Department of Policy and Planning Sciences

Discussion Paper Series

No.1324

Feature Subset Selection for Logistic Regression via Mixed Integer Optimization

by

Toshiki SATO, Yuichi TAKANO, Ryuhei MIYASHIRO, and Akiko YOSHISE

Feb 2015

UNIVERSITY OF TSUKUBA
Tsukuba, Ibaraki 305-8573
JAPAN

Feature Subset Selection for Logistic Regression via Mixed Integer Optimization

Toshiki Sato[a,*], Yuichi Takano[b], Ryuhei Miyashiro[c], Akiko Yoshise[d]

[a] *Doctoral Program in Social Systems and Management, Graduate School of Systems and Information Engineering, University of Tsukuba 1-1-1 Tennodai, Tsukuba-shi, Ibaraki 305-8573, JAPAN*
[b] *School of Network and Information, Senshu University 2-1-1 Higashimita, Tama-ku, Kawasaki-shi, Kanagawa 214-8580, JAPAN*
[c] *Department of Computer and Information Sciences, Institute of Engineering, Tokyo University of Agriculture and Technology 2-24-16 Naka-cho, Koganei-shi, Tokyo 184-8588, JAPAN*
[d] *Division of Policy and Planning Sciences, Faculty of Engineering, Information and Systems, University of Tsukuba 1-1-1 Tennodai, Tsukuba-shi, Ibaraki 305-8573, JAPAN*

Abstract

This paper concerns a method of selecting a subset of features for a logistic regression model. Information criteria, such as the Akaike information criterion and Bayesian information criterion, are employed as a goodness-of-fit measure. The feature subset selection problem is formulated as a mixed integer linear optimization problem, which can be solved with standard mathematical optimization software, by using a piecewise linear approximation. Computational experiments show that, in terms of solution quality, the proposed method has superiority over common stepwise methods.

Keywords: Logistic regression, Feature subset selection, Mixed integer optimization, Information criterion, Piecewise linear approximation

1. Introduction

Binary classification aims to develop a model for separating two classes of samples that are characterized by numerical features. Some examples of

*Corresponding author
Email address: tsato@sk.tsukuba.ac.jp (Toshiki Sato)

binary classification problems are prediction of corporate bankruptcy [3], cancer diagnosis [14], and e-mail spam filtering [11]. There are several statistical learning methods for binary classification, e.g., classic discriminant analysis, logistic regression, and support vector machine (see, e.g., [20]). Among them, this paper focuses on logistic regression and its feature subset selection problem.

The feature subset selection problem is one of choosing a set of significant features from many candidates for model construction. The most commonly used algorithm is the stepwise method [12], which consists of forward selection (adding one feature) and backward elimination (eliminating one feature). To evaluate a subset model of logistic regression, goodness-of-fit (GOF) measures, such as Mallows' C_p [28] and Akaike information criterion (AIC) [1], are frequently employed (see, e.g., [17]). However, several shortcomings of stepwise methods have been pointed out (see, e.g., [19]), and consequently, many alternative methods have been proposed (see, e.g., [8, 15, 22, 26]).

Among them, metaheuristics (see, e.g., [38]), for instance, tabu search [33] and particle swarm optimization [36] can be used for feature subset selection. L_1-regularized logistic regression [21, 25] can also be used to select feature subsets. These algorithms perform well even on large-scale problems; however, they do not necessarily find a best subset of features under certain GOF measures.

The purpose of binary classification is twofold, i.e., prediction and description (see, e.g., [18]). Feature subset selection is known to be beneficial for prediction purposes because it can improve the predictive performance of a statistical model by eliminating irrelevant features for prediction. For description purposes, on the other hand, statistical models are used to understand the cause-and-effect relationships between the selected features and the response class. A better subset of features clearly leads to more reliable results for description purposes; accordingly, the best subset of features is required even if the computation takes a significant amount of time.

Branch and bound algorithms [10, 31, 32, 35, 37] are capable of computing the best subset of features according to the criterion functions used in these studies, such as the Bhattacharyya distance and divergence. These algorithms make an efficient computation by assuming that a subset of features should be not better than any larger sets containing it. This monotonicity assumption, however, is not satisfied by the commonly used GOF measures, e.g., Mallows' C_p and AIC.

In view of these facts, we take a mixed integer optimization (MIO) ap-

proach to feature subset selection. This approach was proposed in the 1970s (see [2]), and it has recently received renewed attention as a result of algorithmic advances and hardware improvements. Indeed, the effectiveness of MIO formulations has been verified, e.g., in [7, 23, 24, 29, 30], in the context of feature subset selection for linear regression. To the best of our knowledge, nevertheless, none of the existing studies have considered MIO formulations for feature subset selection in the logistic regression model.

The purpose of the present paper is to devise MIO formulations for feature subset selection in logistic regression. The problem is first formulated as a mixed integer nonlinear optimization (MINLO) problem. Next, it is converted into a mixed integer linear optimization (MILO) problem, which can be solved with standard MIO software, by making a piecewise linear approximation of the logistic loss function. The greatest advantage of our MILO formulation is its ability to provide an optimality guarantee of the selected features on the basis of information criteria.

The effectiveness of our formulation is assessed through computational experiments on various datasets from the UCI Machine Learning Repository [4]. The computational results demonstrate that when the number of candidate features was less than 40, our method successfully provided a feature subset that was sufficiently close to an optimal one in a reasonable amount of time. Furthermore, even if there were more candidate features, our method often found a better subset of features than the stepwise methods did in terms of information criteria.

