4-2020

Why Geometric Progression in Selecting the LASSO Parameter: A Theoretical Explanation

William Kubin

Yi Xie

Laxman Bokati

Vladik Kreinovich

Kittawit Autchariyapanitkul

Comments:
Technical Report: UTEP-CS-20-35

Why Geometric Progression in Selecting the LASSO Parameter: A Theoretical Explanation

William Kubin[1], Yi Xie[1], Laxman Bokati[1],
Vladik Kreinovich[1], and Kittawit Autchariyapanitkul[2]
[1]Computational Science Program
University of Texas at El Paso
ElPaso, Texas 79968, USA
wkubin@miners.utep.edu, yxie3@miners.utep.edu,
lbokati@miners.utep.edu, vladik@utep.edu
[2]Maejo University, Thailand, kittawit_a@mju.ac.th

Abstract

In situations when we know which inputs are relevant, the least squares method is often the best way to solve linear regression problems. However, in many practical situations, we do not know beforehand which inputs are relevant and which are not. In such situations, a 1-parameter modification of the least squares method known as LASSO leads to more adequate results. To use LASSO, we need to determine the value of the LASSO parameter that best fits the given data. In practice, this parameter is determined by trying all the values from some discrete set. It has been empirically shown that this selection works the best if we try values from a geometric progression. In this paper, we provide a theoretical explanation for this empirical fact.

1 Formulation of the Problem

Need for regression. In many real-life situations, we know that the quantity y is uniquely determined by the quantities $x_1, \ldots, x_n$, but we do not know the exact formula for this dependence. For example, in physics, we know that the aerodynamic resistance increases with the body's velocity, but we often do not know how exactly. In economics, we may know that a change in tax rate influences the economic growth, but we often do not know how exactly.

In all such cases, we need to find the dependence $y = f(x_1, \ldots, x_n)$ between several quantities based on the available data, i.e., based on the previous observations $(x_{k1}, \ldots, x_{kn}, y_k)$ in each of which we know both the values x_{ki} of the input quantities x_i and the value y_k of the output quantity y. In statistics, determining the dependence from the data is known as *regression*.

Need for linear regression. In most cases, the desired dependence is smooth – and usually, it can even be expanded in Taylor series; see, e.g., [1, 5]. In many practical situations, the range of the input variables is small, i.e., we have $x_i \approx x_i^{(0)}$ for some values $x_i^{(0)}$. In such situations, after we expand the desired dependence in Taylor series, we can safely ignore terms which are quadratic or of higher order with respect to the differences $x_i - x_i^{(0)}$ and only keep terms which are linear in terms of these differences:

$$y = f(x_1, \ldots, x_n) = c_0 + \sum_{i=1}^{n} a_i \cdot \left(x_i - x_i^{(0)}\right),$$

where $c_0 \stackrel{\text{def}}{=} f\left(x_1^{(0)}, \ldots, x_n^{(0)}\right)$ and $a_i \stackrel{\text{def}}{=} \dfrac{\partial f}{\partial x_{i\,|x_i = x_i^{(0)}}}$. This expression can be simplified into a general linear expression:

$$y = a_0 + \sum_{i=1}^{n} a_i \cdot x_i, \tag{1}$$

where $a_0 \stackrel{\text{def}}{=} c_0 - \sum_{i=1}^{n} a_i \cdot x_i^{(0)}$.

In practice, measurements are never absolutely precise, so when we plug in the actually measured values x_{ki} and y_i, we will only get an approximate equality:

$$y_k \approx a_0 + \sum_{i=1}^{m} a_i \cdot x_{ki}. \tag{2}$$

Thus, the problem of finding the desired dependence can be reformulated as follows:

- given the values y_k and x_{ki},
- find the coefficients a_i for which the property (2) holds for all k.

