REGRESSION
OVERVIEW

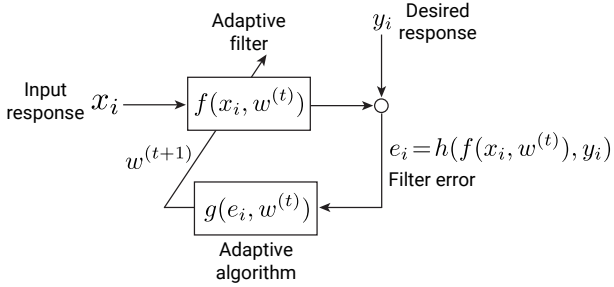
A major problem in machine learning is regression. Regression is aimed at prediction: we want to fit a model to a series of observations to predict the responses of new observations.

This problem can be described by a mapping $f : \mathcal{J} \to \mathcal{K}$, where $\mathcal{J}$ is the input space and $\mathcal{K}$ is the output space. There are many choices for the input spaces and usually only a single choice for the output space:

Input Space: Continuous variables ($\mathcal{R}$, $\mathcal{R}^n$, ...), discrete variables ($\{1, \dots, n\}$, ...), structured variables (trees, strings, ...), and many more. Most regression is performed on continuous variables or discrete variables.

Output Space: Continuous ($\mathcal{R}$, $\mathcal{R}^n$, ...).

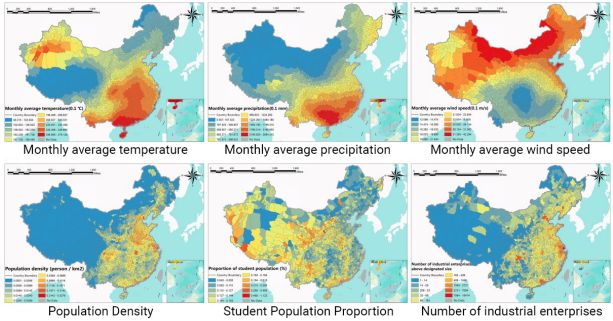
REGRESSION
OVERVIEW

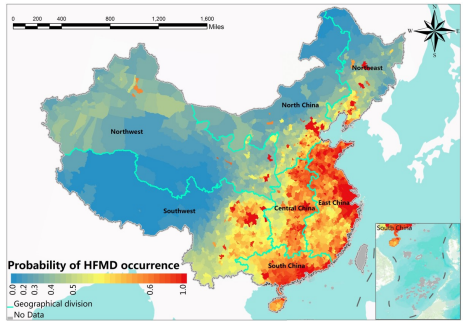


We can view regression as an adaptive filter. We can perform regression in the space of the multivariate input (above figure). In this case, we are trying to predict a desired response $y_i \in \mathcal{R}$ solely from the immediate input $x_i \in \mathcal{R}^m$. We can also do regression in time. In this case, $x_i \in \mathcal{R}$ is a scalar element of a time series. We are trying to predict the response $y_i \in \mathcal{R}$ not only using x_i, but also the history of previous signals $x_{i-1}, x_{i-2}, \cdots$

REGRESSION

MOTIVATING EXAMPLE: DISEASE PREDICTION





Suppose that we were tasked with predicting where a disease (e.g., hand, foot, and mouth disease (HFMD)) might occur. To do this, we would gather a spatial occurrence distribution to show where the disease cases exist. We would then attempt to characterize these occurrences using various features (left figure). Regression can be used to aggregate these characteristics for each region and determine potential outbreaks (right figure).

REGRESSION
OVERVIEW

Suppose that we have a series of examples $\{x_i, y_i\}$, $i=1, \dots, n$, where $x_i \in \mathcal{R}^m$ is the independent variable or input observation and $y_i \in \mathcal{R}$ is the dependent variable or output response. If we are given a new example x_{n+1}, we would like to predict an appropriate response y_{n+1}.

To do this, we will need to assume that there is a relationship $f : \mathcal{R}^m \to \mathcal{R}$ between the independent and dependent variables. We might specify a linear relationship, which suggests that the dependent variable is a linear combination of the independent variables. We might specify a non-linear relationship, implying that the dependent variable is a non-linear combination of the independent variables.

