Against the Trend-A tentative Data Analysis Method using Classical Regression against Machine Learning Approach

Fuqing Yuan[1*], Jinmei Lu[1]
[1]Department of Engineering and Safety, University of Tromsø, Tromsø, Norway
E-MAIL: yuan.fuqing@uit.no

Abstract:
The machine learning approach is a new hot topic in recent years that are widely used in different sections, including industries, economy, disaster prediction and politics. After decades' of development, the available machine learning algorithms are numerous and diverse. Traditional methods such as regression, classical statistical methods, are unfortunately laid aside as non-mainstream. This paper tries to compare the classical regression with machine learning algorithm as classifier. Typical machine learning algorithm support vector machine (SVM) is compared with the classical regression. The classical regression is modified to tailor as classifier. Confidence interval and credibility of prediction from regression is developed to evaluate the prediction uncertainty. Benchmark data from public database is used to demonstrate the performance. The results showed that regression exhibits an efficient computational cost with comparative accuracy.

Keywords:
Classical Regression; Machine learning; Accuracy; Computational Cost; Classification

1. Introduction

The surge of artificial intelligence (AI) in recent years captures the headline of newspaper and magazines. In addition, many universities and governments determines the AI as their priority for future development [1-3]. Machine learning, as one of the major components of the AI, is gaining the eyes of plenty of researchers [4]. Aiming at learning from data, the scope of machine learning is broad and includes a wide variety of techniques [5]. It can be categorized as supervised learning and unsupervised learning, or one class classification, binary classification and multi class classification [6]. For the problems to be solved, it can classify as classification or regression problems. From method perspective, it can classify as linear discriminant, KNN, neural network, support vector machine. The taxonomy of the machine learning is not possible to be standardized, as it has plenty of techniques, with plenty of variants, and complexly interconnected.

In the wave of the arising of the above-mentioned techniques, one point is readily ignored: classical statistical methods can also solve the problems instead of using the above-mentioned techniques. The classical statistical methods have solid theoretical foundation. From practical perspective, it is easy to find mature commercial software to implement them. For example, for one class classification, the multi-variant distribution can perform most of the classification tasks. The classification accuracy has a confidence interval from the multi-variant distribution. Another major advantage is the significant reduction in computational time.

This paper firstly examined the machine learning techniques in general way and compared them with the traditional statistical method. Case studies were presented to demonstrate and validate the argument. The section 2 describes the advantage and disadvantage of both techniques. Section 3 describes the development of confidence interval for classical regression. Section 4 presents the case studies. Section 5 presents the conclusions from this study.

2. Technical comparison

Classical regression has been well-developed theoretically. This paper, regardless of the variant of the regression, concerns only the classical linear regression. We do not investigate the correlation among the data or use any pre-processing techniques such as normalization of the data to perform PCA analysis. The physical meaning of the attributes is also ignored. In one word, for comparison with machine learning techniques, we use the simplest linear regression to conduct pattern recognition task. The pattern recognition is also limited to the binary classification problem, as illustrated in Figure 1.

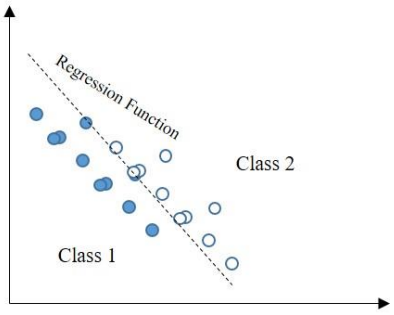

FIGURE 1. Binary Regression Classification

The class label is represented numerically as 1 and -1. The numerical number is considered as the response variable in the regression. The observed data, usually multi-dimensional, is considered as predictor variables. Let the predictor variables written as $Z = z_1, z_2, \dots, z_n$, the response variables as $Y = y_1, y_2, \dots, y_n$. A regression is written as [7]

$$\hat{y} = \hat{\boldsymbol{B}}\boldsymbol{Z} \qquad (1)$$

In this paper, the constant of the regression is considered as a special dimension that is a unity vector. The $\boldsymbol{B}$ is the regression coefficient and is estimated

$$\hat{\boldsymbol{B}} = (\boldsymbol{Z}'\boldsymbol{Z})^{-1}\boldsymbol{Z}'\boldsymbol{Y} \qquad (2)$$

The parameter estimation procedure is equivalent to the training process for the machine learning. From (2), the computational cost of the learning is mainly from the $(\boldsymbol{Z}'\boldsymbol{Z})^{-1}$. Other computational cost is ignorable. The computational complexity is around $o(n^3)$, while there are other estimation methods that can have less complexity.