2. Feature Subset Selection for Logistic Regression

This section gives a brief review of logistic regression and poses the feature subset selection problem for it.

2.1. Logistic regression model

Let us suppose that we are given n samples of the pairs, $(\boldsymbol{x}_i, y_i)$ for $i = 1, 2, \ldots, n$. Here, $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^\top$ is a p-dimensional feature vector, and $y_i \in \{-1, 1\}$ is a binary class label for each sample $i = 1, 2, \ldots, n$. In the logistic regression for binary classification, the occurrence probability of the class label $y = 1$ is modeled with a sigmoid function (see, e.g., [16, 17, 20]),

$$\Pr(y = 1 \mid \boldsymbol{x}) = \frac{1}{1 + \exp(-(\boldsymbol{w}^\top \boldsymbol{x} + b))},$$

where the intercept, b, and the p-dimensional coefficient vector, $\boldsymbol{w} = (w_1, w_2, \ldots, w_p)^\top$, are parameters to be estimated.

By simple calculation, we find that

$$\Pr(y = -1 \mid \boldsymbol{x}) = 1 - \Pr(y = 1 \mid \boldsymbol{x}) = \frac{1}{1 + \exp(\boldsymbol{w}^\top \boldsymbol{x} + b)}.$$

Therefore, the logistic regression model for both $y \in \{-1, 1\}$ can be expressed as follows:

$$\Pr(y \mid \boldsymbol{x}) = \frac{1}{1 + \exp(-y(\boldsymbol{w}^\top \boldsymbol{x} + b))}. \tag{1}$$

The maximum likelihood estimation method estimates the parameters so that the log likelihood function, $\ell(b, \boldsymbol{w})$, is maximized:

$$\begin{aligned}
\ell(b, \boldsymbol{w}) &= \log \left(\prod_{i=1}^{n} \Pr(y_i \mid \boldsymbol{x}_i) \right) \\
&= -\sum_{i=1}^{n} \log \left(1 + \exp(-y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b))\right) \\
&= -\sum_{i=1}^{n} f(y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)),
\end{aligned} \tag{2}$$

where

$$f(v) = \log(1 + \exp(-v)) \tag{3}$$

is called the logistic loss function. This function is convex because its second derivative always has a positive value. Hence, maximizing the log likelihood function (2) is a convex optimization that can be performed by executing standard nonlinear optimization algorithms, such as the Newton-Raphson algorithm (see, e.g., [16]).

2.2. Feature subset selection problem

This paper selects a subset $S \subseteq \{1, 2, \ldots, p\}$ of the candidate features according to the information criteria (see, e.g., [9]).

Eliminating the j-th feature from the logistic regression model (1) is equivalent to setting the corresponding coefficient, w_j, to zero. As a result,

we will try to minimize the weighted sum of the maximum log likelihood and the number of parameters,

$$\mathrm{IC}(S) = -2\max\{\ell(b, \boldsymbol{w}) \mid w_j = 0 \ (j \notin S)\} + F\,(|S| + 1), \tag{4}$$

where F is a preset dimensionality penalty [13]. For instance, $F = 2$ and $F = \log(n)$ correspond to the Akaike information criterion (AIC, [1]) and Bayesian information criterion (BIC, [34]), respectively.

The feature subset selection problem for the logistic regression model (1) is framed as a combinatorial optimization problem,

$$\mathrm{IC}_{\mathrm{opt}} = \min\{\mathrm{IC}(S) \mid S \subseteq \{1, 2, \ldots, p\}\}. \tag{5}$$

3. Mixed Integer Optimization Formulations

This section presents mixed integer optimization (MIO) formulations for feature subset selection.

3.1. Mixed integer nonlinear optimization formulation

Let $\boldsymbol{z} = (z_1, z_2, \ldots, z_p)^\top$ be a vector of 0-1 decision variables; $z_j = 1$ if $j \in S$; $z_j = 0$, otherwise. The feature subset selection problem (5) can be formulated as a mixed integer nonlinear optimization (MINLO) problem,

$$\begin{aligned}
\underset{b,\,\boldsymbol{w},\,\boldsymbol{z}}{\text{minimize}} \quad & 2\sum_{i=1}^{n} f\left(y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)\right) + F\left(\sum_{j=1}^{p} z_j + 1\right) && (6)\\
\text{subject to} \quad & z_j = 0 \ \Rightarrow \ w_j = 0 \quad (j = 1, 2, \ldots, p), && (7)\\
& z_j \in \{0, 1\} \quad (j = 1, 2, \ldots, p). && (8)
\end{aligned}$$

The logical implications (7) can be represented by a special ordered set type one (SOS1) constraint [5, 6]. This constraint implies that not more than one element in the set can have a nonzero value, and it is supported by standard MIO software. By using the SOS1 constraint, the logical implications (7) can be rewritten as follows:

$$\{1 - z_j, w_j\} = \mathrm{SOS1} \quad (j = 1, 2, \ldots, p). \tag{9}$$

If $z_j = 0$, then $1 - z_j$ has nonzero value, and w_j must be zero from the SOS1 constraints (9).

