2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

Evolutionary Algorithms Applied to Likelihood Function Maximization During Poisson, Logistic, and Cox Proportional Hazards Regression Analysis

Leif E. Peterson

Abstract— **Metaheuristics based on genetic algorithms (GA), covariance matrix self-adaptation evolution strategies (CMSA-ES), particle swarm optimization (PSO), and ant colony optimization (ACO) were used for minimizing deviance for Poisson regression and maximizing the log-likelihood function for logistic regression and Cox proportional hazards regression. We observed that, in terms of regression coefficients, CMSA-ES and PSO metaheuristics were able to obtain solutions that were in better agreement with Newton-Raphson (NR) when compared with GA and ACO. The rate of convergence to the NR solution was also faster for CMSA-ES and PSO when compared with ACO and GA. Overall, CMSA-ES was the best-performing method used. Key factors which strongly influence performance are multicollinearity, shape of the log-likelihood gradient, and positive definiteness of the Hessian matrix.**

I. Introduction

The method of maximum likelihood using Newton-Raphson (NR) iteration for stochastic gradient ascent maximization has long been a successful technique used in regression modeling. A particular problem with NR, however, is that it is not a global optimization method, and its solution can get stuck at local optima of multimodal cost functions. Another limitation of NR is that the Hessian matrix **H** of second partial derivatives of the log-likelihood function w.r.t. the parameters must remain positive definite throughout the iterations. Since the covariance matrix is the inverse of **H** at convergence, the importance positive definiteness during matrix inversion increases with the number of parameters. More parameters mean a greater chance of multicollinearity, leading to a singular **H** as a result of zero or near-zero eigenvalues. For large problems, the likelihood function has to be summed over the entire solution space, requiring costly calculation of derivatives, which can overwhelm the desired run time.

Machine learning techniques are rapidly becoming popular in biomedical data analysis as the constraints of traditional techniques become more important. The majority of challenging problems in numerical analysis are high dimensional and involve combinatorial optimization such as timetabling, quadratic assignment, maximum satisfiability problems. Algorithms developed for solving combinatorial optimization problems are termed "exact" or "approximate." Exact algorithms can find an optimal solution within a given run-time and are therefore usually limited to solving small problems. For larger problems requiring much longer run-times, approximate methods are typically used for deriving suboptimal solutions via short run-times. A *heuristic* is a type of labor-intensive approximation algorithm developed for a particular cost function and search space, which can successfully solve a problem based on sound principles. The field of *metaheuristics* was introduced to address development of problem-independent high level search strategies for optimization problems [1]. Examples of early metaheuristics include genetic algorithms [2], [3], evolutionary programming [4], [5], evolution strategies [6]–[9], simulated annealing [10], tabu search [11], and iterated local search [12]. More recently, particle swarm optimization [13] and ant colony optimization [14], [15] were introduced as new swarm intelligence-based metaheuristics.

In light of the potential problems associated with NR likelihood maximization of costly multimodal problems and the rich mixture of metaheuristics now available, it is propitious to begin evaluating performance of metaheuristics when employed for likelihood maximization in regression methods for which only NR has been implemented. This paper addresses the use of genetic algorithms (GA), covariance matrix self-adaptation evolution strategies (CMSA-ES), particle swarm optimization (PSO), and ant colony optimization (ACO) for maximum likelihood optimization of non-linear models for Poisson regression, logistic regression, and Cox proportional hazards (PH) regression. These regression methods are commonly used in biomedicine, epidemiology, and public health to identify host (patient) and risk factors that are associated with or predictive of outcome. Poisson regression is typically used for modeling cancer rates, since many cancer rates are Poisson distributed where the counts are rare in terms of the number of years of follow-up among subjects. Logistic regression is a commonly used supervised classifier, for which the outcome variable is coded (0,1) to represent a case (with disease) or control (without disease). The binary dependent variable in logistic regression is regressed on the input variables, and no assumption is made about the distribution of the input predictors. Therefore, an advantage of logistic regression is that the input data can be categorical (nominal) or continuously-scaled without being normally distributed. Cox PH regression is a maximum likelihood method for modeling time-to-event failure data, otherwise known as survival analysis. The dependent variable for Cox PH is based on both the censoring (1-failed,0-censored) and the survival time (cumulative time until failure or censoring), while the input variables can be either categorical or continuously-scaled. Unlike linear least squares (multiple regression), Poisson regression, logistic regression, and Cox PH regression are termed "multiplicative models," since they involve optimization of a multiplicative likelihood function. Likelihood functions are essentially the product of

L.E. Peterson is with the Center for Biostatistics, Houston Methodist Research Institute, 6565 Fannin Street, Suite MGJ6-031, Houston, Texas 77030, USA. E-mail: lepeterson@houstonmethodist.org.

978-1-4799-1488-3/14/$31.00 ©2014 IEEE

many probabilities, where each probability is derived from a non-linear equation. Because maximization of a multiplicative likelihood function is mathematically intractable, the log of the likelihood function is maximized using partial derivatives of the log-likelihood function w.r.t. the regression coefficients.

This investigation addresses performance of Poisson regression, logistic regression, and Cox PH regression when GA, CMSA-ES, PSO, and ACO metaheuristics are used for solving parameters which maximize their log-likelihood. Within each of the regression models, comparisons are made between regression coefficients for each method, and model goodness-of-fit is provided as a function of learning iterations.

II. Methods

The NR method for maximizing the Poisson, logistic, and Cox PH regression log-likelihood functions is first addressed, followed by descriptions of the metaheuristic approaches involving GA, CMSA-ES, PSO, and ACO. NR is not used as a benchmark against which metaheuristic-based results are compared, but rather as the primary basis for solving the maximum likelihood problem due to its popularity. The rate of convergence of NR will greatly exceed the uphill gradient search performed by most of the slower learning metaheuristics, and since we lacked prior knowledge about metaheuristic performance with these likelihood functions, we generated plots of regression coefficient values as a function of iteration in order to observe the stability and relative change.