For machine learning, the state of art methods are plenty. We cannot enumerate all. The performance of them varies significantly. We choose the support vector machine (SVM) as a representative.

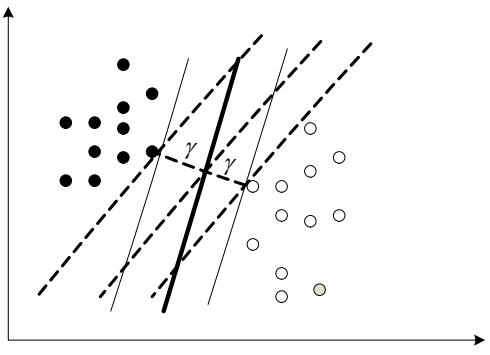

FIGURE 2. Binary SVM Classification

The binary SVM classification is demonstrated in Figure 2. For a problem, which cannot be linearly separated, SVM introduces slack variables ξ_i to tolerate misclassification. The margin for non-separable problem is named soft margin. SVM is to solve a mathematical problem as [8]:

$$\begin{aligned} \max_{\alpha} \quad & -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j < x_i, x_j > +\sum_{k=1}^{m}\alpha_k \\ s.t. \quad & 0 \le \alpha_i \le C, \quad i = 1,2,3...,m \\ & \sum_{i=0}^{m}\alpha_i y_i = 0 \end{aligned} \qquad (3)$$

The C is penalty parameter to penalize misclassification. The α_i is the coefficient to be estimated. The x_i is the same as z_i in (1). The decision function for the classification is an expansion of kernel function as:

$$f(x) = \mathrm{sgn}(\sum_{j=1}^{m}\alpha_j y_j K(x, x_j) + b) \qquad (4)$$

A Gaussian kernel function for (4) is as follows

$$k(x, z) = \exp(-\frac{\|x - z\|}{2\sigma^2}) \qquad (5)$$

The σ is predefined parameter and defines the sensitivity of the classification. Higher σ implies higher sensitivity. The computational complexity of training SVM, i.e. finding the solutions for (3), is $o(n^2 \cdot m)$, where m is the number of attributes, or the dimension of the input data [9].

The SVM has much complex model. Classical regression is computed using matrix manipulation. The SVM is to solve an optimization with two constraints. Obviously, SVM has more complex computational procedure. With knowledge of the SVM and the classical regression, we compare them in the following sections.

2.1. Data driven vs model based

The machine learning algorithms are in general purely data-driven, as illustrated in Figure 3. The physical meaning of the parameters, for example the α_i in the (4) are determined by the data. They are not constant, but evolve with data. The other internal parameters such as the penalty constant C in (3) has no physical meaning neither. The

parameters are determined by the output.

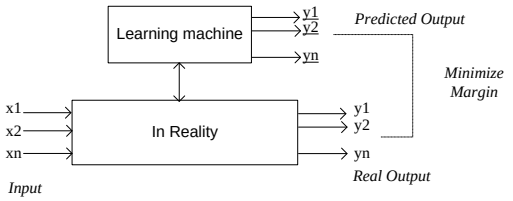

FIGURE 3. Data-Driven Machine Learning

The parameters' evolution with data is a double-edged sword: it can update itself with data. It is result-driven principle that does not require the analyzer to waste time to find out the physical background. This means that the model is not stable since for most practical situation, finding out the physical background is not possible.

The classical regression is model driven. The coefficients in the regression can be interpreted by the data. However, one does not need the physical meaning when it is used for classification.

