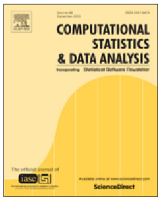
Computational Statistics and Data Analysis 129 (2019) 119–134

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda



Greedy active learning algorithm for logistic regression models

Hsiang-Ling Hsu [a], Yuan-chin Ivan Chang [b], Ray-Bing Chen [c,*]

[a] *National University of Kaohsiung, Taiwan*
[b] *Academia Sinica, Taiwan*
[c] *National Cheng Kung University, Taiwan*

ARTICLE INFO

Article history:
Received 22 December 2017
Received in revised form 13 August 2018
Accepted 14 August 2018
Available online 27 August 2018

Keywords:
Active learning algorithm
D-efficiency criterion
Forward selection
Graft optimization

ABSTRACT

We study a logistic model-based active learning procedure for binary classification problems, in which we adopt a batch subject selection strategy with a modified sequential experimental design method. Moreover, accompanying the proposed subject selection scheme, we simultaneously conduct a greedy variable selection procedure such that we can update the classification model with all labeled training subjects. The proposed algorithm repeatedly performs both subject and variable selection steps until a prefixed stopping criterion is reached. Our numerical results show that the proposed procedure has competitive performance, with smaller training size and a more compact model compared with that of the classifier trained with all variables and a full data set. We also apply the proposed procedure to a well-known wave data set (Breiman et al., 1984) and a MAGIC gamma telescope data set to confirm the performance of our method.

1. Introduction

To train a classification model, labeled data are essential when a training/testing framework is adopted, and its classification performance relies on both the size and the quality of the training subjects used for learning. In a Big Data scenario, we might easily meet a huge data set; however, the labeled information may be limited, and an abundance of unlabeled subjects are available. To prevent money laundering, Deng et al. (2009) studied the method for building a detection model using bank account data. This is a good example because in this situation, the label of interest (money laundering account) is limited in a regular bank account data set. It would require a huge amount of time and resources to verify whether an account is suspicious or non-suspicious, even though the major parts of the transactions in a bank account should be normal. Efficiently determining the potential risks within a bank account in addition to effectively and efficiently using the unlabeled subjects to improve the classification rule is the key issue, and the concept of active learning can be applied to this situation.

When we train a classifier in an active learning manner, we need to annotate the unlabeled data and recruit them into the training set, which can be done with the information of a model built on the labeled data at the current stage. In the literature, it is observed that people can usually learn a satisfactory model economically with such a procedure (Cohn et al., 1994a; Settles, 2011, 2012). There are many classification performance indexes, and it is clear that this subject selection process may depend on the targeted index (Settles, 2009; Hsu, 2010; Settles, 2011). For example, Culver et al. (2006) studied active learning procedures that maximize the area under the ROC curve (AUC), Long et al. (2010) were interested in the

* Corresponding author.
E-mail addresses: hsuhl@nuk.edu.tw (H.-L. Hsu), ycchang@stat.sinica.edu.tw (Y.I. Chang), rbchen@mail.ncku.edu.tw (R.-B. Chen).

https://doi.org/10.1016/j.csda.2018.08.013

ranking of the data, and Deng et al. (2009) used an active learning study focusing on efficiently selecting the most informative subjects to join the training set to construct an accurate classification model via the experimental design methods. However, when there are many redundant variables (predictors), to have an effective design is difficult and in such a situation, this classification model tends to over-fit the training subjects, which usually leads to high prediction uncertainty. A common approach to improving its prediction stability is to increase the size of the training set at the cost of protracting the training stage. Thus, a procedure that can identify a compact classification model during its training course is preferred.

In this paper, we propose a logistic model-based active learning procedure with a batch sampling to address binary classification problems under big data scenarios. In addition to the subject selection scheme, we also integrate a variable selection step into this procedure for systematically improving the prediction ability and avoiding the over-fitting of our final classification model. Hence, the proposed algorithm is an iterative algorithm, in which we select a batch of new samples at each iteration and then update our binary classification model via a variable selection approach. Both subject selection and variable selection are featured in this novel active learning procedure.

We organize the rest of this paper as follows. Section 2 presents the details of the subject selection and variable selection steps, and then we describe our active learning algorithm with an integrated subject and variable selection steps. Section 3 presents numerical results, where in addition to the simulation studies, we apply our algorithm to a well-known wave data set used in Breiman et al. (1984) and a MAGIC gamma telescope data set. We present a brief discussion and conclusion in Section 4.

2. Methodology

We consider a pool-based active learning procedure as studied in Lewis and Gale (1994) and assume that to obtain those unlabeled data is cheap and to query their label information is expensive. Hence, we should rationally select the unlabeled subjects from this large pool for being labeled to reduce the overall cost of model learning. We state the general framework of the pool-based active learning methods as follows.

```
Initialization: An initial labeled training set and a pool of unlabeled data
repeat
    Learn the current model based on the current labeled training data.
    Select points from unlabeled set via a query strategy framework based on the current learned model.
    Query labels for these selected points and update the training set.
until The stopping criterion is satisfied;
```

It is known that when there is a lengthy vector of variables, to train a classifier with an entire set of variables may diminish its prediction power and protract its training time; a compact classification model is usually preferred when the model is sparse. Hence, in our algorithm, we also emphasize the variable selection strategy in its learning process in addition to the common subject selection as in active learning procedures reported in the literature.

The variable selection frame has two common approaches: forward selection and backward elimination (Whitney, 1971). Because active learning procedures will usually start from a small size of training samples and keep accumulating according to a pre-specified selection rule, we can only obtain satisfactory estimation results for a small number of parameters. Hence, in our study, we use the forward selection scheme that increases the size of the variable set by adding a new variable to the current model at a time and adopt a greedy selection approach. Based on the characters discussed above, we propose a modified active learning algorithm, which integrates both batch-subject selection and greedy variable selection features together, and we refer to this Greedy AcTivE learning algorithm as GATE learning (or GATE) algorithm throughout this paper. Basically, we implement a variable selection step once we have an updated training set, and in each iteration of GATE, we add more labeled samples, and re-justify our classification model.

2.1. Logistic model for binary classification

Let Ξ_S denote the index set of the whole sample points and Ξ_s be the current training index set with labeled data. Thus, $\Xi_s^c = \Xi_S \setminus \Xi_s$ is the pool of the unlabeled data. In this section, we focus on how to identify the batch unlabeled subjects from Ξ_s^c for binary classification based on a logistic model and, meanwhile, propose a two-stage query procedure by putting the uncertainty sampling with the optimal design criterion together. Afterward, we introduce a greedy forward selection to update the current model by selecting a candidate variable from $\Xi_v^c = \Xi_V \setminus \Xi_v$, where Ξ_V is the index set of the whole variables and Ξ_v denotes the index set of the current active variables in the logistic model.

Assume that the ith individual variate $Y_i \in \{0, 1\}$ is a binary variable with

$$P(Y_i = 1) = p_i = E(Y_i) \text{ and } P(Y_i = 0) = 1 - p_i,$$

and let the feature values of the ith subject be $\mathbf{x}_{i,p} = (x_{i,j})^\top,\ i \in \Xi_s,\ j \in \Xi_v$ whose dimension is equal to p. Then, we can fit this data set with a logistic regression model below :

$$p_i = F(\mathbf{x}_{i,p}|\boldsymbol{\beta}_p) = \frac{\exp\{\boldsymbol{\beta}_p^\top \mathbf{x}_{i,p}\}}{1 + \exp\{\boldsymbol{\beta}_p^\top \mathbf{x}_{i,p}\}},$$

where $\boldsymbol{\beta}_p$ is a $p \times 1$ unknown parameter vector. The log-likelihood function of the logistic model using the labeled (training) set, $\{(Y_i, \mathbf{x}_{i,p}), i \in \Xi_s\}$, is

$$L = \sum_{i \in \Xi_s} \left\{ Y_i \ln F(\mathbf{x}_{i,p}|\boldsymbol{\beta}_p) + (1 - Y_i) \ln(1 - F(\mathbf{x}_{i,p}|\boldsymbol{\beta}_p)) \right\}.$$

Thus, we can use the maximum likelihood estimation (MLE) to estimate $\boldsymbol{\beta}_p$. It is known that there is no close-form expression in the MLE approach, and the numerical optimization approach, like Newton's method and iteratively re-weighted least squares (IRLS), is commonly used in this case. Once we obtain the estimate of $\boldsymbol{\beta}_p$, $\hat{\boldsymbol{\beta}}_p$, we can predict the label of the ith observation by

$$\hat{Y}_i = 1 \text{ if } \hat{p}_i = \frac{\exp\{\hat{\boldsymbol{\beta}}_p^\top \mathbf{x}_{i,p}\}}{1 + \exp\{\hat{\boldsymbol{\beta}}_p^\top \mathbf{x}_{i,p}\}} > \alpha,$$

with the pre-specified value α, say, for example, $\alpha = 0.5$.