A. Poisson Regression

Let n represent the number of strata in a multiway table partitioned on categories for demographics, follow-up period, time since treatment, and risk factors, c_i the total number of failed cases in the ith table cell ($i = 1, 2, \ldots, n$), and PY_i the total person-years of follow-up in the ith cell. The likelihood function for n table cells is

$$\mathcal{L}(\lambda) = \prod_{i=1}^{n} \lambda^{c_i} e^{-\lambda PY_i}, \tag{1}$$

where λ is the hazard function. Now let $\mathbf{x}_i$ be the p-dimensional vector of covariate (feature) values for subjects in the ith table cell such that $\lambda = \exp(\mathbf{x}_i \boldsymbol{\beta})$. On substitution, the likelihood becomes

$$\mathcal{L}(\boldsymbol{\beta}) = \prod_{i=1}^{n} e^{c_i \mathbf{x}_i \boldsymbol{\beta}} \exp(-e^{\mathbf{x}_i \boldsymbol{\beta}} PY_i). \tag{2}$$

The log of the likelihood is

$$\log \mathcal{L}(\boldsymbol{\beta}) = \sum_{i=1}^{n} c_i \mathbf{x}_i \boldsymbol{\beta} - e^{\mathbf{x}_i \boldsymbol{\beta}} PY_i, \tag{3}$$

with score

$$s_j(\boldsymbol{\beta}) = \frac{\partial \log \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{n} x_{ij}(c_i - e^{\mathbf{x}_i \boldsymbol{\beta}} PY_i), \tag{4}$$

and element (j, k) of the Hessian

$$H_{j,k}(\boldsymbol{\beta}) = \frac{-\partial^2 \log \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j \partial \beta_k} = \sum_{i=1}^{n} x_{ij} x_{ik} e^{\mathbf{x}_i \boldsymbol{\beta}} PY_i. \tag{5}$$

Maximum likelihood estimates of each β_j can be found by determining the vector $\boldsymbol{\beta}$ iteratively with the matrix manipulation:

$$\boldsymbol{\beta}_{i+1} = \boldsymbol{\beta}_i + \mathbf{H}(\boldsymbol{\beta})^{-1} \mathbf{s}(\boldsymbol{\beta}), \tag{6}$$

until convergence is reached when $||\boldsymbol{\beta}|| < \epsilon$. Values of ϵ are typically in the range $10^{-8} \leq \epsilon \leq 10^{-4}$. At convergence, the predicted number of failures is $\hat{c}_i = \lambda PY_i$ for group i, and the deviance residuals are

$$r_i = c_i \log\left(\frac{c_i}{\hat{c}_i}\right) + (\hat{c}_i - c_i), \tag{7}$$

which allow the investigator to determine the goodness of fit of the model, i.e., how well the observed data agree with the fitted values.

B. Unconditional Logistic Regression

Let $y_i = 1$ represent disease and $y_i = 0$ non-disease for subject i having covariate vector $\mathbf{x}_i$ [16]. The probability of disease for subject i is

$$\pi_{i1} = P(y_i = 1|\mathbf{x}_i) = \frac{e^{g_1(\mathbf{x}_i)}}{e^{g_0(\mathbf{x}_i)} + e^{g_1(\mathbf{x}_i)}}, \tag{8}$$

and the probability of not having disease is

$$\pi_{i0} = 1 - \pi_{i1} = P(y_i = 0|\mathbf{x}_i) = \frac{e^{g_0(\mathbf{x}_i)}}{e^{g_0(\mathbf{x}_i)} + e^{g_1(\mathbf{x}_i)}}, \tag{9}$$

where $g_j(\mathbf{x}_i)$ is the *logit*, i.e., $\log(\pi_{ij}/\pi_{i0})$ for response category $j = 0, 1$. The likelihood is

$$\begin{aligned} \mathcal{L}(\boldsymbol{\beta}) &= \prod_{i=1}^{n} (\pi_{i0})^{1-y_i} \pi_{i1}^{y_i} \\ &= \prod_{i=1}^{n} (1 - \pi_{i1})^{1-y_i} \pi_{i1}^{y_i}. \end{aligned} \tag{10}$$

Taking the natural logarithm gives the log-likelihood in the form

$$\log(\mathcal{L}(\boldsymbol{\beta})) = \sum_{i=1}^{n} y_i(\mathbf{x}_i^T \boldsymbol{\beta}) - \log\left[1 + e^{(\mathbf{x}_i^T \boldsymbol{\beta})}\right]. \tag{11}$$

The score is

$$s_j(\boldsymbol{\beta}) = \frac{\partial \log \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{n} x_i \left(y_i - \frac{e^{(\mathbf{x}_i^T \boldsymbol{\beta})}}{1 + e^{(\mathbf{x}_i^T \boldsymbol{\beta})}}\right), \tag{12}$$

and element (j, k) of the Hessian matrix is

$$H_{j,k}(\boldsymbol{\beta}) = \frac{-\partial^2 \log \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j \partial \beta_k} = \sum_{i=1}^{n} x_{ij} x_{ik} \frac{e^{(\mathbf{x}_i^T \boldsymbol{\beta})}}{(1 + e^{(\mathbf{x}_i^T \boldsymbol{\beta})})^2}. \tag{13}$$

Maximum likelihood estimates of each β_j are found using the same matrix operations given in (6).

C. Proportional Hazards Regression

Let $t_{(1)} < t_{(2)} < \cdots < t_{(k)}$ be the failure times for k subjects with covariates $\mathbf{x}_{(1)}, \mathbf{x}_{(2)}, \ldots, \mathbf{x}_{(k)}$, and $R(t_{(i)})$ the risk set of individuals at risk prior to failure time $t_{(i)}$ [17]. For c_i subjects who fail at time $t_{(i)}$ inferences about the regression parameters $\boldsymbol{\beta}$ are estimated by maximizing the likelihood function

$$\mathcal{L}(\boldsymbol{\beta}) = \prod_{i=1}^{k} \frac{e^{(\mathbf{x}_i^T;\boldsymbol{\beta})}}{\left[\sum_{l \in R(t_{(i)})} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}\right]^{c_i}}. \tag{14}$$