2.2. Computational Cost

Most machine learning algorithms are notorious for its extremely high computational cost. We can take formula (2) and (3) as examples. The computational cost of regression is $o(n^3)$. The cost is from the inverse of matrix. The SVM is $o(n^2.m)$. The computational cost seems similar. However, the SVM's internal parameters such as the C and the width of Gaussian kernel function has to be determined before applying them. The determination of these parameters is costly and this is still an open question. The crude grid search is one of the best and feasible approach. If using the grid search, one parameter chooses 1000 samples. For SVM has two internal parameters, one should run the SVM $1000*1000=10^6$ times. The computational cost increases significantly.

Even there are numerous publications aiming at solving this problem, the computational cost is still high. Heuristic method is one of the solutions. However, heuristic method cannot guarantee the best solution. The SVM depends on the internal parameters very sensitively. However, the classical regression based classification does not have this problem.

2.3. Predicting Uncertainty

Another major problem for machine learning is the overfitting problem. Machine learning algorithm can be very flexible to fit data. For the training data, it is not hard to find a learning algorithm that can classify the training data with 100% accuracy. For SVM, one can tune the internal parameters by selecting denser grid, when using grid search method. For Neural network, one can use more neurons, or choose more layers of network. However, the generalization performance, i.e. the predicting capacity might not increase with a more complex learning algorithm. A perfect fitting on the training data does not guarantee a perfect prediction when using the machine learning algorithms.

The learning algorithm is highly dependent on the data. When data changed, the learning algorithm will change. Therefore, the prediction results might vary. The classical regression is less flexible than most learning algorithm. However, when it is used for classicization, it has the same predicting uncertainty problem, since its coefficients are also determined by the data. Fortunately, the classical regression has been already intensively investigated. The distribution of the coefficients can be analytically estimated. Therefore, the prediction uncertainly can be estimated.

3. Confidential interval of accuracy

Considering a binary classification as a regression problem. The training is essentially the parameter estimation of the regression. The class label denoted by the numerical value, if it is positive, i.e.>0, is class 1; otherwise the class label is -1. Let the estimated parameters in the regression be $\hat{\beta}$. The response variable then follows Normal distribution [7]

$$\frac{(y_0 - z_0\hat{\beta})}{\sqrt{s^2(1+z_0(Z'Z)^{-1}z_0)}} \sim N(0,1) \qquad (6)$$

where $s^2 = \frac{Y'(I - Z(Z'Z)^{-1}Z')Y}{n-r-1}$. I is an $n \times n$ identical matrix. The r is the number of predictors. When class label is numerated as 1 and -1, the $y_0 = 0$ is the boundary to classify the two categories. The z_0 is the vector to be labelled. Distance to the 0 denotes the significance of the classified label. A confidence interval can be derived from the distribution of (6).

The $z_0\hat{\beta}$ value of less than 0 is labelled -1; otherwise it is labeled 1. The values locate near 0 is not significant, as shown in the shape area in Figure 4. If we define the probability with certain level, e.g. with probability 0.1, and close to 0 is not significant, we can obtain a confidence

interval.

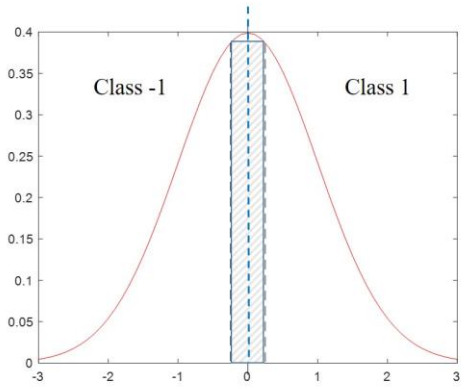

FIGURE 4. Significance of Classification

Suppose the number of objectives to be classified is n, the number of objectives located in shape area is n_u, and the number of corrected objectives are n_c, the confidence interval is then

$$CI = [\frac{n_c}{n} - \frac{n_u}{n}, \frac{n_c}{n} + \frac{n_u}{n}] \qquad (7)$$