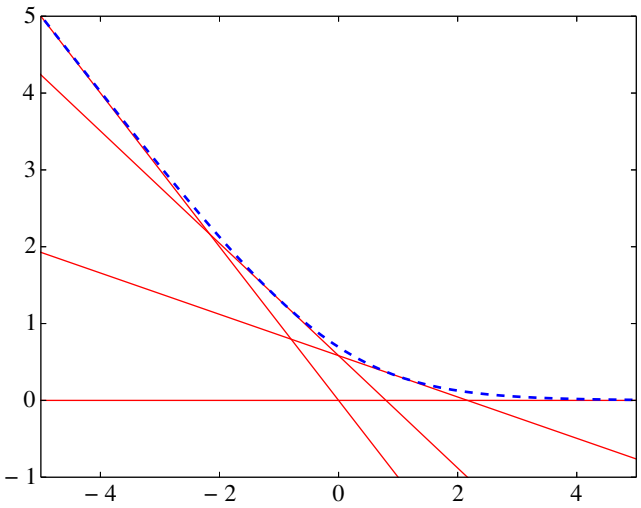


Figure 1: Piecewise linear approximation of the logistic loss function

3.2. Piecewise linear approximation

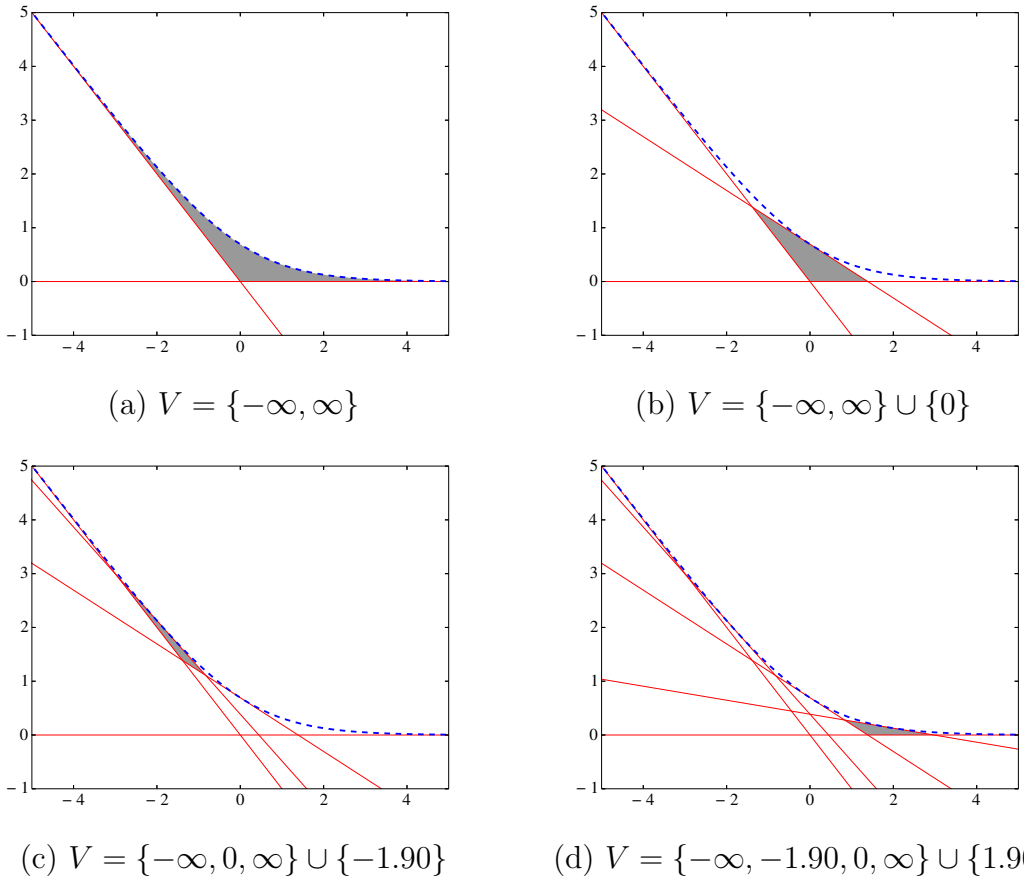
The objective function (6) to be minimized is a convex but nonlinear function, which may cause numerical instabilities in the computation. In addition, most MIO software cannot handle such a nonlinear objective function. In view of these facts, we shall make a piecewise linear approximation of the logistic loss function (3).

Let $V = \{v_1, v_2, \ldots, v_m\}$ be a set of m discrete points. Since the graph of a convex function lies above its tangent lines, the logistic loss function (3) can be approximated by the pointwise maximum of a family of tangent lines; that is,

$$\begin{aligned} f(v) &\approx \max\{f'(v_k)(v - v_k) + f(v_k) \mid k = 1, 2, \ldots, m\} \\ &= \min\{t \mid t \geq f'(v_k)(v - v_k) + f(v_k) \quad (k = 1, 2, \ldots, m)\}. \end{aligned}$$

Figure 1 shows the graph of the logistic loss function (3) together with the tangent lines at $v_1 = -\infty, v_2 = -1, v_3 = 1,$ and $v_4 = \infty$. Also note that

$$f'(v_1)(v - v_1) + f(v_1) = -v, \tag{10}$$
$$f'(v_4)(v - v_4) + f(v_4) = 0. \tag{11}$$



(a) $V = \{-\infty, \infty\}$ (b) $V = \{-\infty, \infty\} \cup \{0\}$

(c) $V = \{-\infty, 0, \infty\} \cup \{-1.90\}$ (d) $V = \{-\infty, -1.90, 0, \infty\} \cup \{1.90\}$

Figure 2: Process of the greedy algorithm for selecting tangent lines

The pointwise maximum of the four tangent lines creates a piecewise linear underestimator to the logistic loss function (3) (see Figure 1).