The usual least squares approach. We want each left-and side y_k of the approximate equality (2) to be close to the corresponding right-hand side. In other words, we want the tuple $(y_1, \ldots, y_K)$ consisting of all the left-hand sides to be close to a similar tuple formed by the right-sides

$$\left(\sum_{i=1}^{m} a_i \cdot x_{1i}, \ldots, \sum_{i=1}^{m} a_i \cdot x_{Ki}\right).$$

It is reasonable to select the values a_i for which the distance between these two tuples is the smallest possible. Minimizing the distance is equivalent to minimizing the square of this distance, i.e., the expression

$$\sum_{k=1}^{K} \left(y_k - \left(a_0 + \sum_{i=1}^{m} a_i \cdot x_{ki}\right)\right)^2. \tag{3}$$

This minimization is know as the *Least Squares method*. This is the most widely used method for processing data. The corresponding values a_i can be easily found if we differentiate the quadratic expression (3) with respect to each of the unknowns a_i and then equate the corresponding linear expressions to 0. Then, we get an easy-to-solve systems of linear equations.

Comment. The above heuristic idea becomes well-justified when we consider the case when the measurement errors are normally distributed with 0 mean and the same standard deviation σ. This indeed happens in many situations when the measuring instrument's bias has been carefully eliminated, and most major sources of measurement errors have been removed. In such situations, the resulting measurement error is a joint effect of many similarly small error components. For such joint effects, the Central Limit Theorem states that the resulting distribution is close to Gaussian (= normal); see, e.g., [4]. Once we know the probability distributions, a natural idea is to select the most probable values a_i, i.e., the values for which the probability to observe the values y_k is the largest. For normal distributions, this idea leads exactly to the least squares method.

Need to go beyond least squares. When we know that all the inputs x_i are essential to predict the value y of the desired quantity, the least squares method works reasonably well. The problem is that in practice, we often do not know which inputs x_i are relevant and which are not. As a result, to be on the safe side, we include as many inputs as possible, perfectly understanding that many of them will turn out to be irrelevant.

If all the measurements were exact, this would not be a problem: for irrelevant inputs x_i, we would get $a_i = 0$, and the resulting formula would be the desired one. However, because of the measurement errors, we do not get exactly 0s. Moreover, the more such irrelevant variables we add, the more non-zero "noise" terms $a_i \cdot x_i$ we will have, and the larger will be their sum – negatively affecting the accuracy of the formula (3) and thus, of the resulting desired (non-zero) coefficients a_i.

LASSO method. Since we know that many coefficients will be 0, a natural idea is, instead of considering all possible tuples $a \stackrel{\text{def}}{=} (a_0, a_1, \ldots, a_n)$, to only consider tuples for which a bounded number of coefficients is 0, i.e., for which $\|a\|_0 \le B$ for some constant b, where $\|a\|_0$ (known as the ℓ_0-norm) denotes the number of non-zero coefficients in a tuple.

The problem with this natural idea is that the resulting optimization problem becomes NP-hard, which means, crudely speaking, that no feasible algorithm is possible that would always solve all the instances of this problem. A usual way to solve such problem is by replacing the ℓ_0-norm with an ℓ_1-norm $\sum\limits_{i=0}^{n} |a_i|$ which is convex and for which, therefore, the optimization problem is easier to solve. So, instead of solving the problem of unconditionally minimizing the expression (3), we minimize this expression under the constraint $\sum\limits_{i=0}^{n} |a_i| \le B$ for some

constant B. This minimum can be attained when we have strict inequality or when the constraint becomes an equality. If the constraint is a strict inequality, then we have a local minimum of (3), which, for quadratic functions, is exactly the global minimum that we try to avoid. Thus, to avoid using least squares, we must consider the case when the constraint becomes an equality $\sum\limits_{i=0}^{n} |a_i| = B$.

According to the Lagrange multiplier method, minimizing a function under an equality-type constraint is equivalent, for an appropriate value of a parameter λ, to unconstraint minimization of the linear combination of the original objective function and the constraint, i.e., to minimizing the expression

$$\sum_{k=1}^{K} \left(y_k - \left(a_0 + \sum_{i=1}^{m} a_i \cdot x_{ki} \right) \right)^2 + \lambda \cdot \sum_{i=0}^{n} |a_i|. \tag{4}$$

This minimization is known as the *Least Absolute Shrinkage and Selection Operator* method – *LASSO method*, for short; see, e.g., [3, 6].