2.2. Evaluation of subjects via an optimal design criterion

If we use the experimental design methodologies properly to query the next points, the information concealed in a large data set will be extracted quickly. Using some well-developed techniques in design theories (see Atkinson, 1996; Fedorov, 1972), we can effectively select samples, and some analytic results of optimal designs for parameter estimation in logistic models have been derived. In the active learning literature, there are already many optimal design-based active learning approaches for recruiting new samples into training sets, for example, see Cohn et al. (1994b), Cohn (1996) and Deng et al. (2009). (For general information about the optimal design theory, please refer to Silvey, 1980). Because in our current problem we only have unlabeled samples instead of a compact design space as in the conventional design problems, it is hard to apply these analytic designs to our problem, in particular when we consider variable selection as a part in an active learning process. How to quickly locate data points that are close to the analytic ones among a huge data set will be the main issue when applying the design criteria to active learning processes. Thus, it becomes a computational problem instead of a construction problem in this area.

Suppose $\{\mathbf{x}_{i,p} = (x_{i,j})^\top, i \in \Xi_s, j \in \Xi_v\}$ denotes the current labeled point set of size n with a variable length equal to p. Following the definition in the optimal design (Silvey, 1980), we set a design $\xi_{n,p}$ at points $\mathbf{x}_{i,p}$, $i \in \Xi_s$ with equal weights $1/n$. Given the parameter estimate, $\hat{\boldsymbol{\beta}}_p$, the information matrix of the logistic model with respect to $\hat{\boldsymbol{\beta}}_p$ is

$$M(\xi_{n,p}, \hat{\boldsymbol{\beta}}_p) = \boldsymbol{X}_{n,p}^\top \boldsymbol{W}_{\hat{F},p} \boldsymbol{X}_{n,p}/n, \tag{1}$$

where $\boldsymbol{X}_{n,p}$ is the $n \times p$ design matrix with $\mathbf{x}_{i,p}$ as its ith row, and $\boldsymbol{W}_{\hat{F},p}$ is an $n \times n$ diagonal matrix with the ith diagonal element w_{ii} equal to

$$w_{ii} = \hat{F}(\mathbf{x}_{i,p}|\hat{\boldsymbol{\beta}}_p)[1 - \hat{F}(\mathbf{x}_{i,p}|\hat{\boldsymbol{\beta}}_p)],\ i \in \Xi_s.$$

It is clear that the information matrix in (1) depends on the current parameter estimate, and therefore, a "locally optimal" criterion will be used for subject selection consideration (Silvey, 1980). Suppose $\mathbf{x}_{t,p} = (x_{t,j}, j \in \Xi_v)^\top$, $t \in \Xi_s^c$ is an unlabeled subject to be added to the design $\xi_{n,p}$; then, following Fedorov (1972), the $(n + 1)$-points design including $\mathbf{x}_{t,p}$ is $\xi_{n+1,p} = \frac{1}{n+1}\bar{\xi}_{t,p} + \frac{n}{n+1}\xi_{n,p}$, where $\bar{\xi}_{t,p}$ is a design that puts all of the mass at the point $\mathbf{x}_{t,p}$. Therefore, $\xi_{n+1,p}$ is equally supported on the points $\{\mathbf{x}_{i,p}, i \in \Xi_s\} \cup \{\mathbf{x}_{t,p}\}$. We then use the efficiency of $\xi_{n+1,p}$ based on $\xi_{n,p}$ via the relative D-efficiency among the corresponding information matrices,

$$\mathrm{reDeff}(\mathbf{x}_{t,p}) = \frac{|M(\xi_{n+1,p}, \hat{\boldsymbol{\beta}}_p)|^{1/p} - |M(\xi_{n,p}, \hat{\boldsymbol{\beta}}_p)|^{1/p}}{|M(\xi_{n,p}, \hat{\boldsymbol{\beta}}_p)|^{1/p}} = \frac{|M(\xi_{n+1,p}, \hat{\boldsymbol{\beta}}_p)|^{1/p}}{|M(\xi_{n,p}, \hat{\boldsymbol{\beta}}_p)|^{1/p}} - 1, \tag{2}$$

to measure the effectiveness of the new subject, and we want to select the next point, $\mathbf{x}^*$,

$$\mathbf{x}^* = \underset{\{\mathbf{x}_{t,p}, t \in \Xi_s^c\}}{\arg\max}\ \mathrm{reDeff}(\mathbf{x}_{t,p}), \tag{3}$$

which maximizes (2) among all points $\mathbf{x}_{t,p}$, $t \in \Xi_s^c$ based on the labeled training set and current logistic model. Since

$$\mathbf{x}^* = \underset{\{\mathbf{x}_{t,p}, t \in \Xi_s^c\}}{\arg\max}\ \mathrm{reDeff}(\mathbf{x}_{t,p}) = \underset{\{\mathbf{x}_{t,p}, t \in \Xi_s^c\}}{\arg\max}\ |M(\xi_{n+1,p}, \hat{\boldsymbol{\beta}}_p)|,$$

to select a point satisfying (3) is equivalent to that in the locally D-optimal criterion. Because $|M(\xi_{n,p}, \hat{\boldsymbol{\beta}}_p)|^{1/p}$ can be viewed as the geometric mean of the eigenvalues, we apply the pth root in (2) to facilitate interpolation. It follows that Eq. (2) is the ratio of the average information of $\xi_{n,p}$ and $\xi_{n+1,p}$. For further details regarding the relative D-efficiency, please refer to Berger and Wong (2009) and Montgomery (2009).

2.3. Two-stage query procedure

Because a complete search is exhausted and computationally inefficient, when the size of the unlabeled data is huge and the uncertainty sampling strategy can reduce the search time, the idea of uncertainty sampling is popularly used in many active learning processes in the literature. Here, we also adopt this strategy and propose a two-stage procedure. Before applying the methods of experimental design, we will first identify a candidate set based on the current logistic model with a pre-specified threshold value α as follows. For each unlabeled point $\mathbf{x}$, we define $d(\mathbf{x}|\hat{\boldsymbol{\beta}}) = |\hat{F}(\mathbf{x}|\hat{\boldsymbol{\beta}}) - \alpha|$ to measure uncertainty with respect to the current model, $\hat{F}(\cdot|\hat{\boldsymbol{\beta}})$. We could encompass the uncertainty candidate subjects from the unlabeled data set $\{\mathbf{x}_{t,p}, t \in \Xi_s^c\}$, i.e.,

$$\{\tilde{\mathbf{x}}\} = \{\mathbf{x}_{t,p} : d(\mathbf{x}_{t,p}|\hat{\boldsymbol{\beta}}) \le d_0, t \in \Xi_s^c\}, \tag{4}$$

where d_0 is a pre-specified constant to determine the scope of the pool of the candidate subjects. In this paper, we suggest setting $d_0 = d_{(h)}$, where h is a given integer and $d_{(j)}, j = 1, \dots, H$, are the distinct order statistic values of $\{d(\mathbf{x}_{t,p}|\hat{\boldsymbol{\beta}}),\ t \in \Xi_s^c\}$.

To extract the concealed information in these subjects, $\{\tilde{\mathbf{x}}\}$, we select the next labeled point $\mathbf{x}^*$ as the one maximizing the relative D-efficiency in Eq. (3), reDeff criterion. Moreover, if we choose an h equal to the largest integer H, then $\{\tilde{\mathbf{x}}\} = \{\mathbf{x}_{t,p}, t \in \Xi_s^c\}$, and in this case, the two-stage procedure is the same as the locally D-optimal approach. When there is only one element in $\{\tilde{\mathbf{x}}\}$, the proposed query approach is equivalent to the uncertainty sampling.

Remark 1. For given $\alpha \in (0, 1)$, the decision boundary of a logistic model can be defined as $l_\alpha(\mathbf{x}) = \{\mathbf{x} : F(\mathbf{x}|\boldsymbol{\beta}) = \alpha\}$. Deng et al. (2009) treated a binary classification to obtain the separation boundary estimation problem; thus, they chose a few candidates close to the estimated decision boundary based on the current learning model, and then selected their next sample using a locally D-optimal criterion. Their two-stage procedure integrates the concept of uncertainty sampling (Lewis and Gale, 1994) and an optimal design method.

2.4. Grafting technique for greedy selection procedure

Fitting a logistic regression model with a large number of variables and too many redundant variables causes computational difficulties in parameter estimation and enlarges prediction variation. Since having a large number of variables, P, is common in this big data era, we want to identify a compact model for the binary classifier due to sparse model assumption. In this paper, we adopt the concept of greedy forward selection algorithm for variable selection from computational consideration. In fact, for reducing the computational cost, Efron and Hastie (2016) also suggests a forward selection approach to identify a proper classification model.

