Journal of Applied Mathematics and Physics, 2021, 9, 969-975
https://www.scirp.org/journal/jamp
ISSN Online: 2327-4379
ISSN Print: 2327-4352

Optimization Model and Algorithm for Multi-Label Learning

Zhengyang Li

Fuzhou University, Fuzhou, China
Email: 870724066@qq.com

How to cite this paper: Li, Z.Y. (2021) Optimization Model and Algorithm for Multi-Label Learning. *Journal of Applied Mathematics and Physics*, **9**, 969-975.
https://doi.org/10.4236/jamp.2021.95066

Received: April 22, 2021
Accepted: May 17, 2021
Published: May 20, 2021

Abstract

This paper studies a kind of urban security risk assessment model based on multi-label learning, which is transformed into the solution of linear equations through a series of transformations, and then the solution of linear equations is transformed into an optimization problem. Finally, this paper uses some classical optimization algorithms to solve these optimization problems, the convergence of the algorithm is proved, and the advantages and disadvantages of several optimization methods are compared.

Keywords

Operations Research, Multi-Label Learning, Linear Equations Solving, Optimization Algorithm

1. Introduction

In some traditional classification learning, the same sample is labeled by one category label at most, and this kind of classification learning problem is called single label classification learning problem. In real life, a sample usually corresponds to multiple different category labels [1] [2] [3] [4]. For example, a sofa may have multiple different labels such as "solid wood", "furniture", "sculpture" and so on. A paper may have different labels such as "highly citation papers", "core journals" and "mathematics discipline" [1] [2] [3] [4].

Multi-label learning [5] [6] [7] is a machine learning [8] problem under supervised learning. It constructs a classifier that can automatically select the most relevant label subset from a large number of label sets to label the sample. At present, artificial intelligence has become a hot research field in today's society, and multi-label classification is a hot issue in the field of artificial intelligence, a variety of multi-label classification algorithms emerge.

In recent years, with the rapid development of urbanization in China, there

are also some hidden dangers, especially some major safety accidents caused some economic losses and casualties, which bring panic to people living in the city. Using multi-label learning method to establish an effective urban safety risk assessment system is of great significance to prevent some security incidents and improve the safety of urban residents.

Two methods commonly used in urban safety risk assessment [9] are risk matrix method [10] [11] [12] (referred to as LS method) and operational condition risk assessment method (referred to as LEC method). The risk matrix method is to multiply the possibility of injury (L) and the severity of injury (S), and the results are called risk values, According to the size of the risk value, risk classification is carried out, and then corresponding risk control measures are taken. The possibility of injury (L) is based on the scores of deviation frequency, safety inspection, operation process, employee competency and control measures. The scores of these five aspects are obtained between 1 and 5, and the highest of the five scores is the final L value (*i.e.*, the possibility of injury, hereinafter referred to as L value). Severity of injury (S) According to the scores of casualties, property losses, compliance with laws and regulations, environmental damage and damage to corporate reputation, the scores of these five aspects are also between 1 and 5, and the highest score is taken as the final S value (the severity of injury, hereinafter referred to as S value).

In practice, there are some difficulties in obtaining the possibility of injury (L) and the severity of injury (S). In the case of not accurately obtaining the value of L and S, how to obtain these two values in other ways becomes the problem to be solved in this paper. The method based on multi-label learning is effective to solve such problems.

2. Question

In real life, an object is usually associated with multiple labels. In this case, it is necessary to use multi-label learning [13] [14] [15]. An object in multi-label learning is associated with multiple labels at the same time, while an object in single-label learning is associated with only one label. In recent years, multi-label learning has been widely used in various scenarios, such as bioinformatics, web mining, text classification, image field and so on.

In the urban safety risk assessment, an evaluation object has multiple characteristics at the same time. The evaluation of the safety of this evaluation object is reflected by the possibility of injury (L) and the severity of injury (S). In this case, it is necessary to use multi-label learning [16] [17] [18].

The central idea is to find n features of the problem, we set it as $x_{1i}, x_{2i}, \ldots, x_{ni}$, Order

$$x_i = \begin{pmatrix} x_{1i} \\ x_{2i} \\ \vdots \\ x_{ni} \end{pmatrix},$$

Which x_i represents the evaluation object $i=1,2,...,m$, The possibility of injury (L) is set to b_{1i}, the severity of injury (S) is set to b_{2i}, so that

$$b_i = \begin{pmatrix} b_{1i} \\ b_{2i} \end{pmatrix},$$

where b_i is the evaluation index, including the possibility of injury (L) and the severity of injury (S), $i=1,2,...,m$. Thus, the original problem is transformed into a problem about solving linear equations.