The likelihood can be linearized by taking the log of the likelihood, which gives

$$\log \mathcal{L}(\boldsymbol{\beta}) = \sum_{i=1}^{k} \mathbf{x}_i^T; \boldsymbol{\beta} - c_i \log \left[\sum_{l \in R(t_{(i)})} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})} \right], \tag{15}$$

with score

$$s_j(\boldsymbol{\beta}) = \frac{\partial \log \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_j} = \sum_{i=1}^{k} \left[x_{ij} - c_i \frac{\sum_{l \in R(t_{(i)})} x_{jl} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}}{\sum_{l \in R(t_{(i)})} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}} \right], \tag{16}$$

and element (h, j) of the Hessian matrix

$$\begin{aligned} H_{h,j}(\boldsymbol{\beta}) &= \frac{-\partial^2 \log \mathcal{L}(\boldsymbol{\beta})}{\partial \beta_h \partial \beta_j} \\ &= \sum_{i=1}^{n} c_i \left[\frac{\sum_{l \in R(t_{(i)})} x_{hl} x_{jl} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}}{\sum_{l \in R(t_{(i)})} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}} - AB \right], \end{aligned} \tag{17}$$

where

$$A = \left(\frac{\sum_{l \in R(t_{(i)})} x_{hl} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}}{\sum_{l \in R(t_{(i)})} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}} \right), \tag{18}$$

and

$$B = \left(\frac{\sum_{l \in R(t_{(i)})} x_{jl} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}}{\sum_{l \in R(t_{(i)})} e^{(\mathbf{x}_l^T;\boldsymbol{\beta})}} \right). \tag{19}$$

Apply matrix operation given in (6) to solve for $\boldsymbol{\beta}$.

D. Genetic Algorithms (GA)

The above likelihood functions were maximized using p input features (variables) from the datasets used. Parametric tuning of unknowns was accomplished using GA genes with a precision of 10^{-6}, based on use of gene length $L = 20$. Binary bits $\{0, 1\}$ were initialized by use of a uniform random variate $U(0, 1)$ which was rounded down to 0 if $U(0, 1) \leq 0.5$ and rounded up to 1 if $U(0, 1) > 0.5$. Each chromosome consisted of multiple L-length binary strings representing the p coefficients. Binary to decimal encoding was performed, and decimal values of coefficients were then applied to all samples in order to calculate the log-likelihood (fitness) for Cox PH and logistic regression, and $1/\sum_i r_i$ for Poisson regression. A total of 200 generations was used for each GA run, with 20 chromosomes. Pairs of chromosomes were selected using tournament selection, and single-point crossover was performed at a randomly selected bit on each chromosome if a randomly drawn $U(0, 1)$ was below the crossover probability $P_c = 0.9$. Each bit of the child chromosomes underwent mutation if a random $U(0, 1)$ was less than $P_m = 0.05$. The process of binary to decimal conversion for obtaining parameters, determination of fitness for each chromosome, tournament selection, crossover, and point mutation, was repeated for the specified number of generations.

E. Covariance Matrix Self-Adaptation (CMSA-ES)

The unknowns determined with covariance matrix self-adaptation (CMSA-ES) were read as

$$\boldsymbol{\beta}^{(g+1)} = \langle \boldsymbol{\beta} \rangle_w^{(g)} + \sigma^{(g)} \mathbf{L}^{(g)} \mathbf{z}, \tag{20}$$

where $\langle \boldsymbol{\beta} \rangle_w^{(g)}$ are the parameter means after generation g based on the μ most fit chromosomes, $\sigma^{(g)}$ is a step size that controls the variance of $\mathbf{z}$, which is a vector of randomly drawn standard normal variates, and $\mathbf{L}^{(g)}$ is the Cholesky factorization matrix of the covariance matrix $\mathbf{C}^{(g)}$. Matrix updates and training parameters used in CMSA-ES are described in detail in [18]. A total of 200 generations were employed for each run using CMSA-ES. The number of chromosomes was set to 8 when $p \leq 8$ and $4 + \lfloor 3 \log(p) \rfloor$ when $p > 8$, using guidelines found in [18].

F. Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) was also employed for solving the unknown regression parameters [13]. At each iteration, the fitness was based on the log-likelihood using the current parameters. The velocity update for the lth particle is

$$\begin{aligned} v_l(t+1) = w(t) v_l(t) &+ c_1 U(0,1) \otimes (\mathbf{b}_l(t) - \mathbf{r}_l(t)) + c_2 U(0,1) \\ &\otimes (\mathbf{b}_g(t) - \mathbf{r}_l(t)), \end{aligned} \tag{21}$$

where where $w(t)$ is the *inertia factor*, c_1 is the cognitive parameter, c_2 is the social parameter, $\mathbf{r}_l$ and $\mathbf{v}_l$ are the position and velocity vectors for particle l, $\mathbf{b}_l(t)$ is the best historical fitness for particle l, $\otimes$ is the direct product, and $\mathbf{b}_g(t)$ is the global best particle. The inertia at iteration t is $w(t) = w_{start} - (w_{start} - w_{end})t/T_{max}$. The particle position update at each iteration is $\mathbf{r}_l(t+1) = \mathbf{r}_l(t) + \mathbf{v}_l(t+1)$. After the required number training generations, regression coefficients $\boldsymbol{\beta}$ were set equal to $\mathbf{r}_l$ for the particle with the greatest fitness. Other parameters in PSO were set to: #numparticles = 20, T_{max}= 300, $v_{min} = -0.05$, $v_{max} = 0.05$, $c_1 = 2$, $c_2 = 2$, $w_{min} = 0.4$, and $w_{max} = 0.9$. A total of 200 generations were used for fitness calculations.

G. Ant Colony Optimization (ACO)

The ant colony optimization (ACO) method employed kernel density estimation (KDE) to improve solutions that yielded the best fitness among a population of solution vectors [14], [15]. Let $\mathbf{S} \in \mathbb{R}^{p \times 100}$ be the solution archive for 100 solution vectors having p parameters ($j = 1, 2, \ldots, p$). Let f_l be the fitness for the lth solution vector applied to the input samples. Solution vectors were ranked $R_{(1)}, R_{(2)}, \ldots, R_{(l)}, \ldots, R_{(100)}$, and the weight at generation t for each solution vector in $\mathbf{S}$ was defined as

$$w_l(t) = \frac{1}{qp\sqrt{(2\pi)}} \exp\left\{ \frac{(R_{(l)} - 1)^2}{2q^2 p^2} \right\}, \tag{22}$$

where q is a fixed parameter for the algorithm. The weight w defined a Gaussian variate with mean 1 and $\sigma = qp$. Small values of q provided greater weight to the best fitting solution vectors, while larger values distributed the weight more uniformly. We used a value of $q = 0.1$ for all runs, which was observed to provide the best fits by solution vectors. The probability of choosing a specific solution vector l was