Singh et al. (2009) introduced a single feature optimization (SFO) procedure for logistic regression models, and this is a greedy-type feature selection procedure in the literature. Suppose x_p is a candidate variable, and in the current logistic model, we do have $x_1, \dots, x_{p-1}$. Then, in SFO, instead of re-estimating the coefficient vector $\boldsymbol{\beta}_p = (\beta_1, \dots, \beta_p)$, it learns an approximate model by fixing the original parameters for $x_1, \dots, x_{p-1}$ and optimizing the parameter of the new variable, β_p via the log-likelihood function L with respect to x_p, i.e.,

$$\hat{\beta}_p = \arg\max_{\beta_p} L.$$

By this method, merely $P - (p - 1)$ approximate models need to be created at each iteration of forward selection. The estimation value of β_p is computed according to Newton's method. To identify the next added variable, Singh et al. (2009) proposed scoring the new feature variable x_p by evaluating the approximate model with a proper evaluating index, like AIC or prediction error.

Instead of evaluating the variable effect based on the approximation of parameter estimation, Perkins et al. (2003) proposed another greedy forward selection approach, called the grafting technique, based on the gradient of the log-likelihood function for the newly added variable. With fixed parameters $\beta_1, \dots, \beta_{p-1}$, the variable with the largest magnitude of gradient is added to the model, i.e.,

$$\arg\max_{p \in \Xi_v^c} \left| \frac{\partial L}{\partial \beta_p} \right| = \arg\max_{p \in \Xi_v^c} \left| \sum_{i \in \Xi_s} x_{i,p}(Y_i - p_i) \right|. \tag{5}$$

Note that based on (5), the grafting technique is similar to the matching pursuit (Mallat and Zhang, 1993) (or weak greedy algorithm) used in the variable selection for the regression model, and in this greedy forward selection algorithm, we only need to compute the inner product operator of $\mathbf{x}_p$ and the response vector.

Because both SFO and the grafting technique sequentially add one variable into the current logistic model, the GATE algorithm typically has a small model when we start with a null model. These approaches can take advantage of the parallel computing techniques (Kubicaa et al., 2011). However, SFO contains a series of one-dimensional optimization problems, while the grafting technique only involves the inner product operator. Due to the computational complexity consideration, we use the grafting technique in our GATE learning algorithm.

2.5. Main algorithm

The proposed GATE algorithm has two parts: (1) identify the proper subjects for labeling, and (2) find a compact classification model. To identify the proper variable to be added at each iteration, additional information is required, and thus, relying on only one newly labeled point is not enough. Thus, we include a batch of size n_q instead, and we believe that this is appropriate for most big data applications. Of course, the batch size n_q can be a tuning parameter in the GATE algorithm and may vary according to the application. Our query procedure relies on the D-optimal criterion. After one iteration of the GATE algorithm, we will have additional n_q points and include a new variable in the model. Then, we use the D-efficiency criterion to measure the improvement of each iteration. As mentioned before, this criterion is the geometric mean with respect to the different model size. Hence, we will stop the iteration of GATE learning algorithm when the relative difference between the D-efficiencies is small enough.

Let ξ_0 be the design with n_q additional points in the current model with k variables, and ξ_1 be the design with n_q additional points with respect to the one more selected variable, i.e., $k+1$ variables. We then stop the GATE algorithm when the design with the additional subjects cannot significantly provide enough information to support the larger model size. Let ε be a pre-specified threshold value; we stop when the following inequality is satisfied:

$$\frac{\left| |M(\xi_0, \hat{\boldsymbol{\beta}}_k)|^{1/k} - |M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1})|^{1/(k+1)} \right|}{|M(\xi_0, \hat{\boldsymbol{\beta}}_k)|^{1/k}} < \varepsilon. \tag{6}$$

The details of each step of the GATE algorithm are stated as Algorithm 1.

```
Initialization: Learn the current logistic classifier model by estimating $\boldsymbol{\beta}_0$ based on the initial labeled set with $n_0$ points;
Let $k = \#\Xi_v$ and set Crit = 1;
while Crit $\geq \varepsilon$ & $k \leq P$ do
    Estimating $\hat{\boldsymbol{\beta}}_k^{(0)} = (\hat{\beta}_v,\ v \in \Xi_v)^\top$ based $\{\mathbf{x}_{i,v},\ i \in \Xi_s,\ v \in \Xi_v\}$;
    (batch active subject learning);
    for $1 \leq t \leq n_q$ do
        for $j \in \Xi_s^c$ do
            Calculate $d_j = |\hat{F}(\mathbf{x}_{j,k}|\hat{\boldsymbol{\beta}}_k^{(t-1)}) - \alpha|$;
        end
        Set $d_0 = d_{(h)}$ as the $h$th order statistic of $d_j$ and $\Xi_I = \{i | i \in \Xi_s^c, d_i \leq d_0\}$;
        Identify the point $\mathbf{x}_{i_t,v} = \arg\max_{\{\mathbf{x}_{t,v}, t \in \Xi_I\}} \text{reDeff}(\mathbf{x}_{t,v})$;
        Query $\mathbf{x}_{i_t,v}$ and denoting $Y_{i_t}$ as its label;
        Update $\Xi_s = \Xi_s \cup \{i_t\}$. Re-estimating $\hat{\boldsymbol{\beta}}_k^{(t)}$ based $\{\mathbf{x}_{i,v},\ i \in \Xi_s,\ v \in \Xi_v\}$;
    end
    Define $\hat{\boldsymbol{\beta}}_k^* = \hat{\boldsymbol{\beta}}_k^{(n_q)}$;
    Compute $M_0 = |M(\xi_0, \hat{\boldsymbol{\beta}}_k^*)|^{1/k}$, where $\xi_0$ equally supports $\{\mathbf{x}_{i,v},\ i \in \Xi_s,\ v \in \Xi_v\}$ and $\hat{p}_i = \hat{F}(\mathbf{x}_{i,v}|\hat{\boldsymbol{\beta}}_k^*)$;
    (variable selecting);
    for $u \in \Xi_v^c$ do
        Compute $g_u = \left| \sum_{i \in \Xi_s} x_{i,u}(Y_i - \hat{p}_i) \right|$;
    end
    Select $u^* = \max_{u \in \Xi_v^c} g_u$ and update $\Xi_{v'} = \Xi_v \cup \{u^*\}$;
    Re-estimate $\hat{\boldsymbol{\beta}}_{k+1}^*$;
    Obtain $M_1 = |M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1}^*)|^{1/(k+1)}$ where $\xi_1$ equally supports $\{\mathbf{x}_{i,v'},\ i \in \Xi_s,\ v' \in \Xi_{v'}\}$;
    Compute Crit= $|M_1 - M_0| / M_0$;
    if Crit $< \varepsilon$ then
        $\Xi_v = \Xi_{v'} \setminus \{u^*\}$;
    else
        Updating $\Xi_v = \Xi_{v'}$ and $k = k + 1$;
    end
end
Finally, re-estimate $\hat{\boldsymbol{\beta}}^*$ based on $\{\mathbf{x}_{i,v},\ i \in \Xi_s,\ v \in \Xi_v\}$ with the selected training data set;
Estimate $\hat{F}(\mathbf{x}_{j,v}|\hat{\boldsymbol{\beta}}^*)$ with $\mathbf{x}_{j,v} = (x_{j,v},\ v \in \Xi_v)^\top$ for all $j$ in the testing data set;
Obtain the estimated labels for the testing data set, that is, $\hat{Y}_j^t = 1$ when $\hat{F}(\mathbf{x}_{j,v}|\hat{\boldsymbol{\beta}}^*) > \alpha$ ;
```

Algorithm 1: Greedy Active Learning Algorithm

Remark 2. Consider the computational complexity of the GATE algorithm. Suppose at the current iteration, there are p variables are chosen in the logistic model. Let n be the labeling points in the current training set and n_r be the number of the unlabeled points. Firstly the computational complexity to identify one next labeled point in the subject selection step is at most $n_r \times O(p^3)$, where $O(p^3)$ is the complexity for computing the determinant of a $p \times p$ information matrix. Secondly for the variable selection part, because the grafting technique only involves the inner product operator, it takes $O(nP)$ for adding one additional variable, where P is the total number of the candidate variables. Therefore, let n_q is the batch size, the GATE algorithm will take, for one iteration,

$$n_q \times (n_r \times O(p^3)) + O(nP).$$

Remark 3. We know that the conventional variable selection schemes do not consider subject selection strategies in their algorithms. However, the subject selection is the highlight of active learning procedures. In this study, we incorporate variable selection and subject selection into an algorithm, hence it is different from those conventional problems. Ideally, one can replace the proposed forward selection scheme with other variable selection approaches. If we adopt a backward selection scheme in active learning algorithm, we also need to adjust the subject selection procedure, accordingly. In this case, the computational complexity will increase for data sets with large p as discussed here. On the other hand, if we adopt a step-wise selection method, then the model will vary a lot during the subject selection of active learning procedures compared to that of the conventional step-wise selection with fixed samples. Hence, directly replacing the current forward selection scheme with other methods may not fully take the advantages of them, and we need to modify our subject selection, accordingly, which will be some potential research problems.

3. Numerical examples

We illustrate the GATE algorithm in terms of the classification rate using some synthesized data sets, the MAGIC gamma telescope data set (Dheeru and Karra Taniskidou, 2017) and a well-known wave data set from Breiman et al. (1984) based on the training sample size used and the variable selection status.