3. Methods

In this paper, the model in practical problems is first transformed into a set of linear equations, and then it is equivalently transformed into a class of optimization problems. Through optimization tools, the numerical solution of linear equations is obtained by gradual approximation. Finally, the advantages and disadvantages of several optimization methods are compared.

At present, the mainstream solution of linear equations generally has two categories, one is the direct solution, and the other is the iterative method [19] [20] [21] [22].

Firstly, This article first finds n features $x_{1i}, x_{2i}, \ldots, x_{ni}$ of the evaluation object x_i. The possibility (L) of injury is set as b_{1i}, and the severity (S) of injury is set as b_{2i}, let

$$x_i = \begin{pmatrix} x_{1i} \\ x_{2i} \\ \vdots \\ x_{ni} \end{pmatrix}, \quad b_i = \begin{pmatrix} b_{1i} \\ b_{2i} \end{pmatrix}, \quad i = 1,2,...,\mathrm{N},$$

Take out m trainings, this set is called training set [23], the remaining set is called test set [23].

The processing method in this paper is to transform it. First of all, this paper lets $X=(x_1,x_2,...,x_m)$, $B=(b_1,b_2,...,b_m)$, and the original equations are $AX=B$. The optimal solution A of the optimization problem [24] [25] [26] [27]

$$\min_A f(A) = \frac{1}{2}\sum_{i=1}^{m} \left\| Ax_i - b_i \right\|_2^2 = \frac{1}{2}\left\| AX - B \right\|_F^2$$

is A in the required linear equation $AX=B$, Where $\left\|AX-B\right\|_F^2$ is a matrix least squares (Frobenius norm).

There are many methods [28] for solving the optimal solution A of optimization problem

$$\min_A f(A) = \frac{1}{2}\sum_{i=1}^{m} \left\| Ax_i - b_i \right\|_2^2 = \frac{1}{2}\left\| AX - B \right\|_F^2,$$

such as Gradient Descend method, Newton method, BFGS method, FR method and so on. Gradient descent method [29] [30]: use the negative gradient direction of the current position as the search direction, because the direction is the

fastest descent direction of the current position, so it is also called the "fastest descent method". The closer the steepest descent method to the target value, the smaller the step size, the slower the forward. There are two varieties, batch gradient descent (BGD) and random gradient descent (SGD). Batch gradient descent method: minimize the loss function of all training samples, so that the final solution is the global optimal solution, that is, the solved parameter is to minimize the risk function, but it is inefficient for large-scale sample problems. Stochastic gradient descent method: to minimize the loss function of each sample, although not every iteration of the loss function is toward the global optimal direction, but the large overall direction is toward the global optimal solution, the final result is often near the global optimal solution, suitable for large-scale training samples. Newton method: a method for approximate solving equations in real and complex fields, with second-order convergence and fast convergence. The disadvantage is that it is an iterative algorithm, and each step needs to solve the inverse matrix of the Hessian matrix of the objective function, so the calculation is complex. Quasi-Newton method: it improves the defect that Newton method needs to solve the inverse matrix of complex Hessian matrix every time. It uses positive definite matrix to approximate the inverse of Hessian matrix, thus simplifying the computational complexity. Conjugate gradient method: a method between steepest descent method and Newton method, it only uses the first derivative information, but overcomes the slow convergence of steepest descent method, and avoids the shortcomings of Newton method that need to store and calculate Hesse matrix and inverse. Conjugate gradient method is not only one of the most useful methods to solve large linear equations, but also one of the most effective algorithms to solve large nonlinear optimization. Among various optimization algorithms, conjugate gradient method is very important. Its advantage is that it requires small storage, has step convergence, high stability, and does not require any external parameters.

4. Algorithm

4.1. Algorithm 4.1: Gradient Descend Method

Step 1 Take $x^1 \in R^n$ as the initial iteration point, precision $\varepsilon \geq 0$, and let $k = 1$.

Step 2 Calculate $d^k = -\nabla f(x^k)$, if $\left\|d^k\right\| \leq \in$, the algorithm terminates, x^k is an approximate stable point.

Step 3 Calculation step size $\alpha_k \geq 0$.

Step 4 Calculation $x^{k+1} = x^k + \alpha_k d^k$, let $k = k+1$, and implementation step 2.

4.2. Algorithm 4.2: BFGS Method

Step 1 Take $x^1 \in R^n$, $B^1 \in R^{n\times n}$ symmetric positive definite, let $k = 1$.

Step 2 If $\nabla f(x^k) = 0$, then x^k is the solution, and the calculation ends.