$$P_l(t) = \frac{w_l}{\sum_l^{100} w_l}. \tag{23}$$

For each successive learning generation, KDE was used to simulate a normally distributed quantile for each parameter, with the mean equal to the current parameter value, that is, $\mu_l = s_{jl}$, and the standard deviation determined as

$$\sigma_l(t) = \xi \sum_e^{100} \frac{d(\mathbf{s}_e, \mathbf{s}_l)}{p-1}, \tag{24}$$

where $\mathbf{s}_e$ are each of the 100 solution vectors. The term ξ is similar to the pheromone evaporation rate employed in discrete ACO methods, and controls the learning rate. We used $\xi = 0.3$, which yielded the best results after performing a grid search with increments of $\Delta\xi = 0.05$ in the range $0 < \xi \leq 1$.

A total of 200 generations were used for fitness calculations. For each generation, values of s_{jl} were obtained by using the rejection method for the pdf derived from KDE with $M = 100$ equally spaced bins over the range $(s_{jl} - 4, s_{jl} + 4)$ [19]. Prior to training, $\mathbf{S}$ is initialized with standard normal variates distributed as $\mathcal{N}(0,1)$. During each generation, the fitness of each solution vector, f_l, is determined by applying each solution vector to the input samples. Fitness is then sorted in descending order. Next, selection weights w_l, probabilities P_l, and standard deviations σ_l for KDE are determined for each solution vector. For each generation, two new solution vectors were simulated. The first new solution vector was simulated by using KDE p times, based on the single value of σ_l and p values of $\mu = s_{jl}$ for the lth existing solution vector for which fitness rank was $R_{(l)} = 1$. This was repeated for the second new solution vector for using values from the existing solution vector for which rank was $R_{(l)} = 2$. If the fitness values of either (both) of the new solution vectors was greater than the worst fitness values, then the solution vectors with the worst fitness were replaced with these new solution vectors. The process of simulating 2 new solution vectors per generation represents 2 ants, which travel through the solution space. Replacement of the solution vectors whose fitness is the worst is similar to pheromone update of a potential pathway through the solution space. The solution vector with the greatest fitness among the 100 solutions stored in $\mathbf{S}$ is used for testing during function approximation.

H. Datasets for Regression Modeling

For Poisson regression, data tabulations from the British doctors mortality study of smoking were used [20]. Table I lists the number of deaths and person-years of follow-up. For logistic regression, the low birth weight dataset [21] includes 189 samples and 7 features representing 59 women who had low birth weight babies (<2500 gm) and 130 who had normal-weight babies. The features are: age of the mother in years (age), weight in pounds at the last menstrual period (lwt), smoking status during pregnancy (smoke, 1-yes,0-no), history of premature labor (ptl, 1-yes,0-no), history of hypertension (ht, 1=yes,0=no), presence of uterine irritability (ui, 1=yes,0=no), and number of physician visits during the first trimester (ftv). Last, for proportional hazards regression, we use the Veterans Lung cancer data [17]. This dataset has 137 records with 8 covariates: standard or test treatment type (trtmnt, 1=test,0=std), cell type2 (1=yes,0=no), cell type 3 (1=yes,0=no), cell type 4 (1=yes,0=no), Karnofsky score (0-100), time since diagnosis (months), age at diagnosis (y), prior therapy (1-yes,0-no), survival or follow-up time (d), and dead or censored.

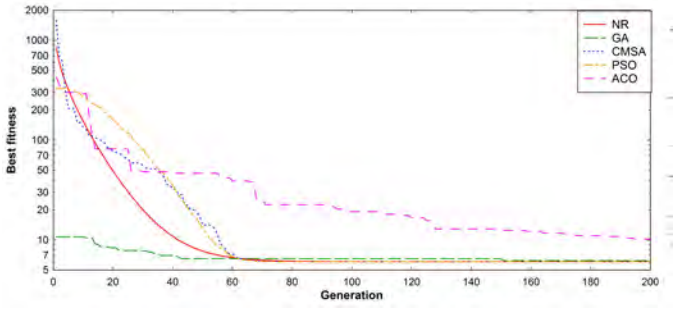

Fig. 1. Deviance for Poisson regression of mortality data as a function of iteration.

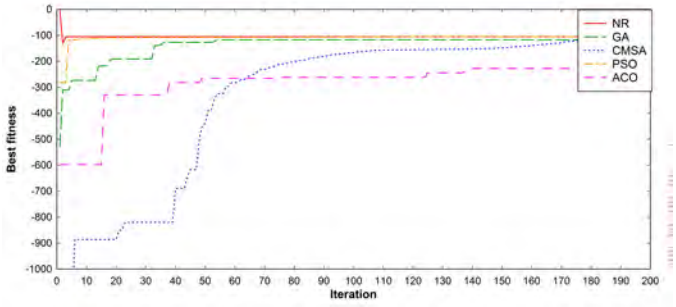

Fig. 2. Log-likelihood for logistic regression of low birth weight data as a function of iteration.

TABLE I

DEATHS AMONG SMOKING AND NON-SMOKING BRITISH MALE DOCTORS [20]. PARAMETER DEFINITIONS ARE: AGE GROUP i, NUMBER OF DEATHS d_i, PERSON-YEARS OF FOLLOW-UP T_i, DEATH RATE AMONG NON-EXPOSED $\lambda_i(0)$, DEATH RATE AMONG EXPOSED $\lambda_i(1)$.

	Non-smokers			Smokers		
Age group, i	d_i	T_i	$\lambda_i(0)$	d_i	T_i	$\lambda_i(1)$
35-44	2	18,790	0.1064	32	52,407	0.6106
45-54	12	10,673	1.1243	104	43,248	2.4047
55-64	28	5,710	4.9037	206	28,612	7.1998
65-74	28	2,585	10.8317	186	12,663	14.6885
75-84	31	1,462	21.2038	102	5,317	19.1838

TABLE II

POISSON REGRESSION COEFFICIENTS FOR THE BRITISH DOCTORS MORTALITY DATA AFTER 200 ITERATIONS. FEATURES IN COLUMNS ARE: AGE IN RANGE 35-44(AGE35-44, 1-YES,0-NO), AGE IN RANGE 45-54(AGE45-54,1-YES,0-NO), AGE IN RANGE 55-64(AGE55-64,1-YES,0-NO), AGE IN RANGE 65-74(AGE65-74,1-YES,0-NO), AGE IN RANGE 75-84(AGE75-84,1-YES,0-NO), SMOKE(1-YES,0-NO), FOLLOWED BY DEVIANCE GOODNESS-OF-FIT.

