathbf{x}_{(i)})$ and $S = (Y - X\hat{B})'(Y - X\hat{B})/n$ is the sample covariance matrix.

Step 3 : Compute

$$\hat{\Sigma} = \frac{p\Gamma(\frac{p}{2\hat{\beta}})}{2^{1/\hat{\beta}}\Gamma(\frac{p+2}{2\hat{\beta}})}S.$$

4. Type II Multivariate PE Regression Model

If we assume that the random error terms $Vec(E')$ in model (1.2) has a multivariate PE distribution $PE_{np}(\mathbf{0}, \Phi \otimes \Sigma, \beta)$, i.e., $Vec(Y') \sim PE_{np}((X \otimes I_p)\mathbf{b}_{pq\times 1}, \Phi \otimes \Sigma, \beta)$ or with the matrix notation $Y \sim MPE_{n\times p}(XB, \Phi, \Sigma, \beta)$, where Φ is a $(n \times n)$ positive definite symmetric matrix and known, then the density function of E is:

$$f(E; \Phi, \Sigma, \beta) = k|\Phi|^{-p/2}|\Sigma|^{-n/2}\exp(-\frac{1}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta}) \tag{4.1}$$

or, the density function of Y is:

$$f(Y; B, \Phi, \Sigma, \beta) = k|\Phi|^{-p/2}|\Sigma|^{-n/2}\exp(-\frac{1}{2}(tr(\Sigma^{-1}(Y - XB)'^{-1}(Y - XB)))^{\beta}). \tag{4.2}$$

We denote the multivariate regression model under this assumption as Type II MVPER model. It is noted that Type I and Type II MVPER models coincide only if $\beta = 1$ and $\Phi = I_n$, which corresponds to the multivariate normal distribution. When $\Phi = I_n$, each row of E, or Y, are still considered as an observation of a p dimensional random vector and the n rows have the same distribution. But the n observations are assumed to be *uncorrelated* rather than *independent* from each other. The identity matrix I_n reflects the lack of correlation among the observations.

Under the assumption of Type II MVPER model, the likelihood function is:

$$\mathcal{L}(B,\Sigma,\beta)=k|\Phi|^{-p/2}|\Sigma|^{-n/2}\exp(-\frac{1}{2}(tr(\Sigma^{-1}(Y-XB)'^{-1}(Y-XB)))^{\beta}), \tag{4.3}$$

and the log likelihood function is:

$$\begin{array}{rl} & l(B,\Sigma,\beta)\equiv\log\mathcal{L}(B,\Sigma,\beta)\\ = & \log(np\Gamma(\frac{np}{2}))-\frac{np}{2}\log(\pi)-\log\Gamma(1+\frac{np}{2\beta})-(1+np/2\beta)\log 2\\ & -\frac{p}{2}\log|\Phi|-\frac{n}{2}\log|\Sigma|-\frac{1}{2}(tr(\Sigma^{-1}(Y-XB)'^{-1}(Y-XB)))^{\beta}. \end{array} \tag{4.4}$$

Let $\boldsymbol{\theta}$=($\mathbf{b}', Vec'(\Sigma), \beta)'$. Differentiating (4.4) with respect to $\mathbf{b}$, $Vec(\Sigma)$ and β respectively, we have

$$\frac{\partial l(\theta)}{\partial\mathbf{b}}=Vec((\frac{\partial l(\theta)}{\partial B})'^{-1}E'^{-1}E))^{\beta-1}Vec(\Sigma^{-1}E'^{-1}X), \tag{4.5}$$

$$\begin{array}{rl} \frac{\partial l(\theta)}{\partial Vec(\Sigma)} = & Vec(\frac{\partial l(\theta)}{\partial\Sigma})\\ = & -\frac{n}{2}Vec(\Sigma^{-1})+\frac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}Vec(\Sigma^{-1}E'^{-1}E) \end{array} \tag{4.6}$$

and

$$\begin{array}{rl} \frac{\partial l(\theta)}{\partial\beta} = & \frac{np}{2\beta^2}\psi(1+\frac{np}{2\beta})+\frac{np}{2\beta^2}\log 2\\ & -\frac{1}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta}\log(tr(\Sigma^{-1}E'^{-1}E)). \end{array} \tag{4.7}$$

To derive and construct the FIM and IFIM of the model parameters, we have

$$\begin{array}{rl} \frac{\partial l^2(\theta)}{\partial\mathbf{b}'\partial\mathbf{b}} = & \frac{\partial}{\partial\mathbf{b}'}(\frac{\partial l(\theta)}{\partial\mathbf{b}})\\ = & (I_{pq}\otimes Vec'(I_p))(I_q\otimes\frac{\partial l^2(\theta)}{\partial B'\partial B'}\otimes I_p)(Vec(I_q)\otimes I_{pq}) \end{array} \tag{4.8}$$

where

$$\begin{array}{rl} \frac{\partial l^2(\theta)}{\partial B'\partial B'} = & \beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}\frac{\partial(\Sigma^{-1}E'^{-1}X)}{\partial B'}\\ & +(\Sigma^{-1}E'^{-1}X)\otimes\frac{\partial(\beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1})}{\partial B'}, \end{array} \tag{4.9}$$

$$\frac{\partial(\Sigma^{-1}E'^{-1}X)}{\partial B'} = -Vec(\Sigma^{-1})Vec'(X'^{-1}X) \tag{4.10}$$

and

$$\frac{\partial(\beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1})}{\partial B'}=-2\beta(\beta-1)(tr(\Sigma^{-1}E'^{-1}E))^{\beta-2}\Sigma^{-1}E'^{-1}X. \tag{4.11}$$

$$\frac{\partial l^2(\theta)}{\partial Vec'(\Sigma)\partial Vec(\Sigma)} = (I_{p^2}\otimes Vec'(I_p))(I_p\otimes\frac{\partial l^2(\theta)}{\partial\Sigma\partial\Sigma}\otimes I_p)(Vec(I_p)\otimes I_{p^2}) \tag{4.12}$$

where

$$\begin{array}{rl} \frac{\partial l^2(\theta)}{\partial\Sigma\partial\Sigma} = & \frac{n}{2}Vec(\Sigma^{-1})Vec'^{-1})\\ & -\frac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(Vec(\Sigma^{-1})Vec'^{-1}E'^{-1}E\Sigma^{-1})\\ & +Vec(\Sigma^{-1}E'^{-1}E\Sigma^{-1})Vec'^{-1}))\\ & -\frac{\beta(\beta-1)}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-2}\\ & ((\Sigma^{-1}E'^{-1}E\Sigma^{-1})\otimes(\Sigma^{-1}E'^{-1}E\Sigma^{-1})). \end{array} \tag{4.13}$$

$$\begin{array}{rcl}\frac{\partial l^2(\theta)}{\partial\beta^2} = & -\frac{np}{\beta^3}\psi(1+\frac{np}{2\beta}) - \frac{n^2p^2}{4\beta^4}\psi'(1+\frac{np}{2\beta}) - \frac{np}{\beta^3}\log 2 \\ - & \frac{1}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta}\log^2(tr(\Sigma^{-1}E'^{-1}E)).\end{array} \tag{4.14}$$

$$\frac{\partial l^2(\theta)}{\partial\beta\partial\mathbf{b}} = (tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}Vec(\Sigma^{-1}E'^{-1}X)(1+\beta\log(tr(\Sigma^{-1}E'^{-1}E))). \tag{4.15}$$

$$\begin{array}{rcl}\frac{\partial l^2(\theta)}{\partial\beta\partial Vec(\Sigma)} = & \frac{1}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}Vec(\Sigma^{-1}E'^{-1}E\Sigma^{-1}) \\ & (1+\beta\log(tr(\Sigma^{-1}E'^{-1}E))).\end{array} \tag{4.16}$$

$$\frac{\partial l^2(\theta)}{\partial Vec'(\Sigma)\partial\mathbf{b}} = (I_{pq}\otimes Vec'(I_p))(I_q\otimes\frac{\partial l^2(\theta)}{\partial\Sigma\partial B'}\otimes I_p)(Vec(I_q)\otimes I_{p^2}) \tag{4.17}$$

where

$$\begin{array}{rcl}\frac{\partial l^2(\theta)}{\partial\Sigma\partial B'} = & -\beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}Vec(\Sigma^{-1})Vec'^{-1}E'^{-1}X) \\ & -\beta(\beta-1)(tr(\Sigma^{-1}E'^{-1}E))^{\beta-2}(\Sigma^{-1}E'^{-1}X)\otimes(\Sigma^{-1}E'^{-1}E\Sigma^{-1}).\end{array} \tag{4.18}$$

Further details of the above derivations are given in the Appendix of the paper.

By the probability characteristics of multivariate PE distribution, we have

$$\begin{array}{rcl} & E[((Vec(Y')-Vec(B'X'))'Var(Vec(Y'^{-1}(Vec(Y')-Vec(B'X'^2] \\ = & \frac{(np)^2\Gamma(\frac{np}{2\beta})\Gamma(\frac{np+4}{2\beta})}{\Gamma^2(\frac{np+2}{2\beta})}.v\end{array}$$

Then, the method of moment estimate of β, denoted as $\hat{\beta}$, can be obtained by solving the equation:

$$((Vec(Y')-Vec(\hat{B}'X'))'^{-1}(Vec(Y')-Vec(\hat{B}'X'^2 = \frac{(np)^2\Gamma(\frac{np}{2\hat{\beta}})\Gamma(\frac{np+4}{2\hat{\beta}})}{\Gamma^2(\frac{np+2}{2\hat{\beta}})}$$

where

$$S = (Vec(Y')-Vec(\hat{B}'X'))(Vec(Y')-Vec(\hat{B}'X'))'$$

and

$$\hat{B} = (X'X)^{-1}X'Y$$

is the method of moment estimate of B. By Anderson and Fang ([17, p.215]), the unbiased estimator of Σ can be obtained as

$$\hat{\Sigma} = \frac{np\Gamma(\frac{np}{2\hat{\beta}})}{(n-q)2^{1/\hat{\beta}}\Gamma(\frac{np+2}{2\hat{\beta}})}(Y-X\hat{B})'^{-1}(Y-X\hat{B}).$$

Under the assumption of Type II MVPER model, indeed the sample size is only 1. According to Gupta and Varga ([21, p.224]), the MLEs of the model parameters do not exist without imposing some restrictions on Σ and Φ even if Φ is known. For $n \geq p$, since $h(t) = k\exp(-t^{\beta}/2)$ is monotone decreasing on $(0,+\infty)$, there is

$$\hat{B}_{MLE} = \hat{B}$$

by Theorem 7.1.1 of ([21, p.226]) and the MLE of Σ is

$$\hat{\Sigma}_{MLE} = \frac{p}{\lambda_{max}}(Y - X\hat{B})'^{-1}(Y - X\hat{B})$$

where λ_{max} is the maximum point of the function

$$f(\lambda) = \lambda^{np/2}h(\lambda)$$

by Theorem 7.1.3 of ([21, p.235]). It is easy now to show that

$$\lambda_{max} = (\frac{np}{\hat{\beta}_{MLE}})^{1/\hat{\beta}_{MLE}}$$

by the Lemma 2 of ([17, p.204]). However, if we substitute $\hat{\Sigma}_{MLE}$ and $\hat{B}_{MLE}$ into the log likelihood function (4.4), the log likelihood function of β becomes:

$$\begin{aligned} f(\beta) = \quad & \log(np\Gamma(\tfrac{np}{2})) - \tfrac{np}{2}\log(\pi) - \log\Gamma(1 + \tfrac{np}{2\beta}) - \log 2 - \tfrac{np}{2}\log p - \tfrac{p}{2}\log|\Phi| \\ & -\tfrac{n}{2}\log|(Y - X\hat{B})'^{-1}(Y - X\hat{B})| + \tfrac{np}{2\beta}\log(\tfrac{np}{2\beta}) - \tfrac{np}{2\beta} \end{aligned}$$

which actually has no maximum.