Step 3 Calculate the search direction d^k,

$$B^k d^k = -\nabla f(x^k).$$

Step 4 Calculation of search step size α_k satisfies

$$f(x^k + \alpha_k d^k) - f(x^k) \le \rho \alpha_k \nabla f(x^k)^T d^k,$$

$$\nabla f(x^k + \alpha_k d^k)^T d^k \ge \sigma \nabla f(x^k)^T d^k, \quad 0 < \rho < \sigma < 1.$$

Step 5 Calculation x^{k+1}, δ^k, y^k, B^{k+1},

$$x^{k+1} = x^k + \alpha_k d^k, \quad \delta^k = \alpha_k d^k, \quad y^k = \nabla f(x^{k+1}) - \nabla f(x^k),$$

$$B^{k+1} = B^k - B^k \delta^k \delta^{kT} B^k / \delta^{kT} B^k \delta^k + y^k y^{kT} / \delta^{kT} y^k.$$

Step 6 $k = k + 1$, turn to step 2.

4.3. Algorithm 4.3: FR Method

Step 1 Take $x^1 \in R^n$, $d^1 = -\nabla f(x^1)$, $0 < \rho \le \sigma < 1/2$, order $k = 1$.

Step 2 If $\nabla f(x^k) = 0$, x^k is the stable point of f, the calculation terminates.

Step 3 Calculation of search step size α_k satisfies

$$f(x^k + \alpha_k d^k) - f(x^k) \le \rho \alpha_k \nabla f(x^k)^T d^k,$$

$$\left|\nabla f(x^k + \alpha_k d^k)^T d^k\right| \le \sigma \left|\nabla f(x^k)^T d^k\right|.$$

Step 4 Calculate the search direction d^k,

$$x^{k+1} = x^k + \alpha_k d^k, \quad \beta_{k+1,k}^{FR} = \left\|\nabla f(x^{k+1})\right\|^2 / \left\|\nabla f(x^k)\right\|^2,$$

$$d^{k+1} = -\nabla f(x^{k+1}) + \beta_{k+1,k}^{FR} d^k.$$

Step 5 $k = k + 1$, Turn to step 2.

5. Experiments and Conclusions

In order to facilitate the research of different $x_i = \begin{pmatrix} x_{1i} \\ x_{2i} \\ \vdots \\ x_{ni} \end{pmatrix}$, $b_i = \begin{pmatrix} b_{1i} \\ b_{2i} \end{pmatrix}$, $i = 1, 2, \ldots, \mathrm{N}$ calculated results, In this paper, x_i, b_i is obtained by taking random numbers, respectively, using Gradient Descend method, BFGS method and FR method to calculate the results in **Table 1**:

Table 1. In matrices of different sizes, these three optimization methods converge to the number of steps of the optimal solution.

	Gradient Descend	BFGS	FR
The first experiment (m = 300, n = 500)	258	2	24
The second experiment (m = 1000, n = 1500)	115	2	19
The third experiment (m = 3000, n = 5000)	80	4	18
The fourth experiment (m = 5000, n = 8000)	72	4	32
The fifth experiment (m = 10,000, n = 12,000)	64	52	32

According to the operation results, we found that the BFGS method had the fastest convergence rate in this group of experiments, followed by the FR method, and the Gradient Descend method had the worst convergence rate.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Weston, J., Makadia, A. and Yee, H. (2013) Label Partitioning for Sublinear Ranking.

[2] Agrawal, R., Gupta, A., Prabhu, Y., *et al.* (2013) Multi-Label Learning with Millions of Labels: Recommending Advertiser Bid Phrases for Web Pages. *Proceedings of the* 22*nd International Conference on World Wide Web*, *ACM*, 13-24. https://doi.org/10.1145/2488388.2488391

[3] Cisse, M.M., Usunier, N., Artieres, T., *et al.* (2013) Robust Bloom Filters for Large Multilabel Classification Tasks. *Advances in Neural Information Processing Systems*, 1851-1859.

[4] Prabhu, Y. and Varma, M. (2014) Fastxml: A Fast, Accurate and Stable Tree-Classifier for Extreme Multi-Label Learning. *Proceedings of the* 20*th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *ACM*, 263-272. https://doi.org/10.1145/2623330.2623651

[5] Jain, H., Prabhu, Y. and Varma, M. (2016) Extreme Multi-Label Loss Functions for Recommendation, Tagging, Ranking & Other Missing Label Applications. *Proceedings of the* 22*nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *ACM*, 935-944. https://doi.org/10.1145/2939672.2939756

[6] Jasinska, K., Dembczynski, K., Busa-Fekete, R., *et al.* () Extreme F-Measure Maximization Using Sparse Probability Estimates. *International Conference on Machine Learning*, 1435-1444.

[7] Si, S., Zhang, H., Keerthi, S.S., *et al.* (2017) Gradient Boosted Decision Trees for High Dimensional Sparse Output. *Proceedings of the* 34*th International Conference on Machine Learning*, **70**, 3182-3190.