3.1. Synthesized data

Let $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,P})^\top$ be a $P \times 1$ predictor vector and $\boldsymbol{\beta}$ be the corresponding parameter vector, and for the ith predictor vector $\mathbf{x}_i$, we assume that $x_{i,1} = 1$ and $x_{i,j}, j > 1$ are independently generated from a normal distribution with mean μ_j and the unit standard deviation, where μ_j is assumed from the uniform distribution within $(-1, 1)$. Let P be 100; then, we generate the response, Y_i's, from Bernoulli distribution with the probability

$$p_i = \frac{\exp\{\boldsymbol{\beta}^\top \mathbf{x}_i\}}{1 + \exp\{\boldsymbol{\beta}^\top \mathbf{x}_i\}},$$

and consider the following sparse model scenarios with the different parameter vectors:

Case 1: The first five parameters are chosen as follows, $(\beta_1, \beta_2, \ldots, \beta_5)^\top = (0.5, -2.0, -0.6, 0.5, 1.2)$, and the other $\beta_i, i > 5$, are set as zeros.
Case 2: The parameter vector is $(\beta_1, \beta_2, \ldots, \beta_5)^\top = (5, -20, -6, 5, 12)$, which is equal to 10 times of the vector in Case 1.
Case 3: The number of nonzero parameters are 6, and these 6 parameters are set as $(\beta_1, \beta_2, \ldots, \beta_6)^\top = (1, -4, -2, 2, 3, 7)$. The other parameters are fixed as zeros.

For each logistic regression model, we generate 20,000 independent samples, which contains a training and a testing data set with sample sizes $N_t = 15{,}000$ and $N_v = 5000$, respectively. There are 1000 replications for each case.

We randomly select $n_0 = 100$ points from the potential training set as an initial labeled training subject set for the GATE algorithm and pretend that, in this simulation study, the remaining unselected training points remain unlabeled. At each iteration, we set $h = 200$ as the scope for the pool of the candidate subjects and the batch size $n_q = 30$. Thus, we sequentially select 30 points using a two-stage query procedure. The label information of points is revealed once they are selected but not before the selection. We assume no prior knowledge for the samples; hence, we set a threshold value, $\alpha = 0.5$, for the logistic classification probability as follows:

$$\hat{Y}_i = 1 \text{ if } \hat{p}_i = \frac{\exp\{\hat{\boldsymbol{\beta}}^\top \mathbf{x}_i\}}{1 + \exp\{\hat{\boldsymbol{\beta}}^\top \mathbf{x}_i\}} > \alpha(= 0.5),$$

and we set $\varepsilon = 10^{-2}$ for the stopping criterion stated in (6).

We use both Accuracy (ACC), defined as the proportion of true results (both true positives and true negatives) among the total number of cases examined and Area Under the Curve (AUC) of the receiver operating characteristic (ROC) to evaluate the performance in all cases. Both measurements target the fraction of all instances that are correctly categorized. Thus, they are larger-the-better characteristics.

To assess the variable selection of the proposed algorithm, we first let $\mathcal{J}$ be the set of true active variables and J be the set of variables identified via the considered learning procedure, i.e.,

$$\mathcal{J} = \{j : \beta_j \neq 0\}, \text{ and } J = \{k : \hat{\beta}_k \neq 0\}.$$

We then define the True Positive Rate (TPR) and the False Positive Rate (FPR) below:

$$\text{TPR} = \frac{\text{number of } \{J \cap \mathcal{J}\}}{\text{number of } \{\mathcal{J}\}},$$
$$\text{FPR} = \frac{\text{number of } \{J \cap \mathcal{J}^c\}}{\text{number of } \{\mathcal{J}^c\}},$$

and use them to evaluate the variable selection performance. Because TPR is the rate of active variables identified correctly, it is the larger the better, whereas FPR is defined as the rate of inactive variables that are included in the model, the smaller value of FPR indicates a better performance.

Table 1 summarizes the results of the 1000 simulations with respect to three different parameter cases. Among these three cases, the first case should be the one with the worst classification rate due to the small non-zero parameter values. As shown in Table 1, both classification measurements of Case 1, ACC and AUC are still larger than 0.82, and on average, we only take around 530 points from the whole candidate set, which contains 15,000 points. The TPR values to evaluate the variable selection performance for these three cases are higher than 0.94. This means that most of the true active variables are included in the final model; however, over-selection problems do exist due to the potential weakness of the greedy forward selection algorithm. Finally, without taking the labeling cost into account, the proposed algorithm is quite efficient because the average CPU times are less than 4.5 mins.

In order to illustrate the advantages of the proposed GATE algorithm, we compare the classification performance with those of the three different approaches. The first one is to implement the logistic classification with 15,000 training samples and 100 variables. Second, we perform a comparison with an approach where in each replication, following the same sample size n determined by the proposed active learning procedures, we randomly sample the labeled point from the training set and then implement the logistic classifier based on the same active variables identified by grafting forward selection. In the third approach, the 100 variables are taken into the logistic model, and then due to the sample size in each of our 1000 replications, we randomly select the labeled points from the 15,000 candidate training points for learning the corresponding logistic classifier. Tables 2–4 show the classification results with respect to the different approaches. In addition to summarizing the results in the tables, we also illustrate them in Figs. 1–3 with respect to the different parameter cases. In each figure, we not only display the selection frequencies of the predictors (variables) for our GATE learning approach but also show the 1000 ROC curves for all replications with respect to the four different approaches and we then compare our approaches with three scenarios: (1) the data with whole samples and variables ("FUv-FUn"); (2) the data with the random subjects and selected variables ("GAv-GARn") and (3) the data with random subjects and whole variables ("FUv-GARn").

Because the GATE algorithm simultaneously selects subjects for training and identifies a proper model during its training course and there is a lack of active algorithms with both features, we only compare our results using the reDeff criterion with active learning approaches with a subject selection feature only. Hence, we compare our approach further with other two scenarios: (4) one is the data using whole variables with reDeff criterion to select the subjects with a pre-fixed size as that of GATE ("FUv-GAn-RE"), and (5) another one is the data using a pre-fixed model with subjects selected by reDeff ("GAv-REn-RE").

FUv-FUn scenario. Table 2 shows the classification results based on 1000 replications for each parameter case. Here, we can treat these results in Table 2 as the baselines for comparison. Due to these two classification measurements, ACC and AUC, we focus on the classification results for testing sets. The lowest value of the relative ACC of the proposed GATE approach and the baseline values is $0.821/0.832 = 0.987$, and the lowest relative AUC value is $0.886/0.898 = 0.986$. Both relative values are close to 1.000. This means that the classification performances are similar for both approaches. However, in our GATE learning approach, we averagely take less than 530 points, which is around $1/30$ of the whole samples, and a more compact model is used in the corresponding logistic classifier. Thus, this is the evidence to show the efficiency of the proposed GATE learning approach, whether in the proposed query strategy or the greedy forward selection approach. The sub-figures (b) and (c) of Figs. 1–3 show that both approaches have similar classification results, however the results under FUv-FUn are more stable based on the 1000 replications.

GAv-GARn scenario. In each replication, we fix the model selected by our GATE learning approach, and the labeled subject set is used to randomly sample the subjects from the whole samples, and its size is the same as the one we used in our GATE learning. Then, we learn the corresponding logistic classifier. The results with 1000 replications are recorded in Table 3. Consider the classification results for the testing set. The relative values of ACC and AUC of our GATE learning algorithm and this method are all slightly larger than 1. That is, our GATE algorithm still performs better in these three cases. In addition, it is evidence in support of our greedy selection approach because we still can achieve good classification results based on these selected models. The sub-figures (b) and (d) in Figs. 1–3 also indicate that both approaches have almost the same classification performance.

FUv-GARn scenario. In this classification approach, we take all 100 variables into the logistic regression model. To learn the binary classifier, we randomly select the points from the whole samples with the same sample size used in GATE for each

Table 1
The values of ACC, AUC, TPR, FPT and average CPU times of GATE based on 1000 simulations.

Case	p_o	n	Training		Testing		Select			Ave. time (min)
			ACC	AUC	ACC	AUC	TPR	FPR	$\#\{x_d\}$	
1	5	528.460 (190.255)	0.825 (0.029)	0.887 (0.015)	0.821 (0.029)	0.886 (0.015)	0.941 (0.148)	0.111 (0.061)	14.282 (6.342)	4.470
2	5	314.290 (65.241)	0.981 (0.005)	0.998 (0.003)	0.980 (0.005)	0.998 (0.003)	0.999 (0.026)	0.033 (0.023)	7.143 (2.175)	1.549
3	6	360.250 (78.794)	0.946 (0.016)	0.987 (0.012)	0.944 (0.017)	0.987 (0.012)	0.993 (0.068)	0.040 (0.027)	8.675 (2.626)	1.950

Note: p_o represents the number of non-zero parameters in the true generating model, and the standard deviation of the 1000 simulated repeats is shown in parentheses.

Table 2
The values of ACC, AUC, TPR, and FPT of under FUv-FUn scenario based on 1000 simulations.