We get around this problem by providing two methods to compute the MLEs of Type II MVPER model parameters. One method is to maximize the log likelihood function directly with an algorithm such as the GA given in Bozdogan and Liu ([23]). The other method is to employ a two step procedure given as follows:

Step 1 : Use a set of m pairs of observations randomly selected from the original data to compute $\hat{B}_{MLE}$ and $\hat{\Sigma}_{MLE}$ by considering the shape parameter β fixed.

Step 2 : Substitute $\hat{B}_{MLE}$ and $\hat{\Sigma}_{MLE}$ obtained in *Step 1* and the original sample data into the log likelihood function (4.4), then $\hat{\beta}_{MLE}$ is the value of β which maximizes the log likelihood function of β.

5. Information Criteria for Multivariate PE Regression Models

In recent years, information-based criteria such as Akaike's AIC ([2–5]), which compromises between the goodness-of-fit and the model complexity, have been widely used in statistical modeling and model selection. However, the penalty term used in AIC-type criteria, that is, the number of free parameters, is insufficient to measure the model complexity as noted by many authors (see, e.g. [29]). ICOMP criteria ([7, 9, 10, 12–14]) improve AIC-type criteria by using an information-theoretic measure of "overall" model complexity based on the generalized covariance complexity index of Van Emden ([16]). ICOMP criteria can be defined in several ways. The most general form of ICOMP, referred to as ICOMP(IFIM), exploits the well-known asymptotic optimality properties of the MLE's, and uses the IFIM to measure the complexity of a model. For both types of criteria, the model with the smallest score is chosen to be the best model.

For a general multivariate linear or nonlinear model, AIC is defined as:

$$\text{AIC} = -2\log\mathcal{L}(\hat{\theta}) + 2k \tag{5.1}$$

where k is the number of free parameters estimated within the model. So for Type I MVPER model, we have

$$\begin{aligned}\text{AIC}_{\text{Model I}} = \; & -2n\log(p\Gamma(\tfrac{p}{2})\Gamma(1+\tfrac{p}{2\hat{\beta}})) + np\log(\pi) + 2n(1+p/2\hat{\beta})\log 2 \\ & +n\log|\hat{\Sigma}| + \textstyle\sum_{i=1}^{n}((\mathbf{y}_{(i)}-\hat{B}'\mathbf{x}_{(i)})'\hat{\Sigma}^{-1}(\mathbf{y}_{(i)}-\hat{B}'\mathbf{x}_{(i)}))^{\hat{\beta}} \\ & +2pq + p(p+1) + 2\end{aligned} \tag{5.2}$$

and for Type II MVPER model, we have

$$\begin{aligned}& \text{AIC}_{\text{Model II}} \\ = \; & -2\log(np\Gamma(\tfrac{np}{2})) + np\log(\pi) + 2\log\Gamma(1+\tfrac{np}{2\hat{\beta}}) + (2+np/\hat{\beta})\log 2 \\ & +p\log|\Phi| + n\log|\hat{\Sigma}| + (tr(\hat{\Sigma}^{-1}(Y-X\hat{B})'^{-1}(Y-X\hat{B})))^{\hat{\beta}} \\ & +2pq + p(p+1) + 2.\end{aligned} \tag{5.3}$$

The definition of ICOMP(IFIM) uses the concept of maximal covariance complexity which is defined as:

Definition 5.1 A maximal information theoretic measure of complexity of a covariance matrix Σ of a multivariate normal distribution is

$$\begin{aligned}C_1(\,\Sigma) \equiv \; & \max_T C_0(\Sigma) \\ = \; & \tfrac{p}{2}\log(\tfrac{tr(\,\Sigma)}{p}) - \tfrac{1}{2}\log|\,\Sigma|,\end{aligned} \tag{5.4}$$

where the maximum is taken over the orthonormal transformation T of the overall coordinate system $x_1, x_2, \cdots, x_p$.

For more details on (5.4), we refer the readers to ([9,13,14]). For a multivariate normal linear or nonlinear structural model, we define the general form of ICOMP(IFIM) as

$$ICOMP(IFIM) = -2\log\mathcal{L}(\hat{\theta}) + 2C_1(\hat{\mathcal{F}}^{-1}(\hat{\theta})), \tag{5.5}$$

where C_1 denotes the maximal information-theoretic complexity of $\hat{\mathcal{F}}^{-1}$, the estimated IFIM given in (5.4), and $\hat{\theta}$ is the MLE vector. So, for Type I MVPER model, we have

$$\begin{aligned}& \text{ICOM(IFIM)}_{\text{Model I}} \\ = \; & -2n\log(p\Gamma(\tfrac{p}{2})\Gamma(1+\tfrac{p}{2\hat{\beta}})) + np\log(\pi) + 2n(1+p/2\hat{\beta})\log + n\log 2|\hat{\Sigma}| \\ & + \textstyle\sum_{i=1}^{n}((\mathbf{y}_{(i)}-\hat{B}'\mathbf{x}_{(i)})'\hat{\Sigma}^{-1}(\mathbf{y}_{(i)}-\hat{B}'\mathbf{x}_{(i)}))^{\hat{\beta}} + 2C_1(\hat{\mathcal{F}}^{-1}(\,\hat{\theta})_{Model\ I})\end{aligned} \tag{5.6}$$

and for Type II MVPER model, we have

$$\begin{aligned}& \text{ICOM(IFIM)}_{\text{Model II}} \\ = \; & -2\log(np\Gamma(\tfrac{np}{2})) + np\log(\pi) + 2\log\Gamma(1+\tfrac{np}{2\hat{\beta}}) + (2+np/\hat{\beta})\log 2 \\ & +p\log|\Phi| + n\log|\hat{\Sigma}| + (tr(\hat{\Sigma}^{-1}(Y-X\hat{B})'^{-1}(Y-X\hat{B})))^{\hat{\beta}} \\ & +2C_1(\hat{\mathcal{F}}^{-1}(\,\hat{\theta})_{Model\ II}).\end{aligned} \tag{5.7}$$

The estimated observed inverse Fisher Information matrices for Type I and Type II MVPER models, $\hat{\mathcal{F}}^{-1}(\hat{\theta})_{Model\ I}$ and $\hat{\mathcal{F}}^{-1}(\hat{\theta})_{Model\ II}$, can be computed from the results of Sections 3 and 4 accordingly by

$$\hat{\mathcal{F}}^{-1}(\hat{\theta}) = - \left(\begin{array}{ccc} \frac{\partial l^2(\theta)}{\partial \mathbf{b}' \partial \mathbf{b}} & \frac{\partial l^2(\theta)}{\partial Vec'(\Sigma) \partial \mathbf{b}} & \frac{\partial l^2(\theta)}{\partial \beta \partial \mathbf{b}} \\ \frac{\partial l^2(\theta)}{\partial \mathbf{b}' \partial Vec(\Sigma)} & \frac{\partial l^2(\theta)}{\partial Vec'(\Sigma) \partial Vec(\Sigma)} & \frac{\partial l^2(\theta)}{\partial \beta \partial Vec(\Sigma)} \\ \frac{\partial l^2(\theta)}{\partial \mathbf{b}' \partial \beta} & \frac{\partial l^2(\theta)}{\partial Vec'(\Sigma) \partial \beta} & \frac{\partial l^2(\theta)}{\partial \beta^2} \end{array} \right)_{\hat{\theta}}.$$

Note that the expected Fisher information matrix and its inverse for the Type I and Type II MVPER models involve complicated forms of expected values that is difficult to compute. Therefore, in what follows, it suffices for us to use the complexity of the estimated observed inverse-Fisher Information matrix (IFIM) above in our numerical examples.

6. Genetic Algorithms (GAs)

In this section to be complete and for the benefit of the general readership of the paper, we give the general background and the working of the Genetic Algorithms (GAs) for estimating model parameters and model selection contemporaneously.

Genetic Algorithm (GA) (see, e.g., Goldberg ([19]), Holland ([22]), Mitchell ([27])) is a randomized, population-based heuristic optimization technique that belong to the general class of Evolutionary algorithms (EAs). GA has significant advantages such that it is independent from the complexity of the problem at hand, and not likely to be restricted to a local optimal solution, and it is easy to use in many difficult optimization problems.

Yang and Honavar ([35]) use GA for the selection of a subset of attributes or features to represent the patterns to be classified with Neural network (NN). Bozdogan ([8]) who introduced the GA in statistical model selection, uses the GA in the multiple regression model for subset selection of the best predictors for intelligent data mining under the normality assumption.

In GA, the criterion to rank solutions is often called a *fitness function*. A set of solutions is called a *generation* of *population*. A specific solution is called an *individual* in the population. GA improves solutions by generating a new generation of population on the base of current population through a series of *GA operators*, such as *crossover* and *mutation*. The first generation of population to start the GA process is generated as a set of "wildly' guessed or randomly generated solutions. For the GA to evolve, a solution needs to be represented in binary string format. A binary string represents a solution and it is often called a *chromosome*.