In the classification, it is often of interest to estimate the credibility of the classified label for a given data. A predicted value far from the boundary implies higher credibility of the classification label. A credibility of the label classification can be defined as probability of the value deviating from 0, i.e. the percentage of the shadowed area in Figure 5 capturing the half of the distribution. It is

$$\text{Creditability} = \frac{P(Distance\ to\ 0)}{0.5} \qquad (8)$$

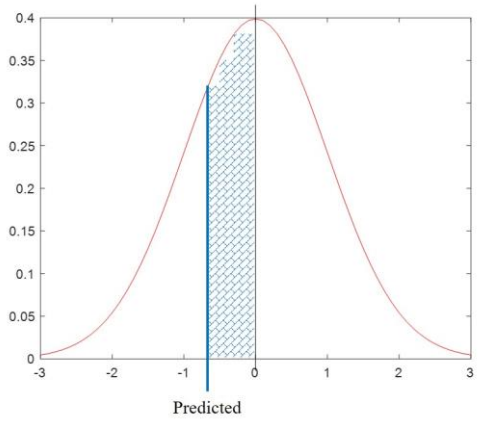

FIGURE 5. Credibility of Classification

Practically, the $P(Distance\ to\ 0)$ can be calculated using

$$0.5 - N(\frac{z_0\hat{\beta}}{\sqrt{s^2(1+z_0(Z'Z)^{-1}z_0)}}) \qquad (9)$$

where N is standard normal distribution. The higher the value of (4), the higher the credibility of the classification is.

The confidence interval and the credibility is derived, which is hard to be estimated for most machine learning algorithms. This is one advantage to use classical regression for classification task.

4. Case studies

Three case studies are performed to use machine learning and the classical regression for binary classification. The data used are open data that are available for the public. Aim of the case studies is not to find the best approach, but to compare the machine learning with the classical regression for classification.

4.1. Grid stability data sets

The first data used for the case study is from the UCI machine learning repository for grid stability of electricity [10]. We don't investigate the background of the data since it is not the aim of the paper. Data is multivariate with number of instances 10000. It is with 14 attributes. All attributes are real and continuous. The classification is binary with "stable" or "unstable" label. In the classification, we label the "stable" as 1 and the "unstable" as -1 numerically. Just for comparison, we use 80% of data for training or for parameter estimation, and the remaining 20% for validation.

The data size 10000 is large. For regression, it is sufficient for the reliable estimated parameters. Running on the normal PC, the time consumed for computation is 0.008909 seconds. The parameters obtained is 0.288, -0.00166, -0.00272, -0.00319, -0.00252, 0.00300, 0.01126, 0, -0.0072, -0.1658, -0.1585, -0.1364, -0.1626 with constant coefficient 23.07 in the linear regression. The accuracy achieved is 97.9%. According to (7), the confidence interval is [87.5%, 98.95%].

The initial regression shows one attribute that is with 0 coefficient. We can exclude it in the regression to expect a better accuracy. After running using Matlab, the accuracy remains 97.9%, but the confident interval is narrower with [88.15%, 99.55%].

Machine learning methods algorithms were also run for comparison. Support vector machine, linear

discriminant, KNN are used. It is also 80% of the data for training, and 20% of the data for validating. The results are presented in Table 1.

TABEL 1. Performance of the Machine Learning

Machine Learning	Key Parameter Values	Accuracy	Training Time (s)
SVM 1	Linear	99.7%	5.0462
SVM 2	Quadratic	98.8%	20.02
SVM 3	Cubic	98%	23.47
SVM Fine Gaussian	0.9	64.1%	20.90
SVM Medium Gaussian	3.6	98.6%	9.42
SVM Coarse Gaussian	14	98.8%	12.83
Linear Discriminant		96.2%	0.9723
Fine KNN	Equal Euclidean Distance. Nr of Neighbors 1	88.6%	2.03
Medium KNN	Equal Euclidean Distance. Nr of Neighbors 10	94.7%	2.13
Coarse KNN	Equal Euclidean Distance. Nr of Neighbors 100	94.7%	2.04
Cubic KNN	Equal Cubic Distance. Nr of Neighbors 10	94.5%	17.4
Weighted KNN	Squared inverse Cubic Distance. Nr of Neighbors 10	94.7%	2.36
Boosted Tress		63.8%	3.28
Bagged Trees		100%	4.58
Rusboosted Trees		100%	3.32