Case	p_o	n	Training		Testing	
			ACC	AUC	ACC	AUC
1	5	15 000	0.832 (0.025)	0.898 (0.005)	0.832 (0.025)	0.898 (0.005)
2	5	15 000	0.982 (0.004)	0.999 (0.000)	0.982 (0.004)	0.999 (0.000)
3	6	15 000	0.949 (0.009)	0.990 (0.001)	0.949 (0.010)	0.989 (0.002)

Note: p_o represents the number of non-zero parameters in the true generating model, and the standard deviation of the 1000 simulated repeats is shown in parentheses.

replication. Table 4 shows the average classification results based on 1000 replications. Compared with the ACC and AUC values shown in Table 1, our active learning approach significantly outperforms this random selection approach. This means that to obtain better classification results, we need to have more labeled samples if we still want to take whole variables into the model or, due to a smaller sample set, we need to have a compact logistic model. Thus, the one possible solution is our proposed active learning approach because it can be used to sequentially select the next labeled points but also identify the active variables for the binary classifier. Based on the sub-figure (e) in these three figures, the instability of this classification approach is clearly illustrated.

FUv-GAn-RE scenario. Table 5 summarizes results of logistic models under FUv-GAn-RE scenario with a fixed sample size as that in the GATE algorithm based on the 1000 replications. We found that overall performance is worse than that in FUv-GARn. This may be due to the selection criterion, since the reDeff criterion is model-dependent. Because the logistic model with 100 variables is very different from the true model, these results imply that the points identified by the reDeff criterion cannot properly improve the logistic model fitting.

GAv-REn-RE scenario. Under GAv-REn-RE setup, we stop the active learning when the relative difference between the D-efficiencies of two consecutive iterations is less than 10^{-2}. Table 6 is the results of 1000 replications under GAv-REn-RE. The subject size of Case 1 is less than what we are accustomed to and the other two sizes are larger than that of the GATE. However, all these differences are less than one standard deviation of sample sizes based on 1000 replications. We know from Table 1 that the model identified with the GATE algorithm is very close to the true one. Thus, with this advantage, the performance of GAv-REn-RE is also close to that of the GATE algorithm.

Remark 4. To use our GATE algorithm, we need to specify (1) the initial sample size, n_0, (2) the batch size n_q, (3) uncertainty sample step h and (4) threshold value in the stopping criterion ε. Selecting the most suitable parameters for a given data set is always an important and time-consuming issue, and cross validation (CV) is a commonly used tuning procedure. As we have reported in Case 2, we simply fix $\varepsilon = 10^{-2}$, and choose three candidate parameter vectors, $(n_0, n_q, h) = (50, 20, 200)$; $(100, 30, 200)$ and $(150, 40, 300)$. We then use a 5-fold CV for parameter tuning, and the selection principle is the average ACC in CV. After 100 replications, the average ACC for the testing set is 0.981, and TPR for the variable selection is 1.000. Both measurements are slightly better than what we showed in Table 1. The differences are not significant. The corresponding CPU time is 26.6999 mins because we need to run the GATE algorithm $3 \times 5 + 1 = 16$ times.

3.2. Real and artificial examples

We apply the GATE algorithm to two data sets with the same tuning parameters stated in Section 3.1. That is that the size of the initial training set, n_0, is equal to 100 points, the order for the subject candidate is $h = 200$, the batch size, n_q is equal to 30, and set the threshold value $\varepsilon = 10^{-2}$ for the stopping criterion.

Table 3
The values of ACC and AUC under GAv-GARn scenario based on 1000 simulations.

Case	p_o	n	Training		Testing	
			ACC	AUC	ACC	AUC
1	5	528.460 (190.255)	0.820 (0.029)	0.885 (0.014)	0.819 (0.029)	0.884 (0.015)
2	5	314.290 (65.241)	0.974 (0.006)	0.997 (0.003)	0.974 (0.007)	0.997 (0.003)
3	6	360.250 (78.794)	0.941 (0.016)	0.986 (0.012)	0.940 (0.017)	0.986 (0.012)

Note: p_o represents the number of non-zero parameters in the true generating model, and the standard deviation of the 1000 simulated repeats is shown in parentheses.

Table 4
The values of ACC and AUC under FUv-GARn scenario based on 1000 simulations.

Case	p_o	n	Training		Testing	
			ACC	AUC	ACC	AUC
1	5	528.460 (190.255)	0.763 (0.055)	0.816 (0.061)	0.759 (0.056)	0.812 (0.061)
2	5	314.290 (65.241)	0.858 (0.030)	0.922 (0.033)	0.856 (0.031)	0.921 (0.033)
3	6	360.250 (78.794)	0.851 (0.031)	0.917 (0.034)	0.848 (0.031)	0.915 (0.034)

Note: p_o represents the number of non-zero parameters in the true generating model, and the standard deviation of the 1000 simulated repeats is shown in parentheses.

Table 5
The values of ACC and AUC under FUv-GAn-RE scenario based on 1000 simulations.

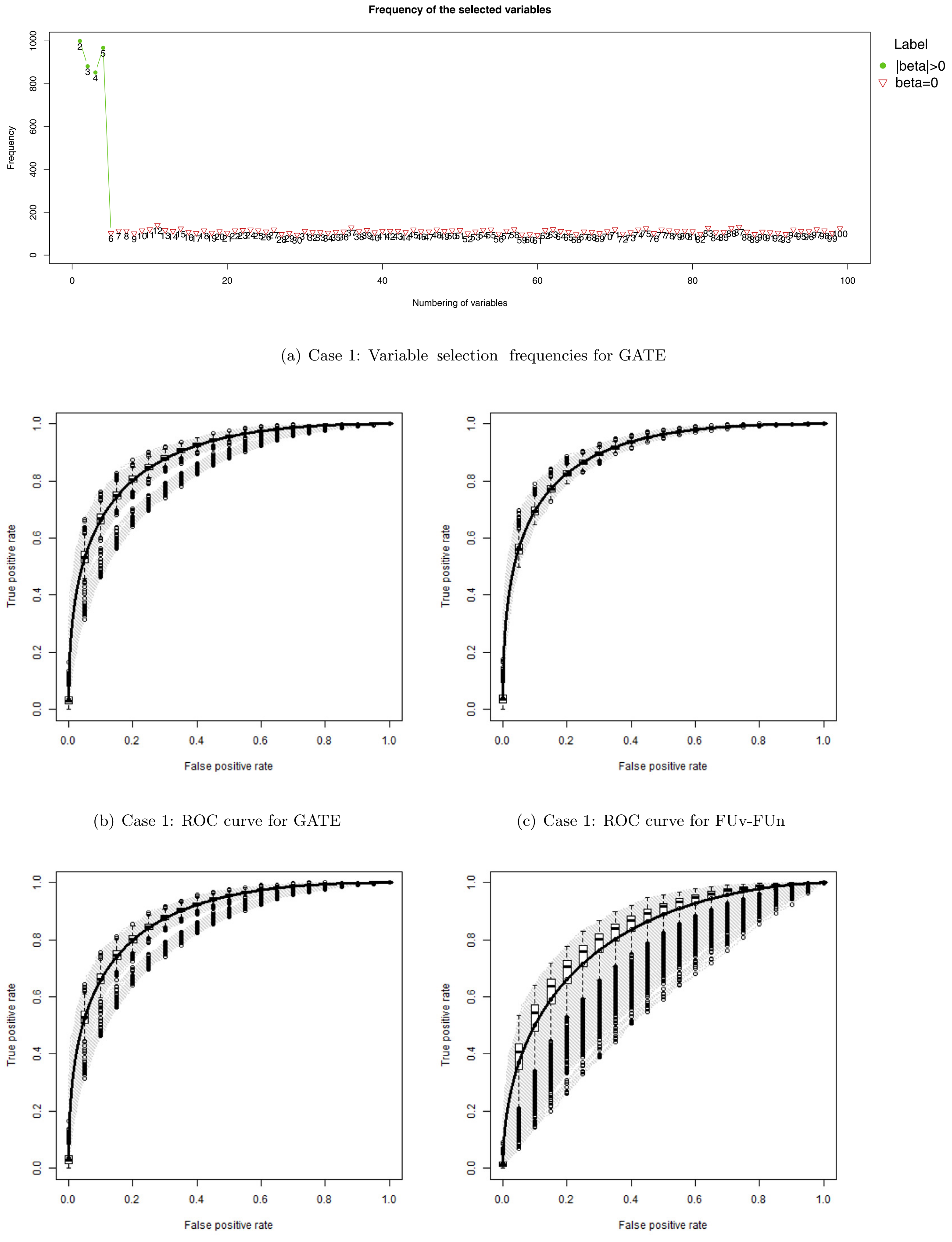
Case	p_o	n	Training		Testing	
			ACC	AUC	ACC	AUC
1	5	528.460 (190.255)	0.660 (0.089)	0.676 (0.091)	0.650 (0.088)	0.674 (0.089)
2	5	314.290 (65.241)	0.841 (0.085)	0.900 (0.087)	0.838 (0.086)	0.898 (0.088)
3	6	360.250 (78.794)	0.841 (0.078)	0.903 (0.081)	0.837 (0.078)	0.901 (0.081)

Note: p_o represents the number of non-zero parameters in the true generating model, and the standard deviation of the 1000 simulated repeats is shown in parentheses.