[8] Chen, H.H., Huang, B., Liu, F. and Chen, W.G. (2017) Machine Learning Principles and Applications. University of Electronic Science and Technology Press, Chengdu, 2-19.

[9] Song, Y.H. (2007) Urban Security Risk Assessment System Based on Economic Marginal Effect.

[10] Ni, H., Chen, A. and Chen, N. (2010) Some Extensions on Risk Matrix Approach. *Safety Science*, **48**, 1269-1278. https://doi.org/10.1016/j.ssci.2010.04.005

[11] Ruan, X., Yin, Z. and Chen, A. (2013) A Review on Risk Matrix Method and Its Engineering Application. *Journal of Tongji University*, *Natural Science*, **41**, 381-385.

[12] Garvey, P.R. and Lansdowne, Z.F. (1998) Risk Matrix: An Approach for Identifying, Assessing, and Ranking Program Risks. *Air Force Journal of Logistics*, **22**, 18-21.

[13] Hsu, D.J., Kakade, S.M., Langford, J., *et al.* (2009) Multi-Label Prediction via Compressed Sensing. *Advances in Neural Information Processing Systems*, 772-780.

[14] Zhang, Y. and Schneider, J. (2011) Multi-Label Output Codes Using Canonical Correlation Analysis. *Proceedings of the Fourteenth International Conference on*

Artificial Intelligence and Statistics, 873-882.

[15] Tai, F. and Lin, H.T. (2012) Multilabel Classification with Principal Label Space Transformation. *Neural Computation*, **24**, 2508-2542. https://doi.org/10.1162/NECO_a_00320

[16] Balasubramanian, K. and Lebanon, G. (2012) The Landmark Selection Method for Multiple Output Prediction.

[17] Chen, Y.N. and Lin, H.T. (2012) Feature-Aware Label Space Dimension Reduction for Multi-Label Classification. *Advances in Neural Information Processing Systems*, 1529-1537.

[18] Yu, H.F., Jain, P., Kar, P., *et al.* (2014) Large-Scale Multi-Label Learning with Missing Labels. *International Conference on Machine Learning*, 593-601.

[19] Vinsome, P.K.W. (1976) Orthomin, an Iterative Method for Solving Sparse Sets of Simultaneous Linear Equations. *SPE Symposium on Numerical Simulation of Reservoir Performance*, *Society of Petroleum Engineers*. https://doi.org/10.2118/5729-MS

[20] Golub, G. (1965) Numerical Methods for Solving Linear Least Squares Problems. *Numerische Mathematik*, **7**, 206-216. https://doi.org/10.1007/BF01436075

[21] Tuff, A.D. and Jennings, A. (1973) An Iterative Method for Large Systems of Linear Structural Equations. *International Journal for Numerical Methods in Engineering*, **7**, 175-183. https://doi.org/10.1002/nme.1620070207

[22] Lanczos, C. (1952) Solution of Systems of Linear Equations by Minimized Iterations. *J. Res. Nat. Bur. Standards*, **49**, 33-53. https://doi.org/10.6028/jres.049.006

[23] Zhou, Z.H. (2016) Machine Learning. Tsinghua University Press, Beijing.

[24] Schäffler, S., Schultz, R. and Weinzierl, K. (2002) Stochastic Method for the Solution of Unconstrained Vector Optimization Problems. *Journal of Optimization Theory and Applications*, **114**, 209-222. https://doi.org/10.1023/A:1015472306888

[25] Moré, J.J., Garbow, B.S. and Hillstrom, K.E. (1978) Testing Unconstrained Optimization Software. Argonne National Lab. IL (USA). https://doi.org/10.2172/6650344

[26] Powell, M.J.D. (1970) A New Algorithm for Unconstrained Optimization. Nonlinear Programming, Academic Press, 31-65. https://doi.org/10.1016/B978-0-12-597050-1.50006-3

[27] Andrei, N. (2008) An Unconstrained Optimization Test Functions Collection. *Adv. Model. Optim*, **10**, 147-161.

[28] Bhatia, K., Jain, H., Kar, P., *et al.* (2015) Sparse Local Embeddings for Extreme Multi-Label Classification. *Advances in Neural Information Processing Systems*, 730-738.

[29] Concus, P. and Golub, G.H. (1976) A Generalized Conjugate Gradient Method for Nonsymmetric Systems of Linear Equations. *Computing Methods in Applied Sciences and Engineering*. Springer, Berlin, Heidelberg, 1976, 56-65. https://doi.org/10.1007/978-3-642-85972-4_4

[30] Kershaw, D.S. (1978) The Incomplete Cholesky—Conjugate Gradient Method for the Iterative Solution of Systems of Linear Equations. Journal of Computational Physics, **26**, 43-65. https://doi.org/10.1016/0021-9991(78)90098-0