How the LASSO parameter λ is selected: main idea. The success of the LASSO method depends on what value λ we select. When λ is close to 0, we retain all the problems of the usual least squares method. When λ is too large, the λ-term dominates, so we select the values $a_i = 0$, which do not provide any good description of the desired dependence.

In different situations, different values λ will work best. The more irrelevant inputs we have, the more important it is to deviate form the least squares, and thus, the larger the parameter λ – that describes this deviation – should be. We rarely know beforehand which inputs are relevant – this is the whole problem – so we do now know beforehand what value λ we should use. The best value λ needs to be decided based on the data.

A usual way of testing any dependence is by randomly dividing the data into a (larger) training set and a (smaller) testing set. We use the training set to find the value of the desired parameters (in our case, the parameters a_i), and then we use the testing set to gauge how good is the model. As usual with the methods using randomization, to get more reliable results, we can repeat this procedure several times, and make sure that the results are good for all cases,

In precise terms, we select several training subsets $S_1, \ldots, S_m \subseteq \{1, \ldots, K\}$. For each of these subsets S_j, we find the values $a_{ij}(\lambda)$ that minimize the functional

$$\sum_{k \in S_j} \left(y_k - \left(a_0 + \sum_{i=1}^{m} a_i \cdot x_{ki} \right) \right)^2 + \lambda \cdot \sum_{i=0}^{n} |a_i|. \tag{5}$$

We can then compute the overall inaccuracy, as

$$\Delta(\lambda) \stackrel{\text{def}}{=} \sum_{j=1}^{m} \left(\sum_{k \notin S_j} \left(y_k - \left(a_{j0}(\lambda) + \sum_{i=1}^{m} a_{ji}(\lambda) \cdot x_{ki} \right) \right)^2 \right). \tag{6}$$

We then select the value λ for which the corresponding inaccuracy is the smallest possible.

How the LASSO parameter λ is selected: details. In the ideal world, we should be able to try all possible real values λ. However, there are infinitely many real numbers, and in practice, we can only test finitely many of them. Which set of values λ should we choose?

It turned out that empirically, the best results are obtained of we use the values λ that form a geometric progression $\lambda_n = c_0 \cdot q^n$. Of course, a geometric progression also has infinitely many values, but we do not need to test all of them: usually, as λ increases from 0, the value $\Delta(\lambda)$ first decreases then increases again, so it is enough to catch a moment when this value starts increasing.

Natural question and what we do in this paper. A natural question is: why geometric progression works best? In this paper, we provide a theoretical explanation for this empirical fact.

2 Our Result

What do we want? At first glance, the answer to this question is straightforward: want to select a discrete set of values, i.e., a set

$$S = \{\ldots < \lambda_n < \lambda_{n+1} < \ldots\}.$$

However, a deeper analysis shows that the answer is not so simple. Indeed, what we are interested in is the dependence between the quantities y and x_i. However, what we have to deal with is not the quantities themselves, but their numerical values, and the numerical values depend on what unit we choose for measuring these quantities. For example:

- a person who is 1.7 m high is also 170 cm high,
- an April 2020 price of 2 US dollars is the same as the price of $2 \cdot 23500 = 47000$ Vietnam Dong, etc.

In most cases, the choice of the units is rather arbitrary. It is therefore reasonable to require that the results of data processing – when converted to original units – should not depend on which units we originally used. And hereby lies a problem. Suppose that we keep the same units for x_i but change a measuring unit for y to a one which is α times smaller. In this case, the new numerical values of y become α times larger: $y \to y' = \alpha \cdot y$. To properly capture these new values, we need to increase the original values a_i by the same factor, i.e., replace the values a_i with the new values $a'_i = \alpha \cdot a_i$. In terms of these new values, the minimized expression (4) takes the form

$$\sum_{k=1}^{K} \left(y'_k - \left(a'_0 + \sum_{i=1}^{m} a'_i \cdot x_{ki} \right) \right)^2 + \lambda \cdot \sum_{i=0}^{n} |a'_i|,$$

i.e., taking into account that $y'_k = \alpha \cdot y_k$ and $a'_i = \alpha \cdot a_i$, the form

$$\alpha^2 \cdot \sum_{k=1}^{K} \left(y_k - \left(a_0 + \sum_{i=1}^{m} a_i \cdot x_{ki} \right) \right)^2 + \alpha \cdot \lambda \cdot \sum_{i=0}^{n} |a_i|.$$