Table 6
The values of ACC and AUC under GAv-REn-RE scenario based on 1000 simulations.

Case	p_o	n	Training		Testing	
			ACC	AUC	ACC	AUC
1	5	435.550 (140.150)	0.824 (0.030)	0.887 (0.016)	0.821 (0.030)	0.886 (0.016)
2	5	396.070 (108.453)	0.981 (0.005)	0.998 (0.003)	0.980 (0.006)	0.998 (0.003)
3	6	396.040 (102.529)	0.947 (0.016)	0.987 (0.012)	0.945 (0.017)	0.987 (0.012)

Note: p_o represents the number of non-zero parameters in the true generating model, and the standard deviation of the 1000 simulated repeats is shown in parentheses.



(a) Case 1: Variable selection frequencies for GATE

(b) Case 1: ROC curve for GATE

(c) Case 1: ROC curve for FUv-FUn

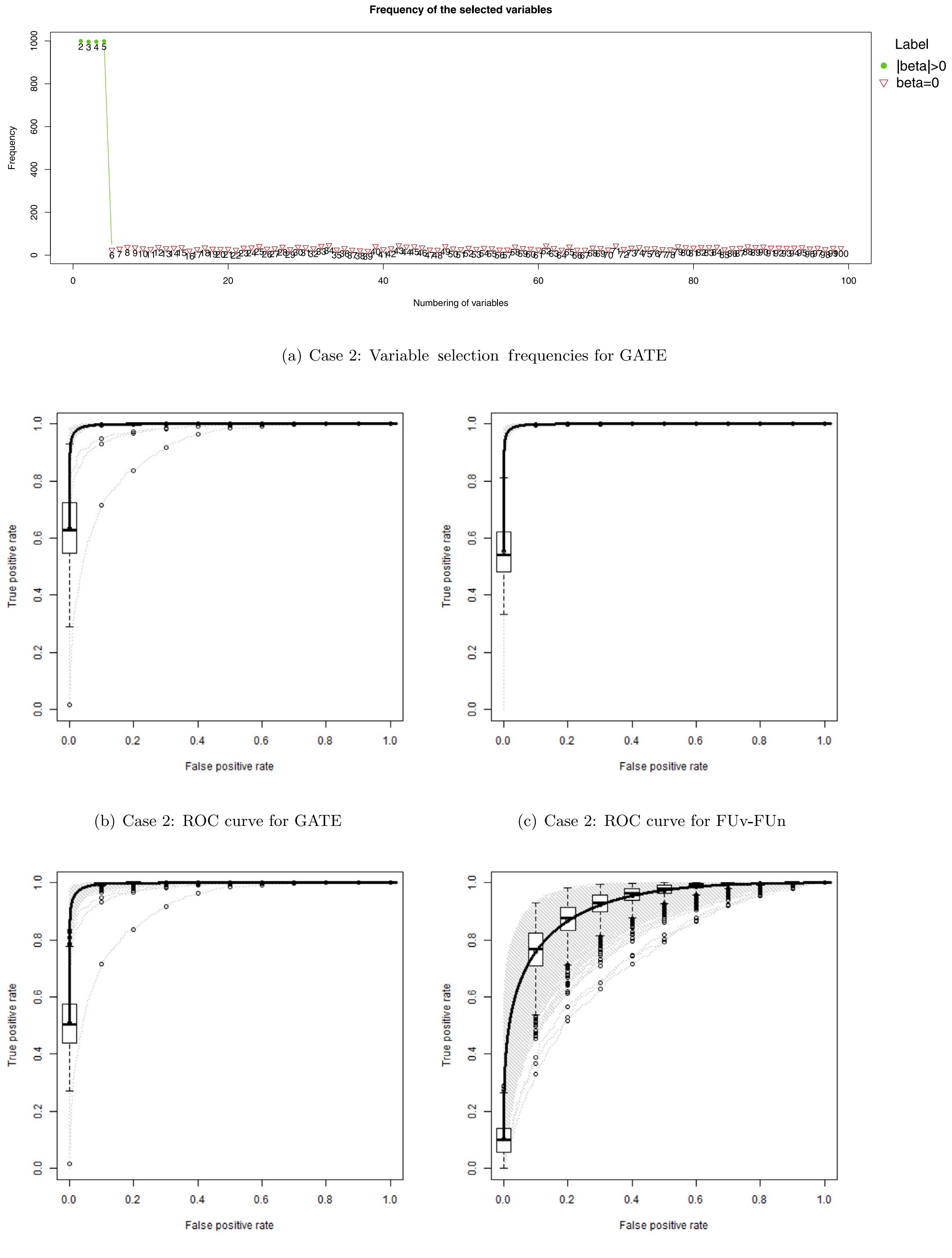
(d) Case 1: ROC curve for GAv-GARn

(e) Case 1: ROC curve for FUv-GARn

Fig. 1. The performances of the different approaches for Case 1 with 1000 replications.

3.2.1. MAGIC gamma telescope data set

The MAGIC gamma telescope data set is a Monte Carlo program simulating registration of high energy gamma particles in a ground-based atmospheric Cherenkov gamma telescope using the imaging technique. This data set consists of 19 020



(a) Case 2: Variable selection frequencies for GATE

(b) Case 2: ROC curve for GATE

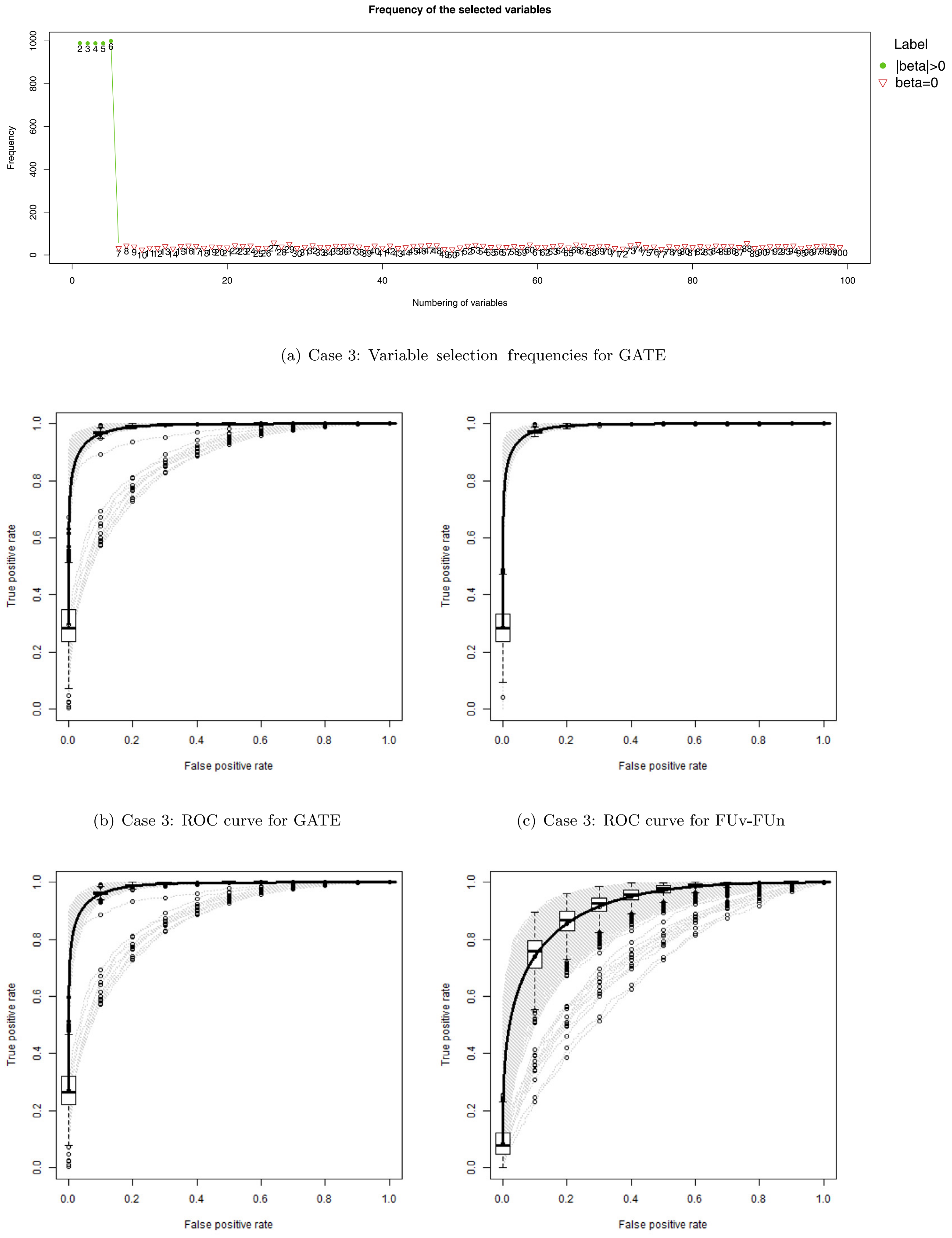
(c) Case 2: ROC curve for FUv-FUn

(d) Case 2: ROC curve for GAv-GARn

(e) Case 2: ROC curve for FUv-GARn

Fig. 2. The performances of the different approaches for Case 2 with 1000 replications.

data points in 2 classes (gamma and hadron classes) with 10 quantitative variables and is available at the Machine Learning Repository website (Dheeru and Karra Taniskidou, 2017), where readers can find more references about it.