For this case, as shown in the Table, the traditional linear regression has an accuracy of 97.9%. It can outperform some of the machine learning techniques. However, in terms of computational time, the traditional regression is significantly shorter with only 0.008909 seconds. None of the machine learning can reach this efficient level. The simple regression gains a good performance.

4.2. Ionosphere data sets

The second case study uses the ionosphere data from the UCI machine Learning repository [11]. It is also a binary classification problem. Data contain 351 instances and 34 attributes. All 34 attributes are continuous. The data are perfect for the regression. Same as in case study 4.1, we use 80% of the data for training and 20% of the data for validating.

In the classical linear analysis, the accuracy reaches 97.18% with computational time around 0.012 seconds. The computation is very efficient. The machine learning algorithm chooses the SVM only. In the SVM analysis, the penalty parameters choose infinite. The epsilon which represents the tolerance of the error is as small as 0.000001. The kernel function chooses Gaussian function with parameter 1. The coding is performed by Matlab. The classical regression is also coded using Matlab. Both SVM and regression run in the same software environment, aiming at comparing the computational time fairly. The SVM can reach 100% accuracy with computational time 0.77 seconds, whose accuracy is higher than the regression but computational time of regression is much shorter with only 0.012 seconds.

In this case, the SVM achieves a perfect performance of 100% accuracy. The classical linear regression has lower accuracy but uses much shorter computational time. The other machine learning algorithm is not used. One reason was some machine learning failed for prediction. Most of them can achieve only around 70%. It might be due to the software error in the Matlab.

4.3. Parkinson disease data sets

The third case concerns with the Parkinson's disease classification [12]. Data sets contains 756 instances and 754 attributes. It is a binary classification problem. The number of attributes are almost the same as the number of instances. All attributes are numerical. The first attribute of the data is ID. We exclude it in the data analysis since it is obviously not relevant to the classification. The other attributes such as age are kept for analysis.

For the data analysis with regression, the data is not well represented since the number of attributes is too big, in contrast with the number of instances. Theoretically, the estimated regression function will not be with high accuracy. It can be validated from the (1). The standard deviation of the estimation will increase when the n and r is close.

For both linear regression and machine learning, we use 80% data for training, and the remaining 20% data for validating. With computational time around 0.055 seconds, the classical regression has an accuracy of 69.74%. Classical regression does not show a good performance in terms of accuracy.

TABLE 2. Performance of the Machine Learning

Machine Learning	Key Parameter Values	Accuracy	Training Time (s)
SVM 1	Linear	84.1%	9.60
SVM 2	Quadratic	87.4%	9.80
SVM 3	Cubic	90.1%	10.4
SVM Fine Gaussian	6.9	74.2%	11.57
SVM Medium Gaussian	27	84.8%	11.28

SVM Coarse Gaussian	110	77.5%	12.15
Fine KNN	Equal Euclidean Distance. Nr of Neighbors 1	90.7%	13.23
Medium KNN	Equal Euclidean Distance. Nr of Neighbors 10	84.1%	12.95
Coarse KNN	Equal Euclidean Distance. Nr of Neighbors 100	76.2%	13.29
Cosine KNN	Nr of Neighbors 10	84.8%	14.19
Cubic KNN	Equal Cubic Distance. Nr of Neighbors 10	80.8%	20.57
Weighted KNN	Squared inverse Cubic Distance. Nr of Neighbors 10	84.8%	14.46
Fine Tress		80.1%	10.34
Medium Trees		80.1%	9.621
Coarse Trees		76.8%	9.38