3.3. Greedy algorithm for selecting tangent lines

It is crucial to select a limited number of tangent lines that provide a good piecewise linear approximation. To accomplish this, we develop a greedy algorithm for selecting a set of tangent lines.

We begin with the two tangent lines (10) and (11) as in Figure 2 (a), where the shaded portion represents the gap between the logistic loss function (3) and its piecewise linear approximation. Our greedy algorithm adds tangent lines one by one so that the area of the shaded portion in Figure 2 (a) will

be minimized. The process of this algorithm is shown in Figures 2 (b)–(d). Here, since the new tangent lines cut off the shaded triangles, the algorithm amounts to sequentially selecting a tangent line that cuts off the biggest triangle. The area of a triangle can be easily calculated from the coordinates of its three vertices.

3.4. Mixed integer linear optimization formulation

Let $\boldsymbol{t} = (t_1, t_2, \ldots, t_n)^\top$ be a vector of auxiliary decision variables for calculating the value of the logistic loss function for each sample. By making a piecewise linear approximation of the logistic loss function (3), the problem (6)–(8) reduces to a mixed integer linear optimization (MILO) problem,

$$\begin{aligned}
\underset{b,\,\boldsymbol{t},\,\boldsymbol{w},\,\boldsymbol{z}}{\text{minimize}} \quad & 2\sum_{i=1}^{n} t_i + F\left(\sum_{j=1}^{p} z_j + 1\right) && (12)\\
\text{subject to} \quad & t_i \geq f'(v_k)\left(y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) - v_k\right) + f(v_k) \\
& \qquad (i = 1, 2, \ldots, n; k = 1, 2, \ldots, m), && (13)\\
& z_j = 0 \;\Rightarrow\; w_j = 0 \quad (j = 1, 2, \ldots, p), && (14)\\
& z_j \in \{0, 1\} \quad (j = 1, 2, \ldots, p). && (15)
\end{aligned}$$

It is clear that the MILO problem (12)–(15) approaches the original MINLO problem (6)–(8) by increasing the number of tangent lines at appropriate points. The MILO problem (12)–(15) has the advantage of being able to give an optimality guarantee to the obtained solution.

Let $\mathrm{obj}^*_{\mathrm{MILO}}$ be the optimal value of the objective function (12). Since it is an underestimator to the original objective function (4), $\mathrm{obj}^*_{\mathrm{MILO}}$ is less than $\mathrm{IC}_{\mathrm{opt}}$. We denote an optimal solution to the MILO problem (12)–(15) by $(b^*, \boldsymbol{t}^*, \boldsymbol{w}^*, \boldsymbol{z}^*)$. Although the associated feature subset $S^* = \{j \mid j = 1, 2, \ldots, p,\ z_j^* = 1\}$ is not necessarily an optimal solution to problem (5), it is possible to make a posteriori accuracy evaluation as follows:

$$\mathrm{obj}^*_{\mathrm{MILO}} \leq \mathrm{IC}_{\mathrm{opt}} \leq \mathrm{IC}(S^*). \tag{16}$$

In other words, if

$$\mathrm{IC}(S^*) - \mathrm{obj}^*_{\mathrm{MILO}}$$

is small, it is guaranteed that the feature subset S^* is sufficiently close to an optimal one in the sense that $\mathrm{IC}(S^*)$ is nearly equal to $\mathrm{IC}_{\mathrm{opt}}$.

4. Computational Results

This section assesses the computational performance of our MILO approach.

We downloaded 12 datasets for the classification analysis from the UCI Machine Learning Repository [4]. The datasets for multi-class classification were converted into instances of binary classification by giving a class label $y_i = 1$ to the samples belonging to the largest class, and by giving $y_i = -1$ to all other samples. Table 1 lists these instances, where n and p are the number of samples and number of candidate features, respectively.

Table 1: List of instances

abbreviation	n	p	original dataset [4]
`Mammo`	830	18	Mammographic Mass
`Image`	2310	18	Image Segmentation
`Parkin`	195	22	Parkinsons
`Stat-H`	270	25	Statlog (Heart)
`Breast`	194	33	Breast Cancer Wisconsin (Prognostic)
`Biodeg`	1055	41	QSAR biodegradation
`SPECTF`	267	44	SPECTF Heart
`Spam`	4610	58	Spambase
`Stat-G`	1000	61	Statlog (German Credit Data)
`Digits`	3823	62	Optical Recognition of Handwritten Digits
`Flags`	194	67	Flags
`Libras`	360	90	Libras Movement

For all the instances, each integer and real variable was standardized so that its mean was zero and its standard deviation was one. Each categorical variable was transformed into dummy variable(s). Samples including missing values and redundant variables having the same value in all samples were eliminated.

The computational experiments compared the performances of the following methods:

- $\mathrm{SW}_{\mathrm{const}}$: stepwise method starting with $S = \emptyset$,
- $\mathrm{SW}_{\mathrm{all}}$: stepwise method starting with $S = \{1, 2, \ldots, p\}$,

- MILO(V): MILO formulation (12)–(15), where V is the set of points for tangent lines.