LINEAR REGRESSION
DEFINITION

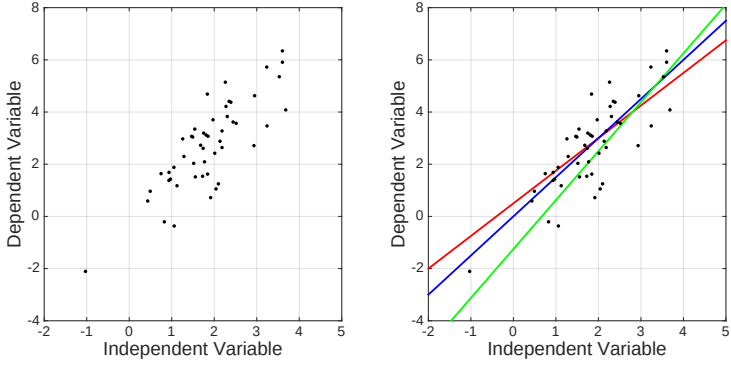
We will consider the linear case, for simplicity. We wish to find an estimate $\hat{y}_{n+1}$ of the true response y_{n+1} as a weighted linear function of the data x_{n+1}:

$$\begin{aligned}\hat{y}_{n+1} &= w_0 + w_1 x_{n+1,1} + w_2 x_{n+1,2} + \cdots + w_m x_{n+1,m} \\ &= w^\top [1, x_{n+1}].\end{aligned}$$

Here, $w \in \mathcal{R}^{m+1}$ is the parameter to be estimated by the input-output samples $\{x_i, y_i\}$, $i = 1, \ldots, n$. The form of this relationship suggests that the input-output mapping will be determined by a line, in one dimension, a plane, in two dimensions, or a hyperplane, for the general case. We assume that w has dimensionality $m+1$ so that the first term will be a bias of the linear variety away from the origin.

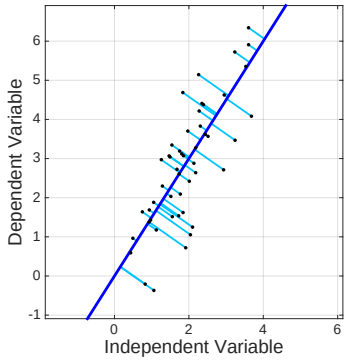
LINEAR REGRESSION

CONCEPT ILLUSTRATION



Suppose that we have the following relationship (left plot) between an independent variable and a dependent variable. We would like to define a linear function (right plot) that can be used to estimate the dependent variable for a continuum of independent variables. There are an infinite number of linear functions that can be chosen. How do we choose the 'best'?

LINEAR REGRESSION
CONCEPT ILLUSTRATION



One option is to define some notion of error. This error would measure how incorrect (light blue) a linear function's prediction (blue) would be for a known independent and dependent variable pair. We would like to consider the error not just for a single pair, but for a set of pairs, so as to ensure it fits the trend of the observations well.

LINEAR REGRESSION
CHOOSING THE REGRESSOR

For this linear model, we would like to uncover the 'best' set of parameters $w \in \mathcal{R}^{m+1}$ that match x_i with y_i for all $i = 1, \ldots, n$. Ideally, we would like the parameters to be chosen by minimizing the following average error function

$$\sum_{i=1}^{n} \left\| y_i - \sum_{j=1}^{m} w_j^* x_{i,j} \right\| = \min_w \sum_{i=1}^{n} \left\| y_i - \sum_{j=1}^{m} w_j x_{i,j} \right\|.$$

Here, we assume that $x_{1,j} = 1$, which is to handle the bias of the linear variety away from the origin.

In the event that we use the L_2 norm in the above expression, we will be minimizing the Euclidean distance between the desired response $y_i \in \mathcal{R}$ and the estimated response $\hat{y}_i \in \mathcal{R}$. This distance is referred to as the error or residual. We could use other norms beyond the L_2 norm (e.g., L_1 norm and L_∞ norm); however, the L_2 norm gives rise to simple solutions for the parameters.

ORDINARY LEAST SQUARES REGRESSION

DERIVATION

Surprisingly, we can arrive at a closed form solution for the parameters that 'best' fit the model in the L_2 sense. We do this by expanding out the L_2 norm

$$\sum_{i=1}^{n} \left\| y_i - \sum_{j=1}^{m} w_j x_{i,j} \right\|_2^2 = \left((y - Xw)^\top (y - Xw) \right)$$

$$= \left(y^\top y - w^\top X^\top y - y^\top X w + w^\top X^\top X w \right).$$

Here, we assume that $x_{1,j} = 1$, which is to handle the bias of the linear variety away from the origin. We have written the expansion in a matrix form, where $X \in \mathcal{R}^{n \times m}$ and $y \in \mathcal{R}^n$.