The individuals in the current population are used to generate the new population. There are different strategies to generate a new population. One strategy commonly used is the so-called called "natural" selecting strategy. With this strategy, the chance of an individual being selected is proportional to the ratio:

$$r_j = \Delta Fitness_j / \Delta \overline{Fitness} \tag{6.1}$$

where $\Delta Fitness_j = Fitness_{Max} - Fitness_j$ and $\Delta \overline{Fitness}$ is the mean of $\Delta Fitness_j$. The chance of an individual being selected is proportional to this ratio. In other words, an individual

with a ratio of two is twice as likely to be selected as an individual with a ratio of one. This approach is called the *proportional selection*. There are also other selection strategies such as the *rank order selection,* and so forth.

A pair of individuals selected from the current population are used to generate a pair of new solutions, often called "offsprings", through the GA operator *crossover*. Crossover mimics the process of mating. The pair of chromosomes chosen for crossover is controlled by the *crossover probability* (P_c) which is an input parameter of the algorithm. Crossover point, where the binary string is broken for crossover, is picked randomly along each pair of parent chromosomes. In the algorithm, we give three choices of crossover types corresponding to different locations of crossover points. In the following, "|" represents a crossover point.

- **Single point crossover**

$$\begin{array}{ll} \text{Parent A} & 10|001001001 \\ & \downarrow\uparrow \\ \text{Parent B} & 00|010011000 \end{array} \rightarrow \begin{array}{ll} \text{Offspring A} & 10|010011000 \\ \text{Offspring B} & 00|001001001 \end{array}$$

- **Two point crossover**

$$\begin{array}{ll} \text{Parent A} & 10|001001|001 \\ & \downarrow\uparrow \\ \text{Parent B} & 00|010011|000 \end{array} \rightarrow \begin{array}{ll} \text{Offspring A} & 10|010011|001 \\ \text{Offspring B} & 00|001001|000 \end{array}$$

- **Uniform crossover** - bits are randomly switched between parents:

$$\begin{array}{ll} \text{Parent A} & 10001001001 \\ & \downarrow\uparrow \\ \text{Parent B} & 00010011000 \end{array} \rightarrow \begin{array}{ll} \text{Possible offspring A} & 00000011001 \\ \text{Possible offspring B} & 10011001000 \end{array}$$

Mutation is another parameter or operator used in GA to realize a global search. During mutation, each bit in a binary string can change from 0 to 1, or from 1 to 0, with a user input probability called the *mutation probability* (P_m). Hence, the searching process can jump to another area of the fitness landscape, instead of being limited in a local optimum area.

Our GA also allows the *Elitism rule* (*ER*). When the Elitism rule is applied, the best solution of a generation will be copied without changes to the next generation. *ER* guarantees the individual with the best fitness in the current generation to survive in the next generation. In other words, the best solution is passed from one generation to the next and the survival of the fittest is achieved until the GA converges.

The outline of the GA procedures for model parameter estimation and model selection is summarized as follows:

Step 1: Create a generation of population with a given population size.

Step 2: Encode the individuals into binary strings.

Step 3: Rank each individual in the population according to the given fitness function.

Step 4: Select individuals to be used to generate the new population.

Step 5: Do crossover on selected individuals with a given crossover type and crossover probability and create a new population.

Step 6: Do mutation on the new population with a given mutation probability.

Step 7: Do GA engineering on the new population with a given engineering probability.

Step 8: Do elitism if required. Elitism means that the best individual in current population is guaranteed to be included in the new population.

Step 9: Replace the current population with the new population.

Step 10: Repeat Steps 2-9 until a certain condition of the result is satisfied.

The GA for model parameter estimation and the GA for model selection share the shame outline above, but with different objective functions (fitness functions) and solution representations. This reflects one of the significant advantages of GA method. Once a GA is set up, it can be expanded to solve different problems easily by only changing the fitness function and the representation of solution space. We will explain the fitness function and solution representation for model parameter estimation and model selection in following sections. The *GA engineering* (or *GA cloning*) in Step 7 above is a new GA operator we developed to improve the evolution of the GA process. This will be explained in Section 6.4. The pseudo code of the steps of the GA outlined above is given in the Appendix.

6.1. GA for Model Parameter Estimation

We use GA to estimate the maximum likelihood estimators (MLEs) of the multivariate regression model parameters; the coefficients and the estimated inverse-Fisher information matrix (IFIM) of the model. We use the negative log likelihood function as the fitness function. With this choice, the best solution is the minimum fitness value.

We encode each model parameter to be estimated in a binary string of fixed length, which is given by the investigator as an input. Since the parameters are real numbers, we use the following scheme to encode them. Given a real interval $[a, b]$ and the length of binary string l, the binary string $000\cdots000$ represents a and $111\cdots111$ represents b. Adding binary 1 to an existing binary number increases its real value by $(b-a)/2^{l-1}$. With this approach, decoding a binary string to real number is easy. For example, if $l = 5$, then 10010 represents the real value $a + (b-a) \times 18/31 = (13a + 18b)/31$.

With the above encoding approach, we first obtain the starting point and search the interval with other methods, such as the Method of Moments (MOM). Then, we use the GA to estimate the MLEs.

6.2. GA for Model Selection

In the GA for model selection, we use the ICOMP(IFIM) as the fitness function. The algorithm can also be easily edited to use other model selection criteria. We encode each model using the following scheme.

Each model, or subset, is encoded as a binary string with a fixed length based on the number of total available independent variables (including the constant term). Each bit in the string is a binary code indicating the presence (1) or absence (0) of a given predictor variable in the model. For example, if there are 10 predictor variables available in a given data set, then the string 1010110110 represents a model, where constant term is included in the model, variable 1 is excluded from the model, variable 2 is included in the model, and so on.

6.3. GA on GA Hybridization

We combine and hybridize the GA for parameter estimation and the GA for model selection as follows:

- First, the GA for model selection is called to select the best subset of predictor variables.
- At each step a model is chosen to be evaluated. Then the GA for model parameter estimation is called to obtain the MLEs of the model parameters, or the MLEs are retrieved if the model has been evaluated before. The fitness, i.e., ICOMP(IFIM) of the model is computed using the MLEs of the model parameters.

With the GA on GA approach, the output of the GA for estimation is used in the fitness function (ICOMP) to evaluate the best subset candidate models. The estimation in fact initially acts as the fitness function of the model selection in GA.

The GA on GA approach inherits both advantages and disadvantages of the general GA. But the major convenience of the GA on GA approach is that both the estimation and model selection in GA can share the same GA procedure and code. Only the fitness functions and representations of the solutions need to be changed correspondingly.

The disadvantage of the GA on GA is that, since the output of the GA for model parameter estimation is random, the fitness function of the GA for model selection is also random. In this case, the evaluation of models will be inconsistent during the model selection GA process. For example, in generation i, if model A is better than model B according to their fitness values, but in generation j, model B can be better than model A since their fitness changes. To solve this problem, we remember the fitness and parameter estimation of all evaluated models and retrieve them when they are needed to keep the evaluation process to be consistent. Our simulation results show that this approach is practical with GA properly set up and engineered (or cloned). We further note that the GA on GA approach improves the computational efficiency by eliminating repeated model parameter estimation process when we evaluate and fit the models.

6.4. A New GA Operator: GA Engineering

GA engineering is a new operator we developed and introduced to improve the evolution of the GA process. Since GA is a "random" or "stochastic" search method, it is not guaranteed that each run of the GA with the same settings will converge to the same optimal solution. With classic GA operators and the proper setting of the parameters, the bias and variance of MLEs caused by the GA are usually acceptable according to Chatterjee ([15]). But for the GA on GA approach, the bias and variance caused by GA during the estimation of the model parameters will affect the GA for model selection.

For this reason, we introduce the GA engineering (or GA cloning) operator to improve the estimation further. If the population evolves "*naturally*" with classic GA operators, GA engineering means to improve the quality of the population "*artificially*". The idea of GA engineering comes from the fact that the difference of the fitness values between the two chromosomes is caused by the bits with different binary codes in these two chromosomes and the bit in the chromosome with better fitness are supposed to contain better genes/information. For example, given

$$\begin{array}{lll} \text{Chromosome A} & 100101110 & \\ \text{(Fitness A=10)} & & \\ & & \rightarrow \\ \text{Chromosome B} & 101100101 & \\ \text{(Fitness B=20)} & & \end{array} \quad \begin{array}{ll} \text{Different bits in A} & \sqcup\sqcup 0\sqcup\sqcup 1\sqcup 10 \\ \text{Different bits in B} & \sqcup\sqcup 1\sqcup\sqcup 0\sqcup 01 \end{array} ,$$

$\sqcup\sqcup 0\sqcup\sqcup 1\sqcup 10$ is supposed to have better genes than $\sqcup\sqcup 1\sqcup\sqcup 0\sqcup 01$ if a smaller fitness value is preferred. So, if we compare the best chromosome of generation i with that of generation $i+1$, with probability P_e, and find that they have different binary codes and bits, then we prefer the bits corresponding to the chromosome which has the smaller fitness value, since we are minimizing the information criteria to pick the best model. Indeed, our simulation results show that this new operator does reduce the bias and variance caused by the GA. This we like.

7. Numerical Examples

7.1. Simulation Examples

In the following simulation examples, we use the procedure described by Gómez-Villegas and Sánchez-Manzano et al. ([20]) to generate PE random vectors. The GA on GA approach developed in Bozdogan and Liu ([23]) is used to select predictor variables and to estimate the model parameters. Predictor variables are simulated using the following simulation protocol. The first three predictors are simulated by

$$\begin{aligned} \mathbf{x}_1 &= 10+\varepsilon_1 \\ \mathbf{x}_2 &= 10+0.3\varepsilon_1+\alpha\varepsilon_2, \text{where } \alpha=\sqrt{1-0.3^2}=0.9539 \\ \mathbf{x}_3 &= 10+0.3\varepsilon_1+0.5604\alpha\varepsilon_2+0.8282\alpha\varepsilon_3 \end{aligned} \tag{7.1}$$

where the components of $\varepsilon_1, \varepsilon_2$ and ε_3 are $i.i.d.$ according to $N(0,1)$. The parameter α controls the degree of collinearity in the predictors. Then, we include some redundant variables, $\mathbf{x}_4, \cdots, \mathbf{x}_i$

which are simulated by:

$$\mathbf{x}_4 = 4 \times rand(0,1), \cdots, \mathbf{x}_i = i \times rand(0,1) \tag{7.2}$$

where $rand(0,1)$ generates the uniform random numbers in $(0,1)$.