	Age35-44	Age45-54	Age55-64	Age65-74	Age75-84	Smoke	Deviance
NR	-1.0115	0.4724	1.6159	2.3389	2.6885	0.3545	6.07
GA	-0.0366	0.4605	1.6426	2.5180	3.0414	0.3246	6.22
CMSA-ES	-1.0116	0.4724	1.6159	2.3389	2.6885	0.3545	6.07
PSO	-1.0116	0.4724	1.6159	2.3389	2.6885	0.3546	6.07
ACO	-1.2667	0.1667	1.3265	2.0250	2.4418	0.6882	10.24

TABLE III

LOGISTIC REGRESSION COEFFICIENTS FOR THE LOW BIRTH WEIGHT DATA AFTER 200 ITERATIONS. FEATURES IN COLUMNS ARE: AGE (YEARS), WEIGHT IN POUNDS AT THE LAST MENSTRUAL PERIOD (LWT), SMOKING STATUS DURING PREGNANCY (SMOKE, 1-YES,0-NO), HISTORY OF PREMATURE LABOR (PTL, 1-YES,0-NO), HISTORY OF HYPERTENSION (HT, 1=YES,0=NO), PRESENCE OF UTERINE IRRITABILITY (UI, 1=YES,0=NO), NUMBER OF PHYSICIAN VISITS DURING THE FIRST TRIMESTER (FTV), FOLLOWED BY MODEL LOG-LIKELIHOOD.

	age	lwt	smoke	ptl	ht	ui	ftv	Log(L)
NR	-0.0151	-0.0089	0.6234	0.5950	1.7307	0.8379	0.0075	-105.21
GA	0.1236	-0.0338	0.7734	2.0685	2.5423	0.7251	-0.7031	-117.30
CMSA-ES	-0.0434	-0.0031	0.4787	1.7106	-0.0396	1.0061	-0.1525	-114.12
PSO	-0.0152	-0.0089	0.6235	0.5951	1.7290	0.8379	0.0073	-105.21
ACO	-0.2362	-0.0428	0.9146	0.0028	7.5577	-7.2877	-0.8640	-181.98

TABLE IV

COX PROPORTIONAL HAZARDS REGRESSION COEFFICIENTS FOR THE VETERANS LUNG CANCER DATA AFTER 200 ITERATIONS. FEATURES IN COLUMNS ARE: STANDARD OR TEST TREATMENT TYPE (TRTMNT, 1=TEST,0=STD), CELL TYPE2 (1=YES,0=NO), CELL TYPE 3 (1=YES,0=NO), CELL TYPE 4 (1=YES,0=NO), KARNOFSKY SCORE (0-100), TIME SINCE DIAGNOSIS (MONTHS), AGE AT DIAGNOSIS (Y), PRIOR THERAPY (1-YES,0-NO), FOLLOWED BY MODEL LOG-LIKELIHOOD.

	trtmnt	celltype2	celltype3	celltypr4	Karnof	timesd(m)	ageatDx	priorTX	Log(L)
NR	0.1491	0.4115	0.4762	0.1595	-0.6566	-0.0016	-0.0915	0.0327	-474.38
GA	0.0436	0.2311	0.3163	0.0009	-0.7865	-0.0588	-0.1943	-0.0556	-474.72
CMSA-ES	0.1491	0.4115	0.4762	0.1595	-0.6566	-0.0016	-0.0915	0.0327	-474.38
PSO	0.1491	0.4115	0.4762	0.1595	-0.6566	-0.0016	-0.0915	0.0327	-474.38
ACO	0.5092	0.4780	0.4884	0.0944	-0.6001	-0.0833	-0.0055	0.1281	-482.22

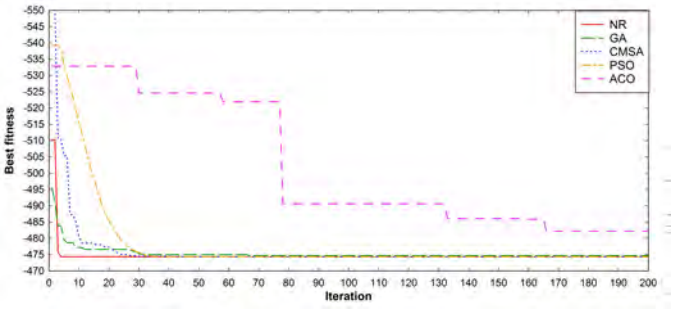

Fig. 3. Log-likelihood for Cox proportional hazards regression of lung cancer data as a function of iteration.

III. RESULTS

Poisson regression coefficients for the mortality data obtained after 200 iterations are listed in Table II. When compared with NR, only CMSA-ES and PSO provided coefficients and deviance values that were similar. Logistic regression coefficients for the low birth weight data obtained after 200 iterations are listed in Table III. PSO was the only method that resulted in coefficients and a log-likelihood value that were similar to results from NR. Results for Cox proportional hazards regression of the Veteran's lung cancer data are listed in Table IV. Here, both CMSA-ES and PSO yielded coefficients and log-likelihood values that were similar to results derived using the NR method for maximizing the log-likelihood functions. Figure 1 shows Poisson regression convergence rates for the most fit chromosome (particle) for various methods based on deviance values at each iteration. Not surprisingly, a smooth convergence occurred with NR. The ACO, PSO, and CMSA-ES methods converged slower than NR, and the GA resulted in low deviance early on but with very heterogeneous coefficient values. The ACO method resulted in the slowest convergence rate, since only 2 chromosomes are used during learning. Figure 2 shows logistic regression convergence rates of the various approaches for the fittest chromosome based on log-likelihood values at each iteration. A different pattern of convergence emerged for the logistic regression analysis, whereby PSO, GA, ACO had faster convergence rates, followed by CMSA. Figure 3 shows the Cox PH regression convergence rates for various methods for the fittest chromosome based on log-likelihood. Here, the convergence rates for fastest for GA, CMSA, PSO, followed by the slower ACO approach.