(a) Case 3: Variable selection frequencies for GATE

(b) Case 3: ROC curve for GATE

(c) Case 3: ROC curve for FUv-FUn

(d) Case 3: ROC curve for GAv-GARn

(e) Case 3: ROC curve for FUv-GARn

Fig. 3. The performances of the different approaches for Case 3 with 1000 replications.

Our results are based on 100 replications, and a 5-fold cross validation is adopted. In each replication, the data set is randomly partitioned into 5 disjoint subsets, and we use 4 of them together as the training set and compute the ACC and AUC using the remaining set, which is not used for training. The averages of 5 ACCs and AUCs are reported.

Table 7
The results of the magic example.

Method	n	5-fold cross-validation		Variable size
		ACC	AUC	$\#\{x_d\}$
GATE	397.900 (37.088)	0.788 (0.003)	0.816 (0.006)	9.048 (1.667)
FUv-FUn		0.791	0.839 (0.000)	
GAv-GARn	397.900 (37.088)	0.784 (0.004)	0.826 (0.006)	
FUv-GARn	397.900 (37.088)	0.783 (0.003)	0.829 (0.004)	

In addition, the other three methods — FUv-FUn, GAv-GARn and FUv-GARn as described in Section 3.1, are implemented for the comparison purpose. The results are summarized in Table 7. On average, the GATE algorithm selects less than 400 samples among 15,216 candidate points, and overall, 9 variables are included to learn the classifier. The average value of the classification rate, ACC, and AUC value are 0.788 and 0.816, respectively, among 100 5-fold cross-validation replications. Using the results of the logistic model with whole variables and full samples (FUv-FUn), as a baseline and comparing the results of the GATE algorithm, we found that the GATE only takes $4/152 \simeq 2.63\%$ of the training samples with the corresponding relative ACC and AUC equal to $0.788/0.791 \simeq 0.996$ and $0.816/0.839 \simeq 0.973$, which are close to 1. This means that the GATE algorithm exhibits a competitive performance with only a small fraction of the labeled samples. Under GAv-GARn scenario as mentioned in Section 3.1, taking advantage of the compact model, its performance should be similar to what the GATE algorithm does. Thus, from Table 7, the performance of GAv-GARn is just as good as expected because its ACC and AUC values are all close to ours and those of FUv-FUn. On the other hand, the performance of FUv-GARn is similar to that of the GATE algorithm; the GATE algorithm nearly selects the whole set of variables; therefore, the sparse learning model offers no advantage in this case.

3.2.2. Wave data set

A well-known artificial wave data set (Breiman et al., 1984) is also used for illustration purposes. Originally, there are three classes with 21 variables in this wave data set, and the variables of each class are generated based on a random convex combination of two of three wave forms with noise. Rakotomalala (2005) expanded the data set by adding noise variables and generating more subjects. He also modified this data set as a binary response data set by only keeping the subjects of the first two classes. This binary classification data set can be downloaded from the following link, http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/wave_2_classes_with_irrelevant_attributes.zip.

Currently, there are 33,334 subjects in this data set, and 10,000 subjects are collected as the training set. It is a balanced classification problem because the subject numbers in both classes are almost the same. In addition to the original 21 predictors, 100 noise variables are added, and they are completely independent from the corresponding classification problem. Rakotomalala (2005) analyzed this data set via a free data mining software, "TANAGRA", and he not only used logistic regression for the classification problem but also performed forward selection according to a SCORE test based on the whole training set. Overall, the classification error rate for the testing set was 7.87%. For the variable selection results, no noise variables are selected into the classification model, and 15 variables from the original 21 active variables are identified as active variables by setting the significant level as 1% in the SCORE test. For more details, please refer to the following website: http://data-mining-tutorials.blogspot.tw/2008/12/logistic-regression-software-comparison.html.

Here, we repeat our GATE learning procedure 100 times by randomly generating the initial design set from the training set and treat the remaining points in the training set as the unlabeled points. In addition to the proposed approach, we also implement the other three types of methods as shown in Section 3.1. The overall performances of these four approaches are summarized in Table 8. Basically, the proposed *GATE* algorithm selects an average of 743 samples to learn the classification model. Then, based on the testing set, the average value of the classification rate is 0.920, and the mean value of the AUC value among 100 replications is 0.980. Compared with the classification error rate of TANAGRA, our classification result is quite good, in particular when we only use less than 7.5% of the entire training samples. For the other three approaches, first, the logistic classification procedure with the whole predictors and the training set has identical results to those of TANAGRA, and it can only be implemented one time. The relations among the performances of these methods are similar to that in the simulation studies under FUv-GARn and GAv-GARn scenarios.

Overall, the case with random subjects and whole variables has the worst performances in terms of ACC and AUC. This should also be related to the fact that the subject size is too small compared to the model size, and thus, the uncertainty of the classification model increases. When we fixed the variables as the selected variables, the performance of this case was similar to those of our proposed GATE learning procedure.

It is of interest to investigate the variable selection results. Given the whole training set, TANAGRA can identify the important variables via the forward selection procedure according to the SCORE test. There are 15 variables, V_4, V_5, V_8, V_9, V_{10}, V_{11}, V_{12}, V_{13}, V_{14}, V_{15}, V_{16}, V_{17}, V_{18}, V_{19} and V_{20}, identified as the important variables. Consider the proposed GATE learning

Table 8
The results of the wave example.

Method	n	Training		Testing		Variable size
		ACC	AUC	ACC	AUC	$\#\{x_d\}$
GATE	743.500 (444.264)	0.920 (0.002)	0.978 (0.001)	0.920 (0.002)	0.980 (0.001)	22.45 (14.809)
FUv-FUn	10 000			0.921	0.980	121
GAv-GARn	743.500 (444.264)			0.917 (0.003)	0.978 (0.001)	22.45 (14.809)
FUv-GARn	743.500 (444.264)			0.857 (0.019)	0.937 (0.014)	121

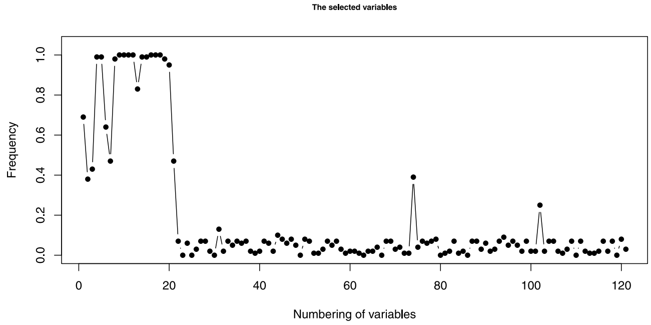


Fig. 4. The variable selection frequencies with wave data set.

procedure. The selected frequencies of each variable are shown in Fig. 4. Suppose we set 0.8 as a threshold value for the selected frequencies. Then, it is clear that our grafting approach can also yield the same set of active variables. For these selected 15 variables, except V_{13}, the selected frequencies are higher than and equal to 95%. In fact, the frequency at which V_{13} is identified is at least 83%. For 100 noise variables, the selected frequencies are essentially lower than those of the first 21 original variables.

4. Discussion and conclusion

In this study, we propose a logistic model-based active learning procedure for binary response data named GATE algorithm. In addition to the common subject selection feature in active learning procedures, our algorithm can also identify the proper classification model with the given data. We propose a two-stage subject selection procedure combining the ideas of uncertainty sampling and relative D-efficiency (reDeff) criterion from the experimental design methods. In our algorithm, we use the grafting technique, a greedy forward selection procedure, and adopt a sequential batch selection strategy at each stage. Hence, the proposed active learning algorithm will repeat the subject selection and forward variable selection steps until a stopping criterion is fulfilled. Both numerical results with our synthesized data and two well-known data sets support the success of the proposed method.