The response variables are generated from:

$$Y_{n\times 2} = [\mathbf{1}, X_{n\times 3}]B_{4\times 2} + E_{n\times 2} \tag{7.3}$$

with $X = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3]$ and

$$B = \begin{bmatrix} 8 & -5 \\ 1 & 0.5 \\ 0.5 & 0 \\ 0.3 & 0.3 \end{bmatrix}.$$

The error terms in (7.3) are now multivariate PE (MPE) distributed rather than multivariate normal (MN).

7.1.1. Example 1: Model Parameter Estimation and Model Selection

In this example, we generate $\mathbf{x}_1, \mathbf{x}_2$,and $\mathbf{x}_3$ from (7.1) and generate Y from (7.3) with sample size $n = 500$. The rows of E are $i.i.d.\ PE_2(\mathbf{0}, \Sigma, \beta)$ with

$$\Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 2 \end{bmatrix}$$

which is positive definite symmetric. We simulate two cases, $\beta = 0.3$ and $\beta = 2$ with the two dimensional plots given in Figure 1. To these simulated data sets, we fit both multivariate regression model under normality assumption and Type I MVPER model. In this case, we expect that the Type I MVPER model would be chosen as our best model according to the minimum of AIC or ICOMP, and that the model parameters would be estimated correctly, since our true model is generated under the MPE assumption. We are especially interested in the estimates of β. The results of 200 runs of both simulation cases are reported in Table 1.

Table 1: Results of the Simulation Example 1.

Model	Avg. β_{mom}	Avg. β_{mle}	Avg. AIC	Avg. ICOMP
Real $\beta = 0.3$				
Normal	1	1	10487	10529
Type I MVPER	0.3312	0.3618	10154	10193
Real $\beta = 2$				
Normal	1	1	3015.7	3021.3
Type I MVPER	1.9668	2.4140	2840.3	2860.0

From Table 1, we see that in all 200 runs of the simulation, the true model, i.e., the Type I MVPER model, is chosen by both AIC and ICOMP criteria. Further, the estimates of the shape

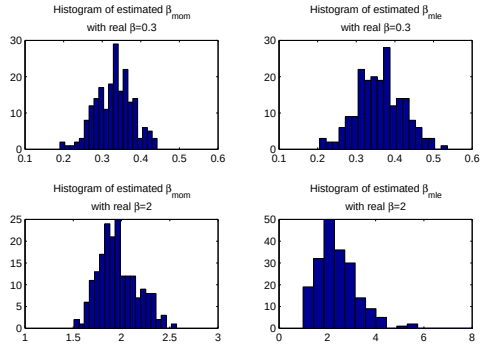

Figure 3: The MOM Estimates and MLEs of β in 200 Runs of Simulation Example 1.

parameter β are close to the true values. The multivariate normal regression model has never been chosen by looking at the average values of AIC and ICOMP across the 200 runs of the simulation. The distributions of MOM estimates and MLEs of β in the 200 runs of the simulation are shown in Figure 3. The Q-Q plots for the Type I MVPER model in one run of the simulation example are shown in Figure 4. From the Q-Q plots, we see that the Type I models are far from normal models for both $\beta = 0.3$ and $\beta = 2$, which we expected. Figure 5 shows a typical GA process used to estimated the model parameters in one run of the simulation. The GA parameters are given in Table 2, where NG is the number of generations, PS is the population size, P_c is the crossover probability, P_m is the mutation probability, P_e is the GA engineering probability, CT is the crossover type, l is the length of the binary string used to encode real numbers, and NR is the number of GA runs. $CT = 3$ means uniform crossover method is used.

Table 2: GA parameters of the Simulation Example 1.

GA parameters	NG	PS	P_c	P_m	P_e	CT	l	NR	Elitisim
Values	50	100	0.7	0.1	0.5	3	32	200	Yes

7.1.2. Example 2: Subset selection

In this example, we show the subset selection of the best predictors under Type I MVPER model with AIC and ICOMP(IFIM). In this simulation, three correct predictors need to be selected from total of ten available predictor variables, where $\mathbf{x}_4, \cdots, \mathbf{x}_{10}$ are considered as redundant variables. We generate $\mathbf{x}_1, \cdots, \mathbf{x}_{10}$ from (7.1) and (7.2), and generate Y from (7.3) with sample size $n = 500$. The rows of E are $i.i.d.\ PE_2(\mathbf{0}, \Sigma, \beta)$ with

$$\Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 2 \end{bmatrix},$$

and we set $\beta = 2$. We fit a Type I MVPER model of Y on $[\mathbf{1}, X_{n\times 10}]$. We expect the algorithm to pick the subset $\{x_0, x_1, x_2, x_3\}$ to be the best subset selected using the minimum AIC or ICOMP(IFIM) criteria, where x_0 denotes the constant term. Parameters of GA are given in Table

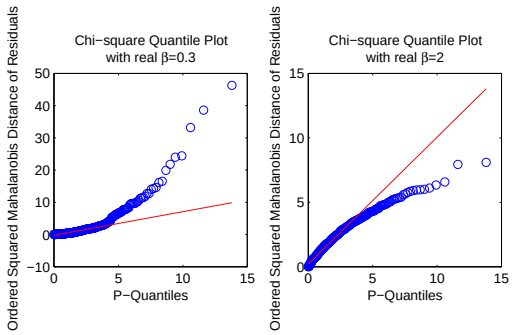

Figure 4: The Q-Q plots for Type I MVPER Models in One Run of the Simulation Example 1.

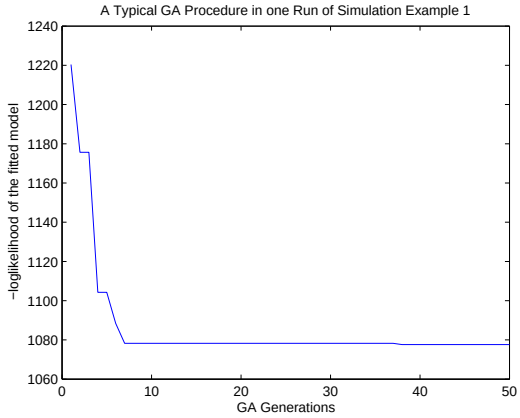

Figure 5: A Typical GA Process in One Run of the Simulation Example 1.

3. Method of moments (MOM) estimates are used as the starting values for the GA process. For 11 independent variables (including the constant term), there are $2^{11} = 2048$ possible subsets. Each subset, or model, is encoded as a binary string with the fixed length as the total number of available independent or predictor variables. Each locus in the string is a binary code indicating the presence (1) or absence (0) of a given predictor variable in the model. For example, the string 101011 represents a model, where constant term is included in the model, variable x_1 is excluded from the model, variable x_2 is included in the model, and so on. Figure 6 shows the plots of all subsets evaluated in the GA process. Both GA processes converge to the true model. The top 5 subsets selected by the minimum ICOMP(IFIM) and AIC are summarized in Table 4. From Table 4, we see that the true model is selected as the best subset according to minimum ICOMP(IFIM) and AIC. The MLEs of the best model parameters chosen by ICOMP are:

$$\hat{\beta}_{MLE} = 1.7488,$$

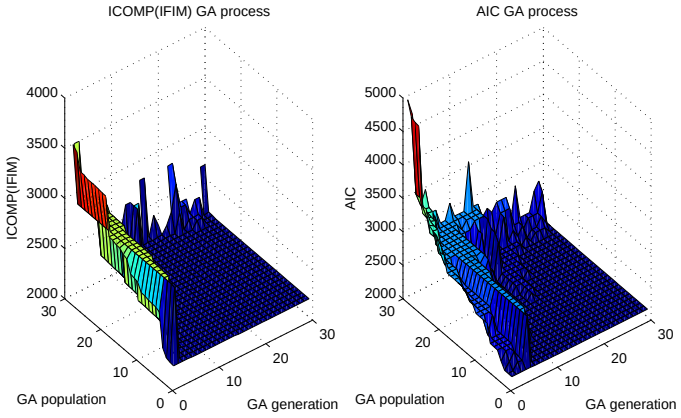

Figure 6: The GA Process for the Simulation Example 2 for ICOMP and AIC.

$$\hat{\Sigma}_{MLE} = \begin{bmatrix} 0.8788 & 0.4007 \\ 0.4007 & 1.7192 \end{bmatrix}$$

and

$$\hat{B}_{MLE} = \begin{bmatrix} 7.2275 & -5.4046 \\ 1.0004 & 0.5128 \\ 0.5142 & 0.0951 \\ 0.3587 & 0.2335 \end{bmatrix}.$$

Our results show that the GA on GA approach with ICOMP(IFIM) or AIC as the fitness function can detect the true relationship and pick the correct models. We notice that the coefficients of the redundant variables in the top 5 best subset models selected are small, which means that these redundant variables can be ignored.

Table 3: GA parameters of the Simulation Example 2 and the Real Data Example.

	NG	PS	P_c	P_m	P_e	CT	l	Elitism
Subset GA	30	30	0.7	0.01	0.5	3		Yes
Model estimation GA	50	100	0.7	0.01	0.5	3	32	Yes

7.2. A Real Example: A Macroeconomic Time Series Data

In this last example, we use the famous quarterly macroeconomic time series data for the United Kingdom during 1948-1956. This data set consists of $n = 36$ quarterly observations, starting with the first quarter of 1948 and ending with the last quarter of 1956. All the $n = 36$ observations are used in our analysis. The descriptions of 5 response variables and 5 independent variables from Klein et al. (1961) are given in Table 5. Now for this data set, we fit both Type I and Type II MVPER models to determine the best fitting model. For Type II MVPER model , we assume $\Phi = I_n$ with sample size $n = 36$. Method of moments (MOM) estimates are used

Table 4: Top 5 subsets according to minimum AIC and ICOMP(IFIM).