Figure 4 shows the Poisson regression coefficients for the mortality data as a function of generation for minimizing model deviance with NR, GA, CMSA, PSO, and ACO. By

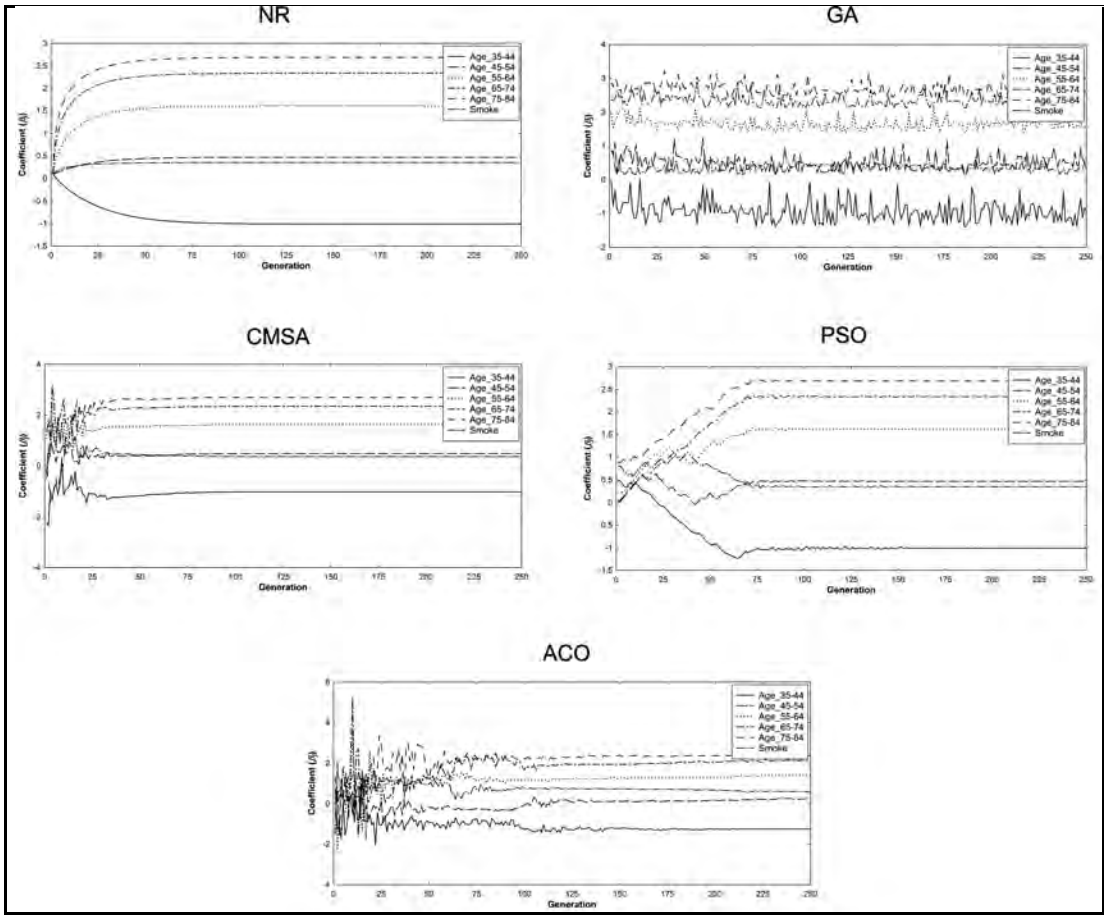

Fig. 4. Poisson regression coefficients for mortality data as a function of generation (iteration) for minimizing model deviance with Newton-Raphson (NR), genetic algorithm (GA), covariance matrix self-adaptation (CMSA), particle swarm optimization (PSO), and ant colony optimization (ACO).

50 generations, CMSA-ES arrived at the same solution as NR at nearly twice the rate. Regression coefficients for the GA method were too heterogeneous to be of value for such a model, and this reflects how stagnant a GA's learning rate can become before arriving at a solution. Certainly, many more generations would have solved the coefficients, but at the expense of longer computing time. PSO showed a smooth transition in coefficient updating and arrived at a solution at approximately 100 generations. The ACO-based method arrived at a solution near 125 generations, and early on revealed a pattern of coefficient updating that was similar to CMSA-ES – but with more jumpy transitions. Figure 5 shows the logistic regression coefficients for the low birth weight data as a function of generation for maximizing log-likelihood with NR, GA, CMSA, PSO, and ACO. For the logistic regression model, GA and ACO provided unreliable coefficient values at 250 generations. CMSA-ES went through a rapid transition period between 50 and 150 generations involving fewer stable updates to coefficients, and finally arrived at a solution. PSO performed the same as it did for Poisson regression, undergoing smooth updating of coefficients until arriving at a solution near 175 generations. Figure 6 shows the Cox PH regression coefficients for the lung cancer data as a function of generation for maximizing the log-likelihood with NR, GA, CMSA, PSO, and ACO. GA resulted in slightly more stable coefficients for the Cox PH regression model, and while CMSA-ES quickly arrived at a solution near 40 generations, PSO arrived at a solution near 75 generations. ACO coefficient values were considerably heterogeneous at 250 generations, due to the slower learning rate imposed by use of just two chromosomes.

IV. Discussion

We observed that, in terms of regression coefficients, the CMSA-ES and PSO metaheuristics were able to provide results which were most similar to results from NR. Results from GA and ACO were usually less in agreement with NR. Looking at the iteration-specific results, the majority of methods rapidly ascended the log-likelihood gradient, only differing by the rate of convergence. Generally speaking, regression by CMSA-ES and PSO rapidly converged, while ACO tended to be slower because of the two chromosomes used in training. The maximum log-likelihood values during the early iterations of regression varied considerably, which was mostly due to differences in parameter initialization. NR parameter initialization used small (0.1) values for all parameters, which may cause the starting log-likelihood fit location to be very far away from the maximum, whereas the metaheuristic methods use randomly generated standard normal variates.