Minimizing an expression is the same as minimizing α^{-2} times this expression, i.e., the modified expression

$$\sum_{k=1}^{K}\left(y_k-\left(a_0+\sum_{i=1}^{m}a_i\cdot x_{ki}\right)\right)^2+\alpha^{-1}\cdot\lambda\cdot\sum_{i=0}^{n}|a_i|.$$

This new expression is similar to the original minimized expression (4), but with a new value of the LASSO parameter $\lambda'=\alpha^{-1}\cdot\lambda$.

What this says is that when we change the measuring units, the values of λ are also re-scaled – i.e., multiplied by a constant. What was the set $\{\lambda_n\}$ in the old units becomes the re-scaled set $\{\alpha^{-1}\cdot\lambda_n\}$ in the new units. Since this is, in effect, the same set but corresponding to different measuring units, we cannot say that one of these sets is better than the other, they clearly have the same quality.

So, we cannot choose a single set S, we must choose a family of sets $\{c\cdot S\}_c$, where the notation $c\cdot S$ means $\{c\cdot\lambda:\lambda\in S\}$.

Natural uniqueness requirement. Eventually, we need to select some set S. As we have just explained, we cannot select one set a priori, since with every set S, a set $c\cdot S$ also has the same quality. To fix a unique set, we can, e.g., fix one of the values $\lambda\in S$. Let us require that with this fixture, we will be end up with a unique optimal set S. This means, in particular, that, if we select a real number $\lambda\in S$, then the only set $c\cdot S$ that contains this number will be the same set S.

Let us describe this requirement in precise terms.

Definition 1. *By a* discrete set*, we mean a subset S of the set $\mathbb{R}^+$ of all positive real numbers for which, for every $\lambda\in S$, there exists an $\varepsilon>0$ such that for every other element $\lambda'\in S$, we have $|\lambda-\lambda'|>\varepsilon$.*

Comment. For such sets, for each element λ, if there are larger elements, then there is the "next" element – i.e., the smallest element which is larger than λ. Similarly, if there are smaller elements, then there exists the "previous" element – i.e., the largest element which is smaller than λ. Thus, such sets have the form $\{\ldots<\lambda_{n-1}<\lambda_n<\lambda_n<\ldots\}$

Notation. *For each set S and for each number $c>0$, by $c\cdot S$, we mean the set*

$$\{c\cdot\lambda:\lambda\in S\}.$$

Definition 2. *We say that a discrete set S is —em uniquely determined if for every $\lambda\in S$ and $c>0$, if $\lambda\in c\cdot S$, then $c\cdot S=S$.*

Proposition 1. *A set S is uniquely determined if and only if it is a geometric progression, i.e., if and only if it has the form*

$$S=\{c_0\cdot q^n:n=\ldots,-2,-1,0,1,2,\ldots\}$$

for some c_0 and q.

Discussion. This results explains why geometric progression is used to select the LASSO parameter λ.

Proof. It is easy to prove that every geometric progression is uniquely determined. Indeed, if for $\lambda = c_0 \cdot q^n$, we have $\lambda \in c \cdot S$, this means that $\lambda = c \cdot c_0 \cdot q^m$ for some m, i.e., $c_0 \cdot q^n = c \cdot c_0 \cdot q^m$. Dividing both sides by $c_0 \cdot q^m$, we conclude that $c = q^{n-m}$ for some integer $n - m$. Let us show that in this case, $c \cdot S = S$. Indeed, each element x of the set $c \cdot S$ has the form $x = c \cdot c_0 \cdot q^k$ for some integer k. Substituting $c = q^{n-m}$ into this formula, we conclude that $x = c_0 \cdot q^{k+(n-m)}$, i.e., that $x \in S$. Similarly, we can prove that if $x \in S$, then $x \in c \cdot S$.