Ranking	1	2	3	4	5
Subsets selected by ICOMP(IFIM)					
Subset	11110000000	11110100000	11111000000	11110000100	11110001000
$\hat{\beta}_{MLE}$	1.7488	1.7472	1.7464	1.7456	1.7408
ICOMP	2182.4	2187.7	2188.4	2188.8	2190.1
Subsets selected by AIC					
Subset	11110000000	11110000100	11110000010	11110000110	11111000000
$\hat{\beta}_{MLE}$	1.9263	1.8944	1.9408	1.9055	1.9082
AIC	2118.5	2119.2	2120.0	2120.8	2121.1

as the starting values for the GA process. AIC and ICOMP(IFIM) criteria developed in Section 5 are used for subset selection of the best predictors. The top 5 best subsets selected according to AIC and ICOMP(IFIM) scores by fitting Type I and Type II MVPER models on the Klein data set. The results are summarized in Tables 6 and 7, respectively. We can see that the results of AIC and ICOMP(IFIM) are slightly different for each type of MVPER models for this data set. For Type I MVPER model, both AIC and ICOMP(IFIM) criteria select the binary string 000001 as the optimal model. In other words, the predictor variable $x_5 = price\ index\ of\ consumption$ is the best predictor to predict all the response variable Y with $\hat{\beta}_{MLE} = 1.3099$ which indicates that this data set is not normal.

The MLEs of B and Σ for the optimal model are:

$$\hat{B}_{MLE,Model\ I} = \begin{pmatrix} 1.0084 & 0.9151 & 0.8590 & 0.9793 & 1.0597 \end{pmatrix}$$

and

$$\hat{\Sigma}_{MLE,Model\ I} = \begin{pmatrix} 84.9514 & 9.1643 & 6.1158 & 9.5936 & 15.5155 \\ 9.1643 & 91.5875 & 138.4841 & 3.8669 & -18.4478 \\ 6.1158 & 138.4841 & 576.3684 & -55.7333 & -54.1991 \\ 9.5936 & 3.8669 & -55.7333 & 34.2577 & -4.7476 \\ 15.5155 & -18.4478 & -54.1991 & -4.7476 & 81.4006 \end{pmatrix}.$$

For Type II MVPER model, both AIC and ICOMP(IFIM) select 110000 as the optimal model. That is, the constant term x_0 and the predictor variable $x_1 = total\ labor\ force$ are chosen as the best subset of predictors.

The MLEs of B and Σ for the optimal model are

$$\hat{B}_{MLE,Model\ II} = \begin{pmatrix} -462.5338 & -115.9112 & 473.4929 & -417.9496 & -505.6549 \\ 5.6339 & 2.1539 & -3.6246 & 5.1773 & 6.0989 \end{pmatrix}$$

and

$$\hat{\Sigma}_{MLE,Model\ II} = \begin{pmatrix} 496.1 & 171.4 & -729.1 & 263.0 & 256.7 \\ 171.4 & 176.3 & -188.7 & 25.6 & 26.7 \\ -729.1 & -188.7 & 9812.5 & -1485.6 & -765.4 \\ 263.0 & 25.6 & -1485.6 & 973.4 & 275.0 \\ 256.7 & 26.7 & -765.4 & 275.0 & 1479.4 \end{pmatrix}.$$

Table 5: Variables of the Klein data set.

response variables	*independent variables*
$y_1 =$ *industrial production*	$x_1 =$ *total labor force*
$y_2 =$ *consumption*	$x_2 =$ *weekly wage rates*
$y_3 =$ *unemployment*	$x_3 =$ *price index of imports*
$y_4 =$ *total imports*	$x_4 =$ *price index of exports*
$y_5 =$ *total exports*	$x_5 =$ *price index of consumption*

Table 6: Top 5 ranking subsets selected under Type I MVPER Model.

Ranking	1	2	3	4	5
Subsets selected by AIC					
Subset	000001	101000	010000	000100	000010
$\hat{\beta}_{MOM}$	1.0641	0.7963	0.7980	1.0177	1.0925
$\hat{\beta}_{MLE}$	1.3099	0.5446	0.5574	1.0446	0.7016
AIC	1379.2	1483.1	1498.3	1521.5	1527.6
Subsets selected by ICOMP(IFIM)					
Subset	000001	101000	000100	000010	010000
$\hat{\beta}_{MOM}$	1.0641	0.7963	1.0177	1.0925	0.7980
$\hat{\beta}_{MLE}$	1.3099	0.5446	1.0446	0.7016	0.5574
ICOMP	1495.0	1540.6	1616.0	1637.0	1655.5

From the results in Tables 6 and 7, we see that the AIC and ICOMP(IFIM) values for the best Type II model are much smaller than those for the best Type I model. So according to the minimum value of both AIC and ICOMP(IFIM) criteria, Type II MVPER model will be selected as the best fitting model for the Klein data set. We note that such a choice is in agreement with our prior knowledge about this data set in that the observations actually are not independent. They are dependent since this data set is time dependent. We can also see that for both Type I and Type II MVPER models, the MLEs of β are different from one another, which means that the residuals are non-normal for the Klein data set. Here we use Q-Q plots to test the normality of the residuals. For the Type I model, if the residuals are i.i.d. multivariate normally distributed, the squared Mahalanobis distance of each residual vector, $d_i = \varepsilon_{(i)}^{\prime -1} S^{-1} \varepsilon_{(i)}$, will be distributed approximately as χ^2 with p degrees of freedom, where $p = 5$ is the number of dependent variables in the model. So the Q-Q plot for Type I model is the ordered distance values d_i against the corresponding theoretical quantiles of $\chi^2(5)$ distribution. For the Type II model, we first vectorize the residuals and then do the plot as the classic normal Q-Q plot. The Q-Q plots are shown in Figures 7 and 8, respectively. From the graphs, we see that the Type I model is closer to the normal model according to the estimated β values. However, we see that the best fitting Type II model does not follow the normal distribution based on the estimated β values. Indeed, this data set has serial correlations which causes the fact that the probability distribution of the model is misspecified. Type II model captures such misspecification as a general and flexible model which

[†]The GA is setup to search β in [0.001 10]. If the MLE of β is very near to 10, we consider the estimate is ∞. If the MOM of β is very near to 10, it means the MOM of β does not exist.

Table 7: Top 5 ranking subsets selected under Type II MVPER Model.

Ranking	1	2	3	4	5
Subsets selected by AIC					
Subset	110000	110100	110001	110010	111001
$\hat{\beta}_{MOM}$	0.0041	0.0083	0.0077	10.0000†	9.9994
$\hat{\beta}_{MLE}$	2.5533	2.5115	2.2101	2.2664	2.1740
AIC	632.8	638.9	658.7	661.4	666.3
Subsets selected by ICOMP(IFIM)					
Subset	110000	100000	110100	110010	110001
$\hat{\beta}_{MOM}$	0.0041	4.1282	0.0083	10.0000	0.0077
$\hat{\beta}_{MLE}$	2.5533	2.2379	2.5115	2.2664	2.2101
ICOMP	939.3	1018.0	1034.1	1043.8	1044.8

takes the dependency structure of the data into account.

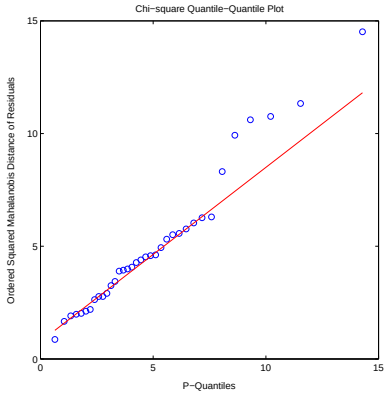

Figure 7: Q-Q plot under the Type I model.

8. Conclusion

In this paper we presented a new and novel model selection technique to deal with non-normality in multivariate regression models using information-theoretic model selection criteria such as AIC and ICOMP. We developed two types of MVPER models. These two types of MVPER models discussed in this paper can be used to model random phenomena whose observations are *dependent* or *independent* when the tails are *thicker* or *thinner* than those of multivariate normal distribution which is used often in the literature. As a subfamily of matrix EC distribution, Type II model has been studied partly in the context of multivariate EC regression model such as the work of Fang and Anderson ([17, p.214]) and Bozdogan ([11]), etc. But the Type I model is seldom discussed in literature. One special difficulty of MVPER models is to estimate the shape parameter β. In this paper, we provided methods to obtain MOM estimates and MLEs for both types of models. From the above results, we see that the MLEs of the Type II model actually can

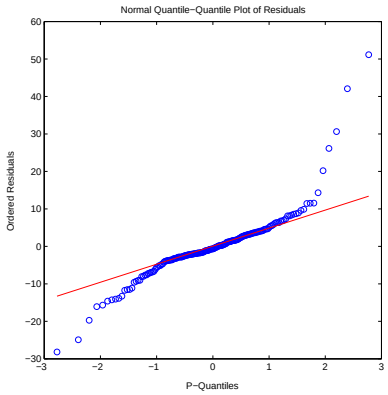

Figure 8: Q-Q plot under the Type II model.

not be obtained from a single sample. But a two step re-sampling method is used and developed to solve this problem. Our simulated as well as the real computational examples show that the hybrid of information criteria such as AIC and ICOMP(IFIM) and the GA approach works well for model selection in both cases. Advantages of this approach introduced in this paper are flexible to resolve many problems in vector autoregressive (VAR) models, in kernel support vector machines (SVMs), etc. by taking the dependency in the data into account.

All our computations are carried out using a newly developed computational MATLAB modules with GA.

Acknowledgements

This paper is based on the results of the Ph.D. thesis of the first author under the supervision of Prof. Bozdogan. First author extends his thanks and gratitude to Prof. Bozdogan for all the help provided throughout. This paper has been presented by Prof. Bozdogan as an invited paper at *Bayes, Multivariate Analysis, and CASM Conference* in Honor of Professor S. Jim Press, Distinguished University Professor, at the University of California at Riverside, California during May 13-14, 2005. We are deeply honored to dedicate this paper to Professor Jim Press in honor of his 50 years of remarkable career and scientific contributions to *Bayesian* and *Multivariate Statistics* on the occasion of his retirement. Professor Bozdogan gratefully acknowledges funding for his research from the Scholarly Research Grant Program (SRGP) Awards of the College of Business Administration at the University of Tennessee in Knoxville, during 2002-03 under the title *Multivariate Regression Model with Nonnormal Error Terms, Genetic Algorithm and Information Complexity*. We also extend our thanks and deep gratitude to Prof. Dr. Eyüp Çetin, Editor-in-Chief of EJPAM, for inviting Prof. Bozdogan to make an contribution to the "Honorary Invited Paper" issue of EJPAM. We are privileged to make this contribution in the opening issue and wish the success of this prestigious journal.