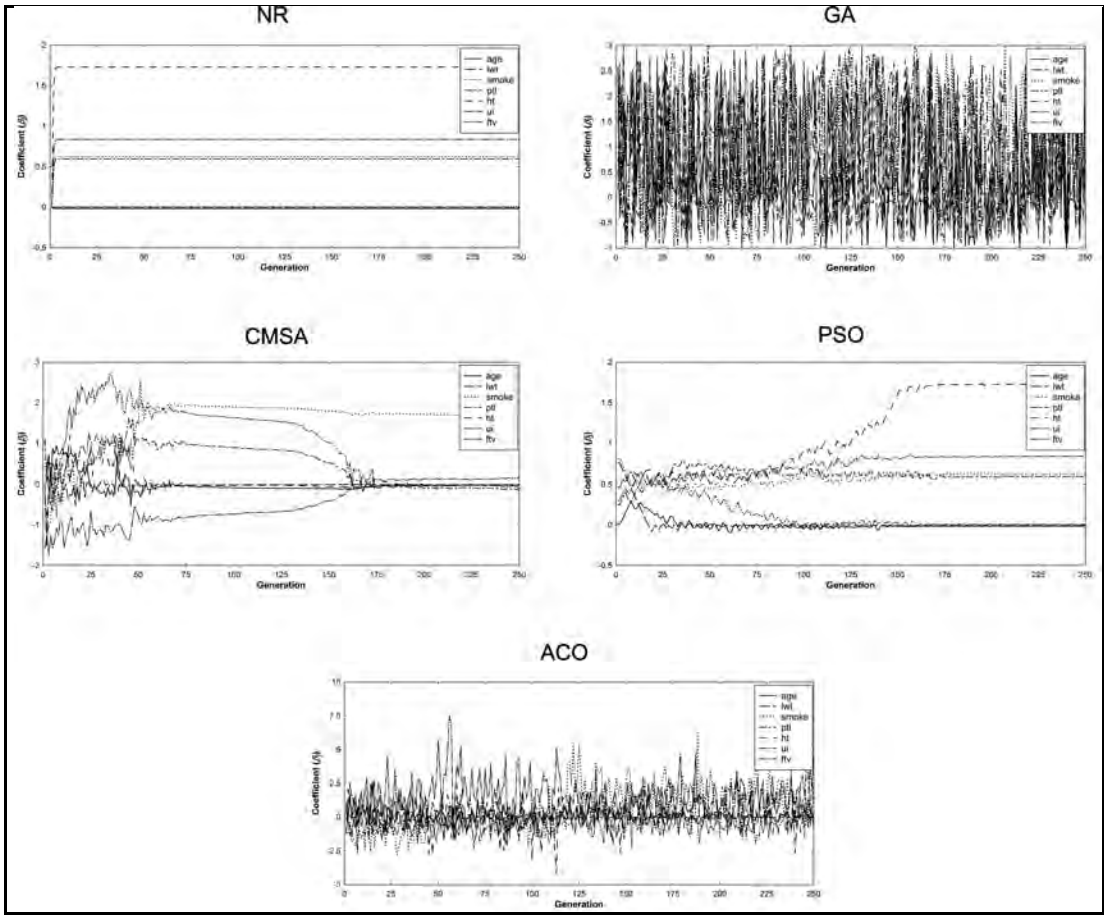

Fig. 5. Logistic regression coefficients for low birth weight data as a function of generation (iteration) for maximizing log-likelihood with Newton-Raphson (NR), genetic algorithm (GA), covariance matrix self-adaptation (CMSA), particle swarm optimization (PSO), and ant colony optimization (ACO).

When compared with GA and ACO, CMSA-ES and PSO took larger steps during learning. CMSA-ES relies on mutations of the step direction to form the covariance matrix – which acts as a surrogate for the Hessian matrix. ACO was observed to yield lower convergence rates, and this is mostly due to the use of only two chromosomes for updating among the 100 chromosomes used for bookkeeping in the solution library. Hence, ACO updates much slower than PSO and CMSA-ES. Initially, a GA crossover rate of $P_c = 0.6$ and mutation rate of $P_m = 0.005$ resulted in slow convergence rates, which were increased when using $P_c = 0.9$ and $P_m = 0.05$. Nevertheless, GA resulted in heterogeneous coefficient values that were jumpy and unreliable. Overall, CMSA-ES was observed to converge the fastest among the metaheuristics used, and also resulted in coefficients that were similar to NR results. PSO arrived at a solution a little more slowly when compared with CMSA-ES, but its coefficient updates were quite smooth. We did not assess the effect of collinearity on each of the methods, and realize that this could have a strong impact on the ability of various metaheuristic approaches for function minimization(maximization). We also did not use the square root of the Hessian matrix during NR iterations, nor did we monitor the condition number of the Hessian matrix to assess singularity and positive-definiteness. The datasets employed were also not overly complex in terms of multimodal fitness landscapes, so our use of NR as a comparison basis would not appreciably bias the observed differences between the metaheuristic and NR results. Multicollinearity (vs. orthogonality) would also elicit a strong influence on the convergence rates of the various methods used, so our future studies will incorporate simulations and evaluations of multicollinearity of the variance-covariance matrix, and positive definiteness of the Hessian matrix, while employing more complex datasets.

V. Conclusions

The major observation of this investigation was that, with respect to the regression coefficients, CMSA-ES and PSO metaheuristics arrived at the NR solution much better than GA and ACO. The CMSA-ES and PSO methods also arrived at the NR convergence much faster when compared with ACO and GA. The drawback of the GA was that it stagnated during many iterations without showing rapid improvement in its learning rate. The ACO was limited by only having two chromosomes, causing lower learning rates through the training iterations, and this was a design limitation of the particular variant of ACO that was employed. Several factors that strongly affected performance of these metaheuristics were likely to be multicollinearity among input features, shape