Both stepwise methods iteratively add or eliminate one feature that leads to the largest decrease in the information criterion. The MILO formulation (12)–(15) employed three sets of points for tangent lines,

$$V_1 = \{0, \pm 1.90, \pm\infty\} \quad (|V_1| = 5, \text{ see also Figure 2 (d)}),$$
$$V_2 = \{0, \pm 0.89, \pm 1.90, \pm 3.55, \pm\infty\} \quad (|V_2| = 9),$$
$$V_3 = \{0, \pm 0.44, \pm 0.89, \pm 1.37, \pm 1.90, \pm 2.63, \pm 3.55, \pm 5.16, \pm\infty\} \quad (|V_3| = 17).$$

These were computed by the greedy algorithm described in Section 3.3. As shown above, the greedy algorithm yielded a symmetric set of points.

All computations were performed on a Windows computer with an Intel Core i7-2600 CPU (3.40 GHz) and 12 GB memory. Gurobi Optimizer 6.0.0 (`http://www.gurobi.com`) was used to solve the MILO problems. Here, the logical implications (14) were imposed in the form of SOS1 constraints (9) with the `SOS_TYPE1` function in Gurobi Optimizer. The stepwise methods were performed with the `step` function implemented in R 3.1.2 (`http://www.R-project.org`).

Tables 2–5 show the computational results of minimizing AIC and BIC. The column labeled "IC(S)" is the value of the information criterion (4), where S is the set of features selected by each method. Note that the smallest values for each instance are bold-faced. The column labeled "obj_{MILO}" is the value of the objective function (12). The column labeled "$|S|$" is the number of selected features, and the column labeled "time (s)" is computation time in seconds. The MILO computation was terminated if it did not finish by itself after 10000 seconds. In this case, the solution obtained within 10000 seconds is not necessarily an optimal one to the problem (12)–(15), and accordingly, obj_{MILO} may be greater than IC_{opt}.

Tables 2–5 reveal that the stepwise methods, SW_{const} and SW_{all}, frequently arrived at different feature subsets for the same instance. In particular, for `Breast`, SW_{const} and SW_{all} respectively selected 12 and 24 features for minimizing AIC in Table 2, and more surprisingly, 2 and 13 features for minimizing BIC in Table 4. It is clear that SW_{const} or SW_{all} failed to find the best subset of features in terms of the information criteria. This is one of the fundamental shortcomings of the stepwise methods; however, it is noteworthy that they completed the search process within 1000 seconds for all the instances.

Let us next discuss the cases where the MILO computation finished within 10000 seconds. In this case, the relationship (16) holds. Tables 2–5 confirmed that many tangent lines narrowed the gap between the upper bound, IC(S) and the lower bound, $\mathrm{obj}_{\mathrm{MILO}}$. In the case of `Stat-H` in Table 2, IC(S) and $\mathrm{obj}_{\mathrm{MILO}}$ of MILO(V_1) were 196.17 and 182.39, and those of MILO(V_3) were 195.78 and 194.90. In other words, MILO(V_3) found a feature subset such that the associated AIC value was 195.78, and it guaranteed that the smallest AIC value was greater than 194.90. This optimality guarantee is the most notable characteristic of our MILO formulation, and it is not shared by a number of heuristic approaches.

The computation time of solving the MILO problems increased as the number of tangent lines grew. This is because the number of the constraints (13) depends on the number of tangent lines. Therefore, it is essential to select a limited number of tangent lines, and our greedy algorithm is useful for accomplishing this objective.

MILO(V_3) always attained the smallest AIC value among the five methods when its computation finished within 10000 seconds. This suggests that the number of tangent lines of V_3 is sufficiently large. MILO(V_2) failed to find a best feature subset for `Stat-H` in Table 4. MILO(V_1) often failed to find a best feature subset even though its computation finished within 10000 seconds.

We next turn to the cases where the MILO computation was terminated due to the time limit. In this case, the feature subsets provided by the stepwise methods were sometimes better than those of the MILO formulations. For instance, both $\mathrm{SW}_{\mathrm{const}}$ and $\mathrm{SW}_{\mathrm{all}}$ found better feature subsets than those of the MILO formulations for `Digits` in Table 5. However, if viewed from the opposite perspective, the MILO formulations successfully yielded good-quality solutions to most of the instances in spite of the time limit. In particular, the MILO formulations significantly outperformed the stepwise methods for `Libras` in Table 5. More specifically, the BIC values of the MILO formulations were 41.20, 47.09 and 47.09, whereas those of the stepwise methods were 82.05 and 52.98. If it is allowed to spend a long time on the computation, our MILO approach will be a reasonable option for subset selection from many candidate features.

5. Conclusion

This paper dealt with the feature subset selection problem for logistic regression. It is easy to frame this problem as a mixed integer nonlinear

optimization (MINLO) problem, but the MINLO problem is hard to handle. Thus, we formulated its approximation as a mixed integer linear optimization (MILO) problem by applying a piecewise linearization to the nonlinear logistic loss functions.

Our research contribution is the computational framework for selecting a subset of features with an optimality guarantee. This is the crucial difference between our method and a number of heuristic approaches. Moreover, raising the number of tangent lines leads to greater accuracy in the piecewise linear approximation. In this sense, our MILO approach is close to the exact method of computing the best subset of features.