The Pseudo Code of Genetic Algorithm

Algorithm 1: The pseudo code of GA for regression model selection

PS=Population size;
NG=Number of generations of GA;
P_c=Crossover probability;
P_m=Mutation probability;
P_e=GA engineering probability;
CT=Crossover type;
$i = 1$;
Generate PS original solutions randomly;
Encode the solutions to binary strings (chromosomes);
For $i \leq NG$
  Evaluate the fitness of each solution of generation i;
  Find the best solution in generation i;
  If elitism is true **Then**
    Select $NG/2 - 1$ pairs of parent chromosomes from generation i;
  Else
    Select $NG/2$ pairs of parent chromosomes from generation i;
  End
  Cross over each pair of parent chromosomes with probability P_c;
  If $CT = 1$ **Then**
    Do single point crossover;
  Else If $CT = 2$ **Then**
      Do two point crossover;
  Else
      Do uniform crossover;
  End
  Mutate new offspring at each lotus with probability P_m;
  Engineer the new offspring with probability P_e;
  If elitism is true **Then**
    Add the best chromosome in generation i to new offspring;
    Add a randomly selected chromosome in generation i to new offspring;
  End
  $i = i + 1$;
  Decode the new offspring to solutions;
  Replace the chromosomes in generation i with new offspring;
  Replace the solutions in generation i with new solutions;
  If any special final condition is satisfied **Then**
    Exit;
  End

End

Generating Univariate PE Pseudo-Random Numbers

Let $Y = \frac{1}{2}|X|^{2\beta}$, where X has a standard PE distribution with shape parameter β. Then

$$\begin{aligned} P(Y < y) &= P(|X|^{2\beta} < 2y) \\ &= P(-(2y)^{1/2\beta} < X < (2y)^{1/2\beta}) \\ &= \int_{-(2y)^{1/2\beta}}^{(2y)^{1/2\beta}} \frac{1}{\Gamma\left(1+\frac{1}{2\beta}\right)2^{1+\frac{1}{2\beta}}} \exp\left(-\frac{1}{2}\,|x|^{2\beta}\right) dx \\ &= \int_{0}^{(2y)^{1/2\beta}} \frac{1}{\Gamma\left(1+\frac{1}{2\beta}\right)2^{\frac{1}{2\beta}}} \exp\left(-\frac{1}{2}x^{2\beta}\right) dx \\ &= \int_{0}^{y} \frac{t^{1/2\beta-1}}{\Gamma\left(\frac{1}{2\beta}\right)} \exp\left(-t\right) dt. \end{aligned} \tag{8.1}$$

So Y has a distribution with density

$$f(y) = \frac{y^{(1/2\beta)-1}e^{-y}}{\Gamma(1/2\beta)}, \tag{8.2}$$

i.e., $Y \sim Gamma(1/2\beta)$. One method generating pseudo-random PE variables is given as follows:

1. Generate ordinates from a Gamma distribution with density (8.2);
2. Generate B from a Bernoulli distribution with $p = 1/2$;
3. If $B = 0$, then generate $X = (2Y)^{1/2\beta}$, otherwise, generate $X = -(2Y)^{1/2\beta}$;
4. Generate $Z = \sigma X + \mu$.

Histograms of random samples of size 1000 generated from above algorithm are given in Figure 9.

Matrix Calculus for Type I MVPER Model

$$\begin{aligned} \frac{\partial \varepsilon_{(i)}}{\partial B} &= \frac{\partial(\mathbf{y}_{(i)} - B'\mathbf{x}_{(i)})}{\partial B} \\ &= \frac{\partial \mathbf{y}_{(i)}}{\partial B} - \frac{\partial B'\mathbf{x}_{(i)}}{\partial B} \\ &= -\frac{\partial B'\mathbf{x}_{(i)}}{\partial B} \\ &= -I_{(q,p)}(\mathbf{x}_{(i)} \otimes I_p) \end{aligned} \tag{8.3}$$

where $I_{(q,p)}$ is the *permuted identity* Macrae ([24]) or *commutation matrix* Magnus and Neudecker ([25]).

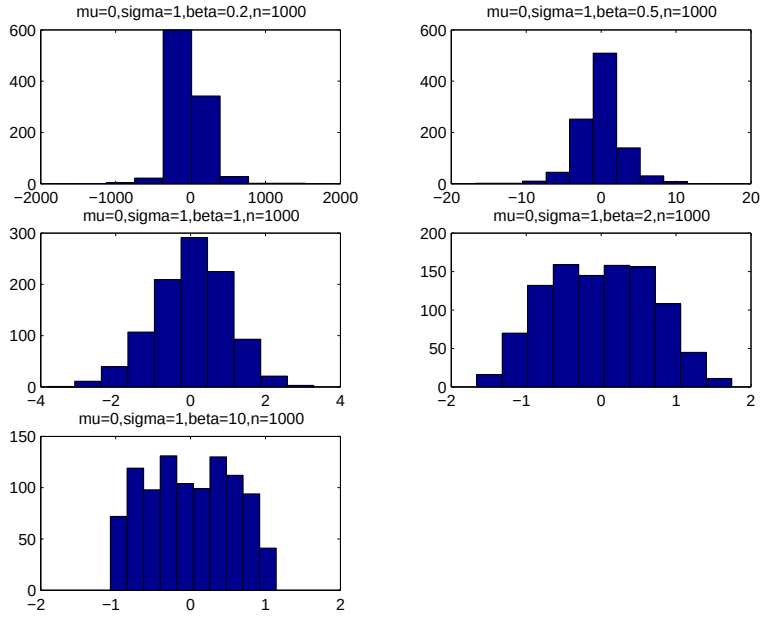

Figure 9: Histograms of PE pseudo-random samples.

$$\begin{aligned}
\frac{\partial \varepsilon_{(i)}}{\partial B'} &= \frac{\partial (\mathbf{y}_{(i)} - B'\mathbf{x}_{(i)})}{\partial B'} \\
&= \frac{\partial \mathbf{y}_{(i)}}{\partial B'} - \frac{\partial B'\mathbf{x}_{(i)}}{\partial B'} \\
&= -\frac{\partial B'\mathbf{x}_{(i)}}{\partial B'} \\
&= -Vec(I_p)Vec'(\mathbf{x}_{(i)})
\end{aligned} \tag{8.4}$$

$$\begin{aligned}
\frac{\partial t_i}{\partial B} &= \frac{\partial \varepsilon_{(i)}'^{-1}\varepsilon_{(i)}}{\partial B} \\
&= \frac{\partial \varepsilon_{(i)}'^{-1}\varepsilon_{(i)}}{\partial \varepsilon_{(i)}} * \frac{\partial \varepsilon_{(i)}}{\partial B} \\
&= 2\Sigma^{-1}\varepsilon_{(i)} * (-I_{(q,p)}(\mathbf{x}_{(i)} \otimes I_p)) \\
&= -2\Sigma^{-1}\varepsilon_{(i)} * (I_{qp}I_{(q,p)}(\mathbf{x}_{(i)} \otimes I_p)) \\
&= -2\Sigma^{-1}\varepsilon_{(i)} * ((I_p \otimes I_q)I_{(q,p)}(\mathbf{x}_{(i)} \otimes I_p)) \\
&= -2\mathbf{x}_{(i)}(\Sigma^{-1}\mathbf{s}_i)'I_p \\
&= -2\mathbf{x}_{(i)}\varepsilon_{(i)}'^{-1}
\end{aligned} \tag{8.5}$$

where $*$ is the *star product* ([24]).

$$\begin{aligned}
\frac{\partial t_i}{\partial B'} &= \frac{\partial \varepsilon_{(i)}'^{-1}\varepsilon_{(i)}}{\partial B'} \\
&= \frac{\partial \varepsilon_{(i)}'^{-1}\varepsilon_{(i)}}{\partial \varepsilon_{(i)}} * \frac{\partial \varepsilon_{(i)}}{\partial B'} \\
&= 2\Sigma^{-1}\varepsilon_{(i)} * (-Vec(I_p)Vec'(\mathbf{x}_{(i)})) \\
&= -2\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)}
\end{aligned} \tag{8.6}$$

$$\begin{aligned}\frac{\partial l(\theta)}{\partial B} &= -\tfrac{1}{2}\sum_{i=1}^{n}\frac{\partial t_i^{\beta}}{\partial B}\\ &= -\tfrac{1}{2}\sum_{i=1}^{n}\frac{\partial t_i^{\beta}}{\partial t_i}\frac{\partial t_i}{\partial B}\\ &= \beta\sum_{i=1}^{n}t_i^{\beta-1}\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1}\\ &= \beta\sum_{i=1}^{n}(((\mathbf{y}_{(i)}-B'\mathbf{x}_{(i)})'^{-1}(\mathbf{y}_{(i)}-B'\mathbf{x}_{(i)}))^{\beta-1}\mathbf{x}_{(i)}(\mathbf{y}_{(i)}-B'\mathbf{x}_{(i)})'^{-1}),\end{aligned} \tag{8.7}$$

$$\begin{aligned}\frac{\partial l(\theta)}{\partial \mathbf{b}} &= \frac{\partial l(\theta)}{\partial Vec(B')}\\ &= vec\frac{\partial l(\theta)}{\partial B'}\\ &= Vec(\frac{\partial l(\theta)}{\partial B})'\\ &= \beta\sum_{i=1}^{n}t_i^{\beta-1}Vec(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')\end{aligned} \tag{8.8}$$

$$\begin{aligned}\frac{\partial^2 l(\theta)}{\partial B'\partial B} &= \frac{\partial}{\partial B'}(\frac{\partial l(\theta)}{\partial B})\\ &= \frac{\partial}{\partial B'}(\beta\sum_{i=1}^{n}t_i^{\beta-1}\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})\\ &= \beta\sum_{i=1}^{n}\frac{\partial(t_i^{\beta-1}\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})}{\partial B'}\\ &= \beta\sum_{i=1}^{n}(t_i^{\beta-1}\frac{\partial(\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})}{\partial B'}+(\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})\otimes\frac{\partial t_i^{\beta-1}}{\partial B'})\\ &= \beta\sum_{i=1}^{n}(t_i^{\beta-1}(\mathbf{x}_{(i)}\otimes I_p)\frac{\partial\varepsilon_{(i)}'}{\partial B'}(\Sigma^{-1}\otimes I_q)+(\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})\otimes(\frac{\partial t_i^{\beta-1}}{\partial t_i}\frac{\partial t_i}{\partial B'}))\\ &= \beta\sum_{i=1}^{n}(-t_i^{\beta-1}(\mathbf{x}_{(i)}\otimes I_p)(\mathbf{x}_{(i)}'\otimes I_p)I_{(p,q)}(\Sigma^{-1}\otimes I_q)\\ &\quad -2(\beta-1)t_i^{\beta-2}(\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})\otimes(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}'))\\ &= -\beta\sum_{i=1}^{n}(t_i^{\beta-1}(\mathbf{x}_{(i)}\mathbf{x}_{(i)}^{\prime-1})\\ &\quad +2(\beta-1)t_i^{\beta-2}(\mathbf{x}_{(i)}\varepsilon_{(i)}^{\prime-1})\otimes(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}'))\end{aligned} \tag{8.9}$$