We have some discussions about the stopping criterion, Eq. (6), in the following. According to the definition of the D-efficiency, it might not be easier to theoretically show the relation among $|M(\xi_0, \hat{\boldsymbol{\beta}}_k)|^{1/k}$ and $|M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1})|^{1/(k+1)}$. Due to the consistency of the $\boldsymbol{\beta}$ estimate, we could assume that $\hat{\boldsymbol{\beta}}_{k+1}^\top = (\hat{\boldsymbol{\beta}}_k^\top\ \hat{\beta}_{k+1})$, where $\hat{\beta}_{k+1}$ is the estimate of the $(k+1)$th variable. Therefore we decompose the information matrix $M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1})$ as

$$M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1}) = \begin{bmatrix} M(\xi_0, \hat{\boldsymbol{\beta}}_k) & Q \\ Q^\top & D_{k+1,k+1} \end{bmatrix},$$

where Q is a $k \times 1$ vector and $D_{k+1,k+1}$ is a real number, and we have that

$$|M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1})| = |M(\xi_0, \hat{\boldsymbol{\beta}}_k)|(D_{k+1,k+1} - Q^\top M(\xi_0, \hat{\boldsymbol{\beta}}_k)^{-1}Q).$$

Here $(D_{k+1,k+1} - Q^\top M(\xi_1, \hat{\boldsymbol{\beta}}_k)^{-1}Q) > 0$, because the information matrix is positive definite. Then

$$|M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1})|^{1/(k+1)} = (|M(\xi_0, \hat{\boldsymbol{\beta}}_k)|^{1/k})^{k/(k+1)}(D_{k+1,k+1} - Q^\top M(\xi_0, \hat{\boldsymbol{\beta}}_k)^{-1}Q)^{1/(k+1)}.$$

When k is large, we can have that $|M(\xi_1, \hat{\boldsymbol{\beta}}_{k+1})|^{1/(k+1)} \approx |M(\xi_0, \hat{\boldsymbol{\beta}}_k)|^{1/k}$, because $k/(k+1) \approx 1$ and $(D_{k+1,k+1} - Q^\top M(\xi_0, \hat{\boldsymbol{\beta}}_k)^{-1}Q)^{1/(k+1)} \approx 1$. Thus at least we can guarantee that the difference among the D-efficiencies for the designs

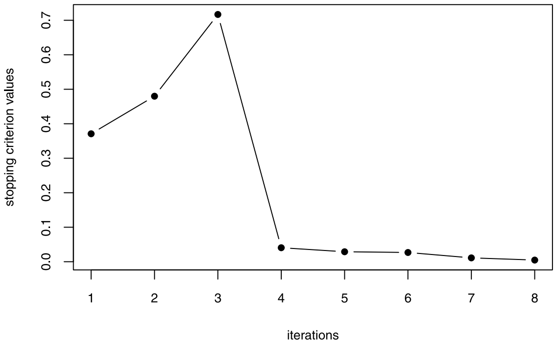


Fig. 5. The trend of the stopping criterion values with respect to the number of iterations of the GATE algorithm.

ξ_0 and ξ_1 should be small when we add more and more variables into the classification model. That is Eq. (6) should be small when we have more variables. In addition, we re-visit the Case 2 in Section 3.1 to illustrate the trend of our stopping criterion values which is shown in Fig. 5. It seems that after 4 iterations, the relative difference is smaller and smaller until its value is less than $\varepsilon = 10^{-2}$.

Albert and Anderson (1984) noted that the MLE of the regression coefficients of logistic models might not exist if the observed data are completely or quasi-completely separated by a hyperplane spanned by the variables considered in the model. The Bayesian approach with a proper prior is one possible outlet to overcome this fitting problem. Another possible solution is reducing the number of variables in the model, which can usually be accomplished with a forward or stepwise selection approach as stated in SAS usage note 22 599, http://support.sas.com/kb/22/599.html. Thus, we modify the grafting technique procedure in the following way: We fit the logistic model based on this new model when we add a new variable, and if the MLE of the parameter vector does not exist after adding this variable, we then remove this newly added variable and add the next candidate variable based on the grafting technique. At this stage, we will continue this procedure until we obtain a model such that the MLE of the parameter vector does exist.

Because we consider subject and variable selection simultaneously, the proposed algorithm tends to over-select variables, that is common phenomenon in the forward selection-based procedures. A natural approach to avoid this situation is to consider a stepwise selection approach instead of a simple forward selection procedure. Once users adopt the stepwise approach, they will have the chance to remove redundant variables from their classification models at the backward elimination step. Certainly, the computational time will be largely extended for this kind of approach; thus, users should take this extra computational cost into consideration for choosing the most proper approach in their applications. The L_1 regularized logistic regression approach, like GLMNET (Friedman et al., 2010), may be another possibility. However, from the computational perspective, the grafting technique still has its advantages for our algorithm.

Acknowledgments

The research of Prof. Hsu was partly supported by the Ministry of Science and Technology (MOST) of Taiwan with Grant No., MOST 105-2118-M-390-004. The research of Dr. Chang was partly supported by the Ministry of Science and Technology (MOST) of Taiwan with Grant No., MOST 106-2118-M-001-007-MY2 and MOST 103-2118-M-001-002-MY3. The research of Prof. Chen was partially supported by the Ministry of Science and Technology (MOST) of Taiwan with Grant No., MOST 105-2628-M-006-002-MY2 and the Mathematics Division of the National Center for Theoretical Sciences in Taiwan.

References

Albert, A., Anderson, J.A., 1984. On the existence of maximum likelihood estimates in logistic regression models. Biometrika 71, 1–10.
Atkinson, A.C., 1996. The usefulness of optimum experimental designs. J. R. Stat. Soc. Ser. B Stat. Methodol. 58 (1), 59–76.
Berger, M.P.F., Wong, W.K., 2009. An Introduction to Optimal Design for Social and Biomedical Eesearch. JohnWiley&Sons., Chichester, UK.
Breiman, L., Friedman, J., Olshen, R., Stone, C., 1984. Classification and Regression Trees. Wadsworth Statistical Press, Belmont, CA.
Cohn, D.A., 1996. Neural network exploration using optimal experiment design. Neural Netw. 9 (6), 1071–1083.
Cohn, D., Atlas, L., Ladner, R., 1994a. Improving generalization with active learning. Mach. Learn. 15 (2), 201–221.
Cohn, D.A., Ghahramani, Z., Jordan, M.I., 1994b. Active learning with statistical models. In: Proceedings of the 7th International Conference on Neural Information Processing Systems. NIPS'94, MIT Press, Cambridge, MA, USA, pp. 705–712.
Culver, M., Kun, D., Scott, S., 2006. Active learning to maximize area under the ROC curve. In: Sixth International Conference on Data Mining, 2006. ICDM '06.
Deng, X., Joseph, V.R., Sudjianto, A., Wu, C.F.J., 2009. Active learning through sequential design, with applications to detection of money laundering. J. Amer. Statist. Assoc. 104 (487), 969–981.
Dheeru, D., Karra Taniskidou, E., 2017. UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences.
Efron, B., Hastie, T., 2016. Computer Age Statistical Inference: Algorithms, Evidence, and Data Science, first ed. Cambridge University Press, New York, NY, USA.

Fedorov, V., 1972. Theory of Optimal Experiments. In: Probability and Mathematical Statistics, Elsevier Science.

Friedman, J., Hastie, T., Tibshirani, R., 2010. Regularization paths for generalized linear models via coordinate descent. J. Stat. Softw. 33 (1), 1–22.

Hsu, D.J., 2010. Algorithms for Active Learning (Ph.D. thesis), Columbia University.

Kubicaa, J., Singhb, S., Sorokinac, D., 2011. Parallel large-scale feature selection. In: Scaling Up Machine Learning: Parallel and Distributed Approaches.

Lewis, D.D., Gale, W.A., 1994. A sequential algorithm for training text classifiers. In: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR '94, Springer-Verlag New York, Inc., New York, NY, USA, pp. 3–12.

Long, B., Bian, J., Chapelle, O., Zhang, Y., Inagaki, Y., Chang, Y., 2010. Active learning for ranking through expected loss optimization. In: SIGIR 2010.

Mallat, S.G., Zhang, Z., 1993. Matching pursuits with time-frequency dictionaries. IEEE Trans. Signal Process. 41 (12), 3397–3415.

Montgomery, D.C., 2009. Design and Analysis of Experiments, seventh ed. JohnWiley&Sons., Hoboken, NJ, USA.

Perkins, S., Lacker, K., Theiler, J., 2003. Grafting: Fast, incremental feature selection by gradient descent in function space. J. Mach. Learn. Res. 3, 1333–1356.

Rakotomalala, R., 2005. TANAGRA: a free software for research and academic purposes. In: Proceedings of European Grid Conference 2005, RNTI-E-3, Vol. 2. pp. 697–702.

Settles, B., 2009. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison.

Settles, B., 2011. From theories to queries: Active learning in practice. In: Journal of Machine Learning Research, Workshop on Active Learning and Experimental Design, Workshop and Conference Proceedings, Vol. 16, pp. 1–18.

Settles, B., 2012. Active Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, San Rafael.

Silvey, S.D., 1980. Optimal Design : An Introduction to the Theory for Parameter Estimation. Chapman and Hall, London ; New York.

Singh, S., Kubica, J., Larsen, S., Sorokina, D., 2009. Parallel large scale feature selection for logistic regression. In: SDM. SIAM, pp. 1172–1183.

Whitney, A.W., 1971. A direct method of nonparametric measurement selection. IEEE Trans. Comput. 20 (9), 1100–1103.