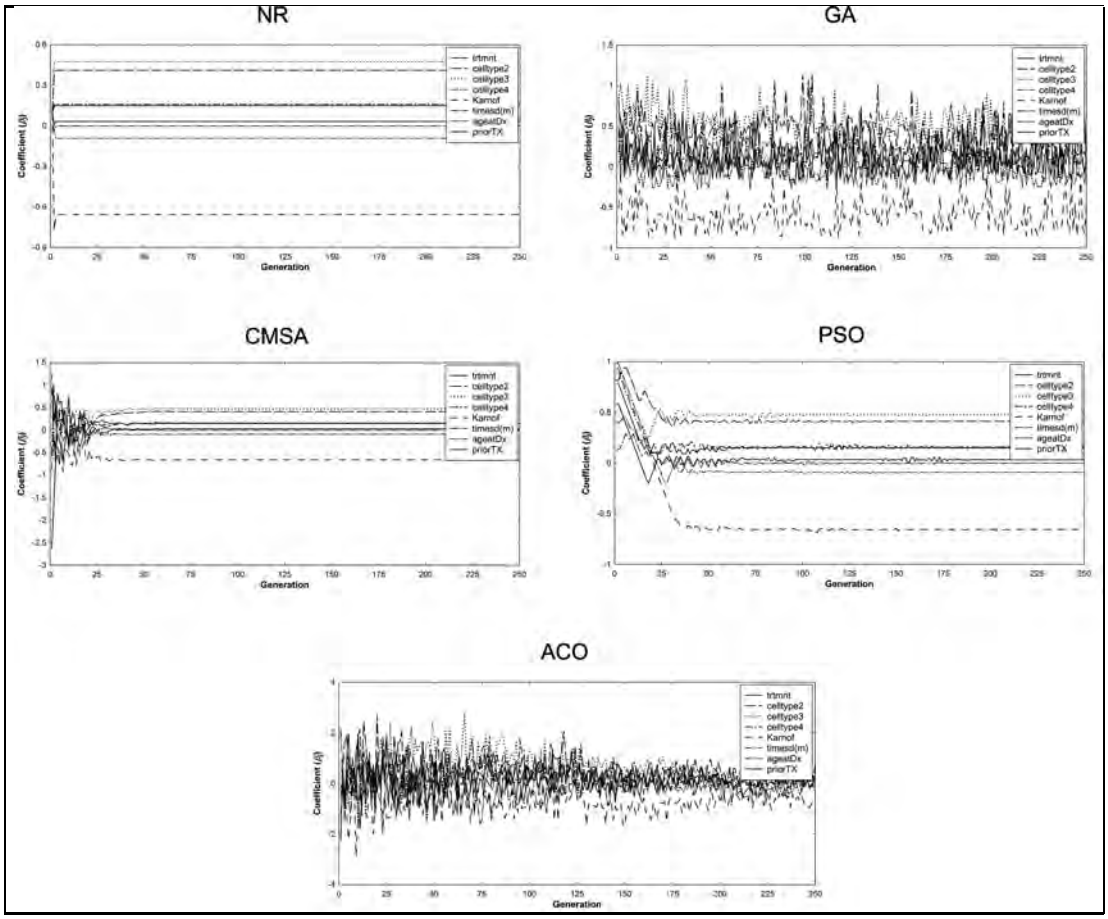

Fig. 6. Cox proportional hazards regression coefficients for lung cancer data as a function of generation (iteration) for maximizing log-likelihood with Newton-Raphson (NR), genetic algorithm (GA), covariance matrix self-adaptation (CMSA), particle swarm optimization (PSO), and ant colony optimization (ACO).

of the log-likelihood gradient, and positive definiteness of the Hessian matrix.

ACKNOWLEDGMENT

We would like to acknowledge helpful suggestions from H-G Beyer over modeling and parameter choices for CMSA-ES.

REFERENCES

[1] Blum C, Roli A: Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys* 2003, **35(3)**:268–308.

[2] Holland JH: Outline for a logical theory of adaptive systems. *JACM* 1962, **9**:297–314.

[3] Holland JH: Adaptation in Natural and Artificial Systems. Ann Arbor (MI), Univ. of Michigan Press, 1975.

[4] Fogel L: Autonomous automata. *Industrial Research* 1962, **4**:14–19.

[5] Fogel L, Owens A, Walsh M: Artificial Intelligence Through Simulated Evolution. New York(NY), John Wiley and Sons, 1966.

[6] Rechenberg I: Cybernetic solution of path of an experimental problem. Tech. rep., Royal Aircraft Establishment, Farnborough, Vol. 1122 1965.

[7] Rechenberg I: Evolution Strategy: Optimization of Technical Systems According to Principles of Biological Evolution. Stuttgart, Frommann-Holzboog Verlag, 1973.

[8] Schwefel HP: Kybernetische Evolution als Strategie der experimentellen Forschung in der Strömungstechni. *Master's thesis*, Technical University of Berlin, 1965.

[9] Schwefel HP: Evolutionsstrategie und numerische Optimierung. *PhD thesis*, Technical University of Berlin, 1975.

[10] Kirkpatrick S, Gelatt CJ, Vecchi M: Optimization by simulated annealing. *Science* 1983, **220**:671–680.

[11] Glover F: Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research* 1986, **13(5)**:533–549.

[12] Lourenço H, Martin O, Stutzle T: Handbook of Metaheuritics, Chap: *Iterated Local Search*. Norwell (MA), Kluwer, 2002.

[13] Kennedy J, Eberhart R: Particle swarm optimization. In *Proc. IEEE International Conference on Neural Networks*, Piscataway(NJ), IEEE Press 1995:1942–1948.

[14] Dorigo M: Optimization, Learning and Natural Algorithms (in Italian). *PhD thesis*, Dipartimento di Elettronica, Politecnico di Milano 1992.

[15] Socha K, Dorigo M: Ant colony optimization for continuous domains. *European Journal of Operational Research* 2008, **185**:1155–1173.

[16] Cox D: *Analysis of Binary Data*. New York (NY), Chapman and Hall, 1970.

[17] Kalbfleisch J, Prentice R: *The Statistical Analysis of Failure Time Data*. New York (NY), John Wiley and Sons, 1980.

[18] Beyer HG, Sendhoff B: Covariance matrix adaptation revisted: The CMSA evolution strategy. *LNCS* 2008, **5199**:123–132.

[19] Fadda A, Slezak E, Bijaoui A: Density methods with non-parametric methods. *Astron. Astrophys. Suppl. Ser.* 1998, **127**:335–352.

[20] Doll R, Hill A: Mortality of British doctors in relation to smoking: Observations on coronary thrombosis, *Volume 19*. Natl. Cancer Inst. Monograph, 1966.

[21] Hosmer D, Lemeshow S: *Applied Logistic Regression*. New York (NY), John Wiley and Sons, 1989.