It was demonstrated that our method frequently outperformed the stepwise methods in terms of solution quality. This fact proves the effectiveness of the mixed integer optimization methodology in feature subset selection, and hence, it should be of use in other statistical analyses as well.

Many heuristic approaches represented by the stepwise methods can quickly reach a good-quality solution in exchange for optimality even if there are a large number of candidate features. By contrast, our MILO formulation spends a long time searching for a solution with an optimality guarantee. For practical purposes, it is necessary to use both approaches as the situation demands.

To improve tractability, we reduced the MINLO problem to the MILO problem. However, the MINLO problem may be solved directly by using a tailored branch-and-bound algorithm, because it becomes a convex optimization problem if the integrality of the problem is relaxed. Another direction of future research will be to extend our MILO formulation to multi-class classification. Specifically, we will tackle a feature subset selection problem in discrete choice models, such as the logit model [27]. We also need to consider selecting a subset of features on the basis of goodness-of-fit measures other than AIC and BIC.

References

[1] Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control, 19,* 716–723. doi: 10.1109/TAC.1974.1100705

[2] Arthanari, T. S., & Dodge, Y. (1981). *Mathematical programming in statistics.* New York, NY: Wiley.

[3] Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance, 23,* 589–609. doi: 10.1111/j.1540-6261.1968.tb00843.x

[4] Bache, K. & Lichman, M. (2013). UCI Machine Learning Repository [`http://archive.ics.uci.edu/ml`]. Irvine, CA: University of California, School of Information and Computer Science.

[5] Beale, E. M. L. (1963). Two transportation problems. In Kreweras, G. & Morlat, G. (Eds.), *Proceedings of the Third International Conference on Operational Research,* 780–788.

[6] Beale, E. M. L., & Tomlin, J. A. (1970). Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables. In Lawrence, J. (Ed.), *Proceedings of the Fifth International Conference on Operational Research* (pp. 447–454). London, UK: Tavistock Publications.

[7] Bertsimas, D., & Shioda, R. (2009). Algorithm for cardinality-constrained quadratic optimization. *Computational Optimization and Applications, 43,* 1–22. doi: 10.1007/s10589-007-9126-9

[8] Blum, A. L., & Langley, P. (1997). Selection of relevant features and examples in machine learning. *Artificial Intelligence, 97,* 245–271. doi: 10.1016/S0004-3702(97)00063-5

[9] Burnham, K. P., & Anderson, D. R. (2002). *Model selection and multimodel inference: A practical information-theoretic approach* (2nd edition). New York, NY: Springer-Verlag. doi: 10.1007/b97636

[10] Chen, X. (2003). An improved branch and bound algorithm for feature selection. *Pattern Recognition Letters, 24,* 1925–1933. doi: 10.1016/S0167-8655(03)00020-5

[11] Cormack, G. V. (2007). Email spam filtering: A systematic review. *Foundations and Trends in Information Retrieval, 1,* 335–455. doi: 10.1561/1500000006

[12] Efroymson, M. A. (1960). Multiple regression analysis. In Ralston A., & Wilf H. S. (Eds.), *Mathematical methods for digital computers* (pp. 191–203). New York, NY: Wiley.

[13] George, E. I. (2000). The variable selection problem. *Journal of the American Statistical Association, 95,* 1304–1308. doi: 10.1080/01621459.2000.10474336

[14] Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning, 46,* 389–422. doi: 10.1023/A:1012487302797

[15] Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *The Journal of Machine Learning Research*, 3, 1157–1182. Retrieved from `http://www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf`

[16] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning* (2nd edition). New York, NY: Springer-Verlag. doi: 10.1007/978-0-387-84858-7

[17] Hosmer, D. W. Jr., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (3rd edition). Hoboken, NJ: Wiley.

[18] Huberty, C. J. (1984). Issues in the use and interpretation of discriminant analysis. *Psychological Bulletin, 95,* 156–171. Retrieved from `http://psycnet.apa.org/journals/bul/95/1/156/`

[19] Huberty, C. J. (1989). Problems with stepwise methods — better alternatives. *Advances in Social Science Methodology, 1,* 43–70. Retrieved from `http://coeweb.gsu.edu/coshima/EPRS8550/Oshima%20Problem.pdf`

[20] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning.* New York, NY: Springer-Verlag.

[21] Koh, K., Kim, S., & Boyd, S. (2007). An interior-point method for large-scale ℓ_1-regularized logistic regression. *Journal of Machine learning research, 8,* 1519–1555. Retrieved from `http://jmlr.org/papers/volume8/koh07a/koh07a.pdf`

[22] Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence, 97,* 273–324. doi: 10.1016/S0004-3702(97)00043-X

[23] Konno, H., & Takaya, Y. (2010). Multi-step methods for choosing the best set of variables in regression analysis. *Computational Optimization and Applications, 46,* 417–426. doi: 10.1007/s10589-008-9193-6

[24] Konno, H., & Yamamoto, R. (2009). Choosing the best set of variables in regression analysis using integer programming. *Journal of Global Optimization, 44,* 273–282. doi: 10.1007/s10898-008-9323-9

[25] Lee, S., Lee, H., Abbeel, P., & Ng, A. Y. (2006). Efficient L_1 regularized logistic regression. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence* (pp. 401–408). Menlo Park, CA: AAAI Press.

[26] Liu, H., & Motoda, H. (Eds.). (2007). *Computational methods of feature selection.* Boca Raton, FL: Chapman & Hall/CRC.