$$\begin{aligned}\frac{\partial(t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')}{\partial B'} &= t_i^{\beta-1}\frac{\partial(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')}{\partial B'}+(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')\otimes\frac{\partial t_i^{\beta-1}}{\partial B'}\\ &= t_i^{\beta-1}(\Sigma^{-1}\otimes I_p)\frac{\partial\varepsilon_{(i)}}{\partial B'}(\mathbf{x}_{(i)}'\otimes I_q)+(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')\otimes\frac{\partial t_i^{\beta-1}}{\partial B'}\\ &= -t_i^{\beta-1}(\Sigma^{-1}\otimes I_p)Vec(I_p)Vec'(\mathbf{x}_{(i)})(\mathbf{x}_{(i)}'\otimes I_q)\\ &\quad -(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')\otimes(\frac{\partial t_i^{\beta-1}}{\partial t_i}\frac{\partial t_i}{\partial B'})\\ &= -t_i^{\beta-1}Vec(\Sigma^{-1})Vec'(\mathbf{x}_{(i)}\mathbf{x}_{(i)}')\\ &\quad -2(\beta-1)t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}^{\prime-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')\end{aligned} \tag{8.10}$$

$$\begin{aligned}
\frac{\partial^2(\theta)}{\partial \mathbf{b}'\partial \mathbf{b}} &= \frac{\partial}{\partial \mathbf{b}'}(\frac{\partial l(\theta)}{\partial \mathbf{b}}) \\
&= \beta \sum_{i=1}^{n} \frac{\partial Vec(t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)})}{\partial \mathbf{b}'} \\
&= \beta \sum_{i=1}^{n} \frac{\partial Vec(t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)})}{\partial Vec'(B')} \\
&= \beta \sum_{i=1}^{n} (I_{pq} \otimes Vec'(I_p))(I_q \otimes \frac{\partial (t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'_{(i)})}{\partial B'} \otimes I_p)(Vec(I_q) \otimes I_{pq}) \\
&= \beta \sum_{i=1}^{n} (I_{pq} \otimes Vec'(I_p))(I_q \otimes M_i \otimes I_p)(Vec(I_q) \otimes I_{pq})
\end{aligned} \tag{8.11}$$

where

$$M_i = -t_i^{\beta-1} Vec(\Sigma^{-1}) Vec'(\mathbf{x}_{(i)}\mathbf{x}'_{(i)}) - 2(\beta-1)t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}'^{-1}_{(i)}\varepsilon_{(i)}\mathbf{x}'_{(i)}).$$

$$\begin{aligned}
\frac{\partial t_i}{\partial \Sigma} &= \frac{\partial(\varepsilon'^{-1}_{(i)}\varepsilon_{(i)})}{\partial \Sigma} \\
&= -\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}
\end{aligned} \tag{8.12}$$

$$\begin{aligned}
\frac{\partial l(\theta)}{\partial \Sigma} &= -\frac{n}{2}\frac{\partial \log|\Sigma|}{\partial \Sigma} - \frac{1}{2}\sum_{i=1}^{n}\frac{\partial t_i^{\beta}}{\partial \Sigma} \\
&= -\frac{n}{2}\Sigma^{-1} - \frac{1}{2}\sum_{i=1}^{n}\frac{\partial t_i^{\beta-1}}{\partial t_i}\frac{\partial t_i}{\partial \Sigma} \\
&= -\frac{n}{2}\Sigma^{-1} - \frac{1}{2}\sum_{i=1}^{n}\beta t_i^{\beta-1}\frac{\partial t_i}{\partial \Sigma} \\
&= -\frac{n}{2}\Sigma^{-1} + \frac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}.
\end{aligned} \tag{8.13}$$

$$\begin{aligned}
\frac{\partial(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})}{\partial \Sigma} &= \frac{\partial \Sigma^{-1}}{\partial \Sigma}(\varepsilon_{(i)}\varepsilon'^{-1}_{(i)} \otimes I_p) + (\Sigma^{-1} \otimes I_p)(\varepsilon_{(i)}\varepsilon'_{(i)} \otimes I_p)\frac{\partial \Sigma^{-1}}{\partial \Sigma} \\
&= -Vec(\Sigma^{-1})Vec'^{-1})(\varepsilon_{(i)}\varepsilon'^{-1}_{(i)} \otimes I_p) \\
&\quad -(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'_{(i)} \otimes I_p)Vec(\Sigma^{-1})Vec'^{-1}) \\
&= -Vec(\Sigma^{-1})Vec'^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}) \\
&\quad -Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})Vec'^{-1})
\end{aligned} \tag{8.14}$$

$$\begin{aligned}
\frac{\partial^2 l(\theta)}{\partial \Sigma'\partial \Sigma} &= \frac{\partial}{\partial \Sigma'}(\frac{\partial l(\theta)}{\partial \Sigma}) \\
&= -\frac{n}{2}\frac{\partial \Sigma^{-1}}{\partial \Sigma} + \frac{\beta}{2}\sum_{i=1}^{n}\frac{\partial(t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})}{\partial \Sigma} \\
&= \frac{n}{2}Vec(\Sigma^{-1})Vec'^{-1}) \\
&\quad +\frac{\beta}{2}\sum_{i=1}^{n}(t_i^{\beta-1}\frac{\partial(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})}{\partial \Sigma} + (\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}) \otimes \frac{\partial t_i^{\beta-1}}{\partial \Sigma}) \\
&= \frac{n}{2}Vec(\Sigma^{-1})Vec'^{-1}) \\
&\quad -\frac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1}(Vec(\Sigma^{-1})Vec'^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}) \\
&\quad +Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)})Vec'^{-1})) \\
&\quad -\frac{\beta(\beta-1)}{2}\sum_{i=1}^{n}(t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}) \otimes (\Sigma^{-1}\varepsilon_{(i)}\varepsilon'^{-1}_{(i)}))
\end{aligned} \tag{8.15}$$

$$\begin{aligned}\frac{\partial l(\theta)}{\partial Vec(\Sigma)} &= vec(\frac{\partial l(\theta)}{\partial \Sigma}) \\ &= -\frac{n}{2}Vec(\Sigma^{-1}) + \frac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1} Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon_{(i)}'^{-1}).\end{aligned} \tag{8.16}$$

$$\frac{\partial^2 l(\theta)}{\partial Vec'(\Sigma)\partial Vec(\Sigma)} = (I_{p^2} \otimes Vec'(I_p))(I_p \otimes \frac{\partial^2 l(\theta)}{\partial \Sigma' \partial \Sigma} \otimes I_p)(Vec(I_p) \otimes I_{p^2}) \tag{8.17}$$

$$\frac{\partial l(\theta)}{\partial \beta} = \frac{np}{2\beta^2}\psi(1+\frac{p}{2\beta}) + \frac{np}{2\beta^2}\log 2 - \frac{1}{2}\sum_{i=1}^{n} t_i^{\beta}\log t_i \tag{8.18}$$

where $\psi(\cdot) = \mathrm{d}\log\Gamma(x)/\mathrm{d}x$ is called digamma function Abramowitz and Stegun ([1]), or psi function.

$$\frac{\partial^2 l(\theta)}{\partial \beta^2} = -\frac{np}{\beta^3}\psi(1+\frac{p}{2\beta}) - \frac{np^2}{4\beta^4}\psi'(1+\frac{p}{2\beta}) - \frac{np}{\beta^3}\log 2 - \frac{1}{2}\sum_{i=1}^{n} t_i^{\beta}\log^2 t_i \tag{8.19}$$

where $\psi'(\cdot) = \mathrm{d}^2\log\Gamma(x)/\mathrm{d}x^2$ is called trigamma function.

$$\begin{aligned}\frac{\partial^2 l(\theta)}{\partial \beta \partial \mathbf{b}} &= \frac{\partial}{\partial \beta}(\frac{\partial l(\theta)}{\partial \mathbf{b}}) \\ &= \sum_{i=1}^{n} t_i^{\beta-1} Vec(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}') + \beta\sum_{i=1}^{n} t_i^{\beta-1}\log(t_i) Vec(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')\end{aligned} \tag{8.20}$$

$$\begin{aligned}\frac{\partial^2 l(\theta)}{\partial \beta \partial Vec(\Sigma)} &= \frac{\partial}{\partial \beta}(\frac{\partial l(\theta)}{\partial Vec(\Sigma)}) \\ &= \frac{1}{2}\sum_{i=1}^{n} t_i^{\beta-1} Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon_{(i)}'^{-1}) \\ &\quad + \frac{\beta}{2}\sum_{i=1}^{n} t_i^{\beta-1}\log(t_i) Vec(\Sigma^{-1}\varepsilon_{(i)}\varepsilon_{(i)}'^{-1}).\end{aligned} \tag{8.21}$$

$$\begin{aligned}\frac{\partial^2 l(\theta)}{\partial Vec'(\Sigma)\partial \mathbf{b}} &= \frac{\partial}{\partial Vec'(\Sigma)}(\frac{\partial l(\theta)}{\partial \mathbf{b}}) \\ &= \beta\sum_{i=1}^{n} \frac{\partial Vec(t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')}{\partial Vec'(\Sigma)} \\ &= \beta\sum_{i=1}^{n} (I_{pq} \otimes Vec'(I_p))(I_q \otimes N_i \otimes I_p)(Vec(I_q) \otimes I_{p^2})\end{aligned} \tag{8.22}$$

where

$$\begin{aligned}N_i &= \frac{\partial(t_i^{\beta-1}\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}')}{\partial \Sigma} \\ &= t_i^{\beta-1}\frac{\partial \Sigma^{-1}}{\partial \Sigma}(\varepsilon_{(i)}\mathbf{x}_{(i)}' \otimes I_p) + (\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}') \otimes \frac{\partial t_i^{\beta-1}}{\partial \Sigma} \\ &= -t_i^{\beta-1} Vec(\Sigma^{-1}) Vec'^{-1})(\varepsilon_{(i)}\mathbf{x}_{(i)}' \otimes I_p) \\ &\quad -(\beta-1)t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}'^{-1}\varepsilon_{(i)}\varepsilon_{(i)}'^{-1}) \\ &= -t_i^{\beta-1} Vec(\Sigma^{-1}) Vec'^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}') \\ &\quad -(\beta-1)t_i^{\beta-2}(\Sigma^{-1}\varepsilon_{(i)}\mathbf{x}_{(i)}'^{-1}\varepsilon_{(i)}\varepsilon_{(i)}'^{-1})\end{aligned}$$