Vice versa, let us assume that the set S is uniquely determined. Let us pick any element $\lambda \in S$ and denote it by λ_0. The next element we will denote by λ_1, the next to next by λ_2, etc. Similarly, the element previous to λ_0 will be denoted by λ_{-1}, previous to previous by λ_{-2}, etc. Thus,

$$S = \{\ldots, \lambda_{-2}, \lambda_{-1}, \lambda_0, \lambda_1, \lambda_2, \ldots\}.$$

Clearly, $\lambda_1 \in S$, and for $q \stackrel{\text{def}}{=} \lambda_1/\lambda_0$, we have $\lambda_1 \in q \cdot S$ – since $\lambda_1 = (\lambda_1/\lambda_0) \cdot \lambda_0 = q \cdot \lambda_0$ for $\lambda_0 \in S$. Since the set S is uniquely determined, this implies that $q \cdot S = S$. Since

$$S = \{\ldots, \lambda_{-2}, \lambda_{-1}, \lambda_0, \lambda_1, \lambda_2, \ldots\},$$

we have

$$q \cdot S = \{\ldots, q \cdot \lambda_{-2}, q \cdot \lambda_{-1}, q \cdot \lambda_0, q \cdot \lambda_1, q \cdot \lambda_2, \ldots\}.$$

The sets S and $q \cdot S$ coincide. We know that $q \cdot \lambda_0 = \lambda_1$. Thus, the element next to $q \cdot \lambda_0$ in the set $q \cdot S$ – i.e., the element $c \cdot \lambda_1$ – must be equal to the element which is next to λ_1 in the set S, i.e., to the element λ_2: $\lambda_2 = q \cdot \lambda_1$. For next to next elements, we get $\lambda_3 = q \cdot \lambda_2$ and, in general, we get $\lambda_{n+1} = q \cdot \lambda_n$ for all n – which is exactly the definition of a geometric progression.

The proposition is proven.

Comment. Similar arguments can explain why in machine learning methods such as deep learning (see, e.g., [2]) – which usually use the gradient method $x_{i+1} = x_i - \lambda_i \cdot \dfrac{\partial J}{\partial x_i}$ to find the minimum of an appropriate objective function J, empirically the best strategy for selecting λ_i also follows approximately a geometric progression: e.g., some algorithms use:

- $\lambda_i = 0.1$ for the first ten iterations,
- $\lambda_i = 0.01$ for the next ten iterations,
- $\lambda_i = 0.001$ for the next ten iterations, etc.

Indeed, in this case, similarly, re-scaling of J is equivalent to re-scaling of λ, and thus, we need to have a family of sequences $\{c \cdot \lambda_i\}$ corresponding to different $c > 0$. A natural uniqueness requirement – as we have shown – leads to the geometric progression.

Acknowledgments

This work was supported in part by the US National Science Foundation grants 1623190 (A Model of Change for Preparing a New Generation for Professional Practice in Computer Science) and HRD-1242122 (Cyber-ShARE Center of Excellence).

References

[1] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics*, Addison Wesley, Boston, Massachusetts, 2005.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, Cambridge, Massachusetts, 2016.

[3] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, Chapman & Hall/CRC, Boca Raton, Florida, 2015.

[4] D. J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, Chapman and Hall/CRC, Boca Raton, Florida, 2011.

[5] K. S. Thorne and R. D. Blandford, *Modern Classical Physics: Optics, Fluids, Plasmas, Elasticity, Relativity, and Statistical Physics*, Princeton University Press, Princeton, New Jersey, 2017.

[6] R. Tibshirani, "Regression shrinkage and selection via the LASSO", *Journal of the Royal Statistical Society*, 1996, Vol. 58, No. 1, pp. 267–288.