[27] McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In Zarembka, P. (Ed.), *Frontiers in Econometrics* (pp. 105–142). New York, NY: Academic Press.

[28] Mallows, C. L. (1973). Some comments on C_p. *Technometrics*, 15, 661–675. doi: 10.1080/00401706.1973.10489103

[29] Miyashiro, R., & Takano, Y. (2013). Mixed integer second-order cone programming formulations for variable selection (Technical Report, No. 2013-7). Retrieved from Tokyo Institute of Technology, Department of Industrial Engineering and Management website: `http://www.me.titech.ac.jp/technicalreport/h25/2013-7.pdf`

[30] Miyashiro, R., & Takano, Y. (2015). Subset selection by Mallows' C_p: A mixed integer programming approach. *Expert Systems with Applications, 42,* 325–331. doi: 10.1016/j.eswa.2014.07.056

[31] Nakariyakul, S., & Casasent, D. P. (2007). Adaptive branch and bound algorithm for selecting optimal features. *Pattern Recognition Letters, 28,* 1415–1427. doi: 10.1016/j.patrec.2007.02.015

[32] Narendra, P. M., & Fukunaga, K. (1977). A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers, C-26,* 917–922. doi: 10.1109/TC.1977.1674939

[33] Pacheco, J., Casado, S., & Núñez, L. (2009). A variable selection method based on tabu search for logistic regression models. *European Journal of Operational Research, 199,* 506–511. doi: 10.1016/j.ejor.2008.10.007

[34] Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics, 6,* 461–464. Retrieved from `http://projecteuclid.org/euclid.aos/1176344136`

[35] Somol, P., Pudil, P., & Kittler, J. (2004). Fast branch & bound algorithms for optimal feature selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26,* 900–912. doi: 10.1109/TPAMI.2004.28

[36] Unler, A., & Murat, A. (2010). A discrete particle swarm optimization method for feature selection in binary classification problems. *European Journal of Operational Research, 206,* 528–539. doi: 10.1016/j.ejor.2010.02.032

[37] Yu, B., & Yuan, B. (1993). A more efficient branch and bound algorithm for feature selection. *Pattern Recognition, 26,* 883–889. doi: 10.1016/0031-3203(93)90054-Z

[38] Yusta, S. C. (2009). Different metaheuristic strategies to solve the feature selection problem. *Pattern Recognition Letters, 30,* 525–534. doi: 10.1016/j.patrec.2008.11.012

Table 2: Results of minimizing AIC ($p \leq 41$)

instance	n	p	method	IC(S)	obj_{MILO}	$\lvert S\rvert$	time (s)
`Mammo`	830	18	SW_{const}	**628.10**	—	8	2.09
			SW_{all}	632.64	—	11	2.89
			MILO(V_1)	628.51	587.69	9	3.47
			MILO(V_2)	**628.10**	616.37	8	7.29
			MILO(V_3)	**628.10**	625.27	8	24.62
`Image`	2310	18	SW_{const}	438.39	—	10	6.72
			SW_{all}	433.32	—	12	6.63
			MILO(V_1)	433.32	403.74	12	131.20
			MILO(V_2)	**433.13**	424.23	12	326.81
			MILO(V_3)	**433.13**	430.80	12	1218.67
`Parkin`	195	22	SW_{const}	123.62	—	5	0.33
			SW_{all}	113.96	—	8	1.33
			MILO(V_1)	**113.50**	106.07	7	4.12
			MILO(V_2)	**113.50**	111.45	7	15.33
			MILO(V_3)	**113.50**	112.93	7	30.79
`Stat-H`	270	25	SW_{const}	197.74	—	12	0.91
			SW_{all}	199.24	—	13	1.31
			MILO(V_1)	196.17	182.39	10	6.81
			MILO(V_2)	**195.78**	192.13	11	71.64
			MILO(V_3)	**195.78**	194.90	11	138.49
`Breast`	194	33	SW_{const}	162.94	—	12	1.06
			SW_{all}	152.13	—	24	1.49
			MILO(V_1)	**147.04**	137.96	18	22.88
			MILO(V_2)	**147.04**	144.56	18	57.72
			MILO(V_3)	**147.04**	146.41	18	240.39
`Biodeg`	1055	41	SW_{const}	657.50	—	20	19.80
			SW_{all}	**653.29**	—	22	40.20
			MILO(V_1)	**653.29**	613.26	22	1441.26
			MILO(V_2)	**653.29**	641.32	22	3863.64
			MILO(V_3)	**653.29**	650.18	22	>10000

Table 3: Results of minimizing AIC ($p \geq 44$)