In this case study, the classical regression does not show a good performance. The best performance is the Fine KNN. The KNN shows an outstanding performance as it is suitable for the high dimensional data. The lower performance of classical regression is mainly due to the data structure. The number of attributes r has the same size as the number of instances. For this original data, the coefficient evaluated from (2) is with wide uncertainty. The variance of the efficiencies is inverse proportional to the n-r. In this case, the n-r is almost 0. The regression essentially fails to be used as classifier. However, the failure is tractable. Before applying it, one can check the n-r values. In spite of this problem, the regression can reach same level of accuracy when the SVM chooses the improper kernel function and internal parameters values, i.e. the SVM using Gaussian Kernel function in the Table 2.

5. Conclusions

The paper uses the classical regression as a classifier to investigate the performance of regression as a learning algorithm. The classical regression used as a classifier has a well-defined statistical foundation. From the demonstrated three case studies, the regression can achieve rather good comparative performance than machine learning algorithms. Albeit it is not the best, it processes stable performance in terms of the accuracy, when the accuracy is defined as the predicting accuracy. While in terms of computational efficiency, the regression has fairly better performance than all the applied machine learning algorithms in the 3 cases. This is a major advantage of this simple regression used for classicization.

For a task to be accomplished by machine learning algorithm, using the simple regression method could achieve an unexpected good performance. The typical machine learning algorithm might achieve higher accuracy, but its performance is not stable. For some cases, it may even fail completely. Practically, as the machine learning algorithms have typically high computational cost and complex procedure for optimization, the computation has to rely on commercial or third party software. The hidden failure of the algorithm in commercial software is not easy to be notified. In this sense, the regression is much easy to be understood. One can code themselves according to their own purpose. Conclusively, one should be aware the classical method might still be feasible and it might be able to achieve a better performance for their applications than machine learning approach.

References

[1] A. Azizi, *Applications of artificial intelligence techniques in Industry 4.0*, Singapore: Springer,, 2019, p. 1 online resource. [Online]. Available: http://dx.doi.org/10.1007/978-981-13-2640-0 MIT Access Only.

[2] L. Luce, *Artificial intelligence for fashion : how AI is revolutionizing the fashion industry*, Berkeley, CA: Apress,, 2019, p. 1 online resource. [Online]. Available: http://dx.doi.org/10.1007/978-1-4842-3931-5 MIT Access Only.

[3] J. Turner, *Robot rules : regulating artificial intelligence*, Cham, Switzerland: Palgrave Macmillan,, 2019, pp. 1 online resource (xx, 377 pages). [Online]. Available: http://dx.doi.org/10.1007/978-3-319-96235-1 MIT Access Only.

[4] J. R. Anderson, R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, *Machine learning : an artificial intelligence approach*. Palo Alto, Calif.: Tioga Pub. Co., 1983, pp. xi, 572 p.

[5] P. Langley, *Elements of machine learning* (Morgan Kaufmann series in machine learning). San Francisco, Calif.: Morgan Kaufmann, 1996, pp. xii, 419 p.

[6] Y. Fuqing, "Failure Diagnosis using Support Vector Machines," PhD, Div of Operation and Maintenance, Lulea University of Technology, Lulea Sweden, 2012.

[7] R. A. Johnson and D. W. Wichern, *Applied multivariate statistical analysis*, 5th ed. Upper Saddle River, N.J.: Prentice Hall, 2002, p. 767 p.

[8] V. N. Vapnik, *The nature of statistical learning theory*. New York: Springer, 1995, pp. xv, 188 p.
[9] A. A. R. Wardoyo, "Time Complexity Analysis of Support Vector Machines (SVM) in LibSVM," *International Journal of Computer Applications,* vol. 128, no. 3, pp. 29-34, 2015.
[10] V. Arzamasov, "Electrical Grid Stability Simulated Data Data Set," K. I. o. Technology, Ed., ed. USA: UCI Machine Learning Repository, 2018.
[11] V. Sigillito. Ionosphere Data Set
[12] G. S. b. C. Okan Sakar a, Aysegul Gunduz c. Parkinson's Disease Classification Data Set