ORDINARY LEAST SQUARES REGRESSION

DERIVATION

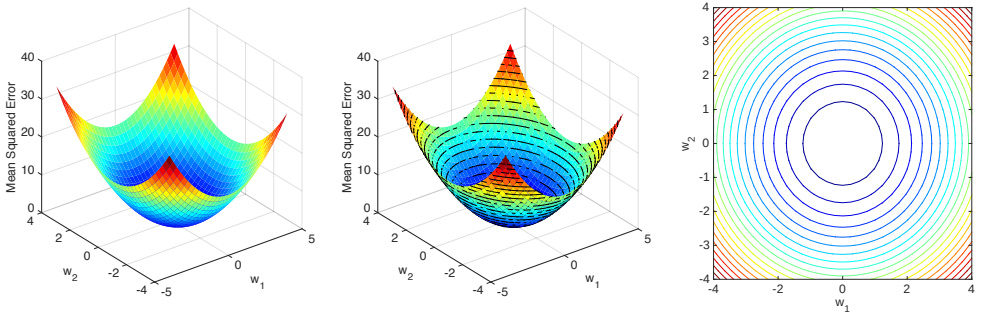
We can actually simplify this expression a bit further. Note that $(w^\top X^\top y)^\top = y^\top X w$ is a scalar and equal to its own transpose. Hence, $w^\top X^\top y = y^\top X w$, and the quantity to minimize becomes:

$$\sum_{i=1}^{n} \left\| y_i - \sum_{j=1}^{m} w_j x_{i,j} \right\|_2^2 = \left(y^\top y - 2 w^\top X^\top y + w^\top X^\top X w \right).$$

From the above expression, we can see that the error is a quadratic function of the weights. We can also see that the squared error is non-negative. The surface formed by the error must therefore be concave upwards.

ORDINARY LEAST SQUARES REGRESSION

ERROR SURFACE INSIGHTS

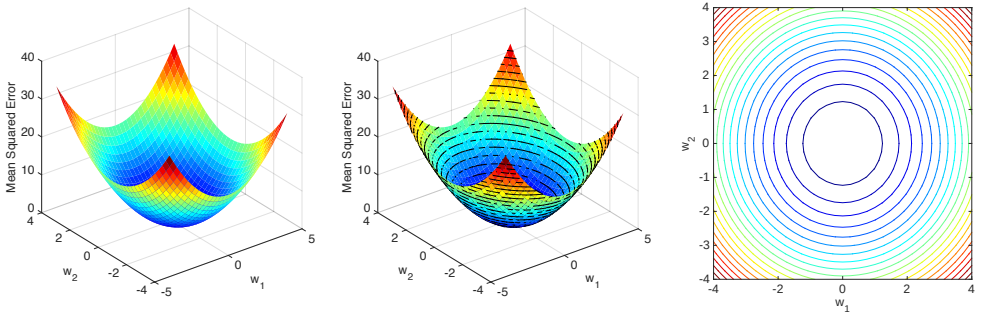


Consider a two-dimensional quadratic error surface (left plot), which is a function of two parameters $w_1, w_2 \in \mathcal{R}$. If we intersect this error surface with a series of planes parallel to the w_1-w_2 plane (middle plot), then we arrive at an elliptical contour plot of the error (right plot). The contour plot shows the error surface steepness.

Note that the error surface won't always be centered at the origin.

ORDINARY LEAST SQUARES REGRESSION

ERROR SURFACE INSIGHTS



Away from the origin, we see that there are warm-colored bands close to each other. This suggest that the error surface is very steep in these regions. As we get closer to the origin, the band spacing increases, indicating that the error surface is becoming more flat.

Since the error surface is parabolic in this example, there exists a single point (parameter solution) where the error surface attains the minimum value.

ORDINARY LEAST SQUARES REGRESSION

DERIVATION

Differentiating $y^\top y - 2w^\top X^\top y + w^\top X^\top X w$ with respect to $w \in \mathcal{R}^m$ and equating to zero yields:

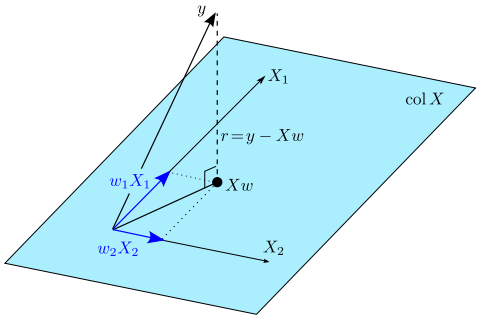
$$\frac{\partial}{\partial w}\left(y^\top y - 2w^\top X^\top y + w^\top X^\top X w\right) = -X^\top y + (X^\top X)w = 0.$$

If X has full rank, which implies that $X^\top X$ is positive definite, then the inverse of $X^\top X$ exists. We can therefore compute the ordinary least squares (