instance	n	p	method	IC(S)	obj_{MILO}	$\lvert S\rvert$	time (s)
SPECTF	267	44	SW_{const}	172.34	—	9	1.28
			SW_{all}	169.42	—	16	8.89
			MILO(V_1)	168.93	158.93	14	490.06
			MILO(V_2)	**168.33**	165.58	14	1197.12
			MILO(V_3)	**168.33**	167.58	14	5470.24
Spam	4610	58	SW_{const}	**1912.88**	—	43	339.62
			SW_{all}	**1912.88**	—	43	246.73
			MILO(V_1)	1914.08	1783.66	41	7827.50
			MILO(V_2)	**1912.88**	1875.71	43	>10000
			MILO(V_3)	**1912.88**	1903.11	43	>10000
Stat-G	1000	61	SW_{const}	958.15	—	23	12.34
			SW_{all}	966.57	—	26	48.64
			MILO(V_1)	**957.92**	904.89	24	>10000
			MILO(V_2)	960.08	946.01	23	>10000
			MILO(V_3)	958.59	955.12	23	>10000
Digits	3823	62	SW_{const}	327.34	—	24	180.33
			SW_{all}	325.74	—	25	769.29
			MILO(V_1)	**325.43**	296.80	25	>10000
			MILO(V_2)	326.71	317.90	26	>10000
			MILO(V_3)	329.65	326.84	26	>10000
Flags	194	67	SW_{const}	42.00	—	20	6.56
			SW_{all}	48.00	—	23	57.69
			MILO(V_1)	**40.00**	40.00	19	>10000
			MILO(V_2)	**40.00**	40.00	19	>10000
			MILO(V_3)	42.00	42.00	20	>10000
Libras	360	90	SW_{const}	22.00	—	10	6.53
			SW_{all}	18.00	—	8	323.22
			MILO(V_1)	**16.00**	14.00	6	>10000
			MILO(V_2)	**16.00**	16.00	7	>10000
			MILO(V_3)	**16.00**	16.00	7	>10000

Table 4: Results of minimizing BIC ($p \leq 41$)

instance	n	p	method	IC(S)	obj_{MILO}	$\lvert S\rvert$	time (s)
`Mammo`	830	18	SW_{const}	**657.96**	—	4	0.49
			SW_{all}	668.83	—	6	3.79
			MILO(V_1)	**657.96**	617.12	4	2.49
			MILO(V_2)	**657.96**	646.06	4	5.73
			MILO(V_3)	**657.96**	655.21	4	20.47
`Image`	2310	18	SW_{const}	496.59	—	9	5.51
			SW_{all}	500.44	—	9	8.00
			MILO(V_1)	497.45	462.02	7	60.90
			MILO(V_2)	**495.51**	486.75	8	446.78
			MILO(V_3)	**495.51**	493.11	8	1109.00
`Parkin`	195	22	SW_{const}	140.70	—	1	0.09
			SW_{all}	140.86	—	5	1.47
			MILO(V_1)	**137.60**	129.26	5	5.26
			MILO(V_2)	**137.60**	135.64	5	13.92
			MILO(V_3)	**137.60**	137.06	5	43.64
`Stat-H`	270	25	SW_{const}	229.50	—	4	0.27
			SW_{all}	239.60	—	8	1.64
			MILO(V_1)	226.77	210.44	5	4.89
			MILO(V_2)	226.77	222.31	5	47.03
			MILO(V_3)	**226.29**	225.37	6	119.29
`Breast`	194	33	SW_{const}	195.17	—	2	0.17
			SW_{all}	200.97	—	13	2.72
			MILO(V_1)	194.22	182.11	3	135.41
			MILO(V_2)	**192.36**	189.62	9	433.64
			MILO(V_3)	**192.36**	191.69	9	1810.77
`Biodeg`	1055	41	SW_{const}	744.25	—	11	9.83
			SW_{all}	748.67	—	14	48.36
			MILO(V_1)	**744.22**	701.77	14	>10000
			MILO(V_2)	744.40	732.65	10	>10000
			MILO(V_3)	744.25	741.01	11	>10000

Table 5: Results of minimizing BIC ($p \geq 44$)

instance	n	p	method	IC(S)	obj_{MILO}	$\lvert S\rvert$	time (s)
SPECTF	267	44	SW_{const}	**196.82**	—	4	0.54
			SW_{all}	**196.82**	—	4	10.81
			MILO(V_1)	**196.82**	186.62	4	93.00
			MILO(V_2)	**196.82**	193.90	4	303.52
			MILO(V_3)	**196.82**	196.04	4	1441.01
Spam	4610	58	SW_{const}	**2154.75**	—	29	406.85
			SW_{all}	**2154.75**	—	29	406.85
			MILO(V_1)	2155.44	2019.14	31	>10000
			MILO(V_2)	2154.77	2113.82	28	>10000
			MILO(V_3)	**2154.75**	2144.35	30	>10000
Stat-G	1000	61	SW_{const}	**1047.32**	—	11	4.56
			SW_{all}	1060.95	—	14	56.13
			MILO(V_1)	**1047.32**	994.54	11	>10000
			MILO(V_2)	1049.03	1034.29	10	>10000
			MILO(V_3)	1051.01	1047.44	11	>10000
Digits	3823	62	SW_{const}	**446.68**	—	14	60.90
			SW_{all}	448.07	—	16	859.33
			MILO(V_1)	449.36	416.41	13	>10000
			MILO(V_2)	454.22	443.48	13	>10000
			MILO(V_3)	468.57	465.13	14	>10000
Flags	194	67	SW_{const}	129.79	—	5	0.84
			SW_{all}	126.42	—	23	58.63
			MILO(V_1)	129.87	112.08	4	>10000
			MILO(V_2)	**105.50**	105.19	18	>10000
			MILO(V_3)	121.83	121.45	12	>10000
Libras	360	90	SW_{const}	82.05	—	5	2.59
			SW_{all}	52.98	—	8	323.98
			MILO(V_1)	**41.20**	41.20	6	>10000
			MILO(V_2)	47.09	47.09	7	>10000
			MILO(V_3)	47.09	47.09	7	>10000