Matrix Calculus for Type II MVPER Model

$$\frac{\partial E}{\partial B} = -\frac{\partial (XB)}{\partial B} = -Vec(X')Vec'(I_p).$$
$$\frac{\partial E'}{\partial B} = -\frac{\partial (B'X')}{\partial B} = -I_{(q,p)}(X' \otimes I_p).$$
$$\begin{aligned}(\Sigma^{-1} \otimes I_q)(E'^{-1} \otimes I_q)\tfrac{\partial E}{\partial B} &= -(\Sigma^{-1}E'^{-1} \otimes I_q)Vec(X')Vec'(I_p) \\ &= -Vec(X'^{-1}E\Sigma^{-1})Vec'(I_p).\end{aligned}$$

$$(\Sigma^{-1} \otimes I_q)\frac{\partial E'}{\partial B}(E \otimes I_p) = -(\Sigma^{-1} \otimes I_q)I_{(q,p)}(X'^{-1}E \otimes I_p)$$

$$\begin{aligned}\tfrac{\partial l(\theta)}{\partial B} &= -\tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * \tfrac{\partial (\Sigma^{-1}E'^{-1}E)}{\partial B}) \\ &= -\tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * (\Sigma^{-1} \otimes I_q)\tfrac{\partial (E'^{-1}E)}{\partial B}) \\ &= -\tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * (\Sigma^{-1} \otimes I_q) \\ &\quad (\tfrac{\partial E'}{\partial B}(\Phi^{-1}E \otimes I_p) + (E'^{-1} \otimes I_q)\tfrac{\partial E}{\partial B})) \\ &= \beta(tr(\Sigma^{-1}E'^{\beta-1}X'^{-1}E\Sigma^{-1}\end{aligned} \tag{8.23}$$

where $*$ is the *star product* Macrae ([24]) and $I_{(q,p)}$ is the *permuted identity* Macrae ([24]) or *commutation matrix* Magnus and Neudecker ([25]).

$$\begin{aligned}\tfrac{\partial l(\theta)}{\partial \Sigma} &= -\tfrac{n}{2}\Sigma^{-1} - \tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * \tfrac{\partial (\Sigma^{-1}E'^{-1}E)}{\partial \Sigma}) \\ &= -\tfrac{n}{2}\Sigma^{-1} - \tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * \tfrac{\partial \Sigma^{-1}}{\partial \Sigma}(E'^{-1}E \otimes I_p)) \\ &= -\tfrac{n}{2}\Sigma^{-1} \\ &\quad +\tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * Vec(\Sigma^{-1})Vec'^{-1})(E'^{-1}E \otimes I_p)) \\ &= -\tfrac{n}{2}\Sigma^{-1} + \tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}(I_p * Vec(\Sigma^{-1})Vec'^{-1}E'^{-1}E)) \\ &= -\tfrac{n}{2}\Sigma^{-1} + \tfrac{\beta}{2}(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}\Sigma^{-1}E'^{-1}E\Sigma^{-1}.\end{aligned} \tag{8.24}$$

$$\begin{aligned}\tfrac{\partial l^2(\theta)}{\partial B'\partial B'} &= \tfrac{\partial (\beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}\Sigma^{-1}E'^{-1}X)}{\partial B'} \\ &= \beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1}\tfrac{\partial (\Sigma^{-1}E'^{-1}X)}{\partial B'} \\ &\quad +(\Sigma^{-1}E'^{-1}X) \otimes \tfrac{\partial (\beta(tr(\Sigma^{-1}E'^{-1}E))^{\beta-1})}{\partial B'},\end{aligned} \tag{8.25}$$

$$\begin{aligned}\tfrac{\partial (\Sigma^{-1}E'^{-1}X)}{\partial B'} &= (\Sigma^{-1} \otimes I_p)(\tfrac{\partial E}{\partial B})'^{-1}X \otimes I_q) \\ &= -(\Sigma^{-1} \otimes I_p)Vec(I_p)Vec'(X'^{-1}X \otimes I_q) \\ &= -Vec(\Sigma^{-1})Vec'(X'^{-1}X).\end{aligned} \tag{8.26}$$

References

[1] Milton Abramowitz and Irene A. Stegun. *Handbook of mathematical functions, with formulas, graphs, and mathematical tables*. Dover Publications, New York,, 1965.

[2] H. Akaike. Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory (Tsahkadsor, 1971)*, pages 267–281. Akadémiai Kiadó, Budapest, 1973.

[3] H. Akaike. Likelihood of a model and information criteria. *Journal of Econometrics*, 16(1):3–14, 1981.

[4] Hirotugu Akaike. A new look at the statistical model identification. *IEEE Trans. Automatic Control*, AC-19:716–723, 1974. System identification and time-series analysis.

[5] Hirotugu Akaike. Factor analysis and AIC. *Psychometrika*, 52(3):317–332, 1987.

[6] George E. P. Box and George C. Tiao. *Bayesian inference in statistical analysis*. Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1973.

[7] H. Bozdogan. ICOMP: A new model-selection criterion. In *Classification and Related Methods of Data Analysis*, pages 599–608, 1988.

[8] H. Bozdogan. *Statistical data mining and knowledge discovery*. Chapman & Hall/CRC, Boca Raton, 2004.

[9] Hamparsum Bozdogan. On the information-based measure of covariance complexity and its application to the evaluation of multivariate linear models. *Communications in Statistics, Part A – Theory and Methods [Split from: @J(CommStat)]*, 19:221–278, 1990.

[10] Hamparsum Bozdogan. Choosing the number of component clusters in the mixture-model using a new informational complexity criterion of the inverse-Fisher information matrix. In *Information and Classification. Concepts, Methods and Applications. Proceedings of the 16th Annual Conference of the Gesellschaft fr Klassifikation e. V.*, pages 40–54, 1993.

[11] Hamparsum Bozdogan. Multivariate regression models for nonnormal data: a new model selection approach. In *52nd Session of the International Statistical Institute*, Helsinki, Finland, 1999.

[12] Hamparsum Bozdogan. Akaike's information criterion and recent developments in information complexity. *Journal of Mathematical Psychology*, 44(1):62–91, 2000.

[13] Hamparsum Bozdogan and Dominique M. A. Haughton. Informational complexity criteria for regression models. *Computational Statistics and Data Analysis*, 28:51–76, 1998.

[14] Hamparsum Bozdogan and Kazuo Shigemasu. Bayesian factor analysis model and choosing the number of factors using a new informational complexity criterion. In *Advances in Data Science and Classification. Proceedings of the 6th Conference of the International Federation of Classification Societies*, pages 335–342, 1998.

[15] S. Chatterjee and M. Laudatto. Genetic algorithms in statistics: Procedures and applications. *Communications in Statistics-Simulation and Computation*, 26(4):1617–1630, 1997.

[16] M. H. van Emden. *An analysis of complexity*. Mathematical Centre tracts. 35. Mathematisch Centrum, Amsterdam, 1971.

[17] Kai-Tang Fang and T. W. Anderson. *Statistical inference in elliptically contoured and related distributions*. Allerton Press, New York, 1990.

[18] Kai-Tang Fang, Samuel Kotz, and Kai W. Ng. *Symmetric multivariate and related distributions*. Monographs on statistics and applied probability. Chapman and Hall, New York, 1990.

[19] David E. Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley Pub. Co., Reading, Mass., 1989.

[20] E. Gomez, M. A. Gomez-Villegas, and J. M. Marin. A multivariate generalization of the power exponential family of distributions. *Communications in Statistics-Theory and Methods*, 27(3):589–600, 1998.

[21] A. K. Gupta and T. Varga. *Elliptically contoured models in statistics*. Kluwer Academic, Dordrecht ; Boston, 1993.

[22] J. H. Holland. Genetic algorithms. *Scientific American*, 267(1):66–72, 1992.

[23] Min-Hui Liu and H. Bozdogan. PE multiple regression model selection with ICOMP and genetic algorithms. *Working paper*, 2004.

[24] E. C. Macrae. Matrix derivatives with an application to an adaptive linear decision problem. *Annals of Statistics*, 2(2):337–346, 1974.

[25] J. R. Magnus and H. Neudecker. Commutation matrix - some properties and applications. *Annals of Statistics*, 7(2):381–394, 1979.

[26] K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate analysis*. Probability and mathematical statistics. Academic Press, London ; New York, 1979.

[27] M. Mitchell. *An Introduction to Genetic algorithms*. MIT Press, Cambridge, MA, 1996.

[28] Svetlozar T. Rachev and S. Mittnik. *Stable paretian models in finance*. Series in financial economics and quantitative analysis. Wiley, Chichester, 2000.

[29] J. Rissanen. Minimax entropy estimation of models for vector processes. In *System Identification: Advances in Case Studies*, pages 97–120, 1976.

[30] Gerald Stanley Rogers. *Matrix derivatives*. M. Dekker, New York, 1980.

[31] Eusebio Gmez Sanchez-Manzano, Miguel Angel Gomez-Villegas, and Juan-Miguel Marn-Diazaraque. A matrix variate generalization of the power exponential family of distributions. *Communications in Statistics, Part A – Theory and Methods [Split from: @J(CommStat)]*, 31(12):2167–2182, 2002.

[32] M. T. Subbotin. On the law of frequency of errors. *Matematicheskii Sbornik*, pages 296–300, 1923.

[33] P. Theodossiou. Financial data and the skewed generalized t distribution. *Management Science*, 44(12):1650–1661, 1998.

[34] J. Toyli, K. Kaski, and A. Kanto. On the shape of asset return distribution. *Communications in Statistics-Simulation and Computation*, 31(4):489–521, 2002.

[35] J. Yang and V. Honavar. Feature subset selection using a genetic algorithm. *Intelligent Systems and Their Applications, IEEE*, 13(2):44 – 49, 1998.

[36] R. Zeckhauser and M. Thompson. Linear regression with non-normal error terms. *Review of Economics and Statistics*, 52(3):280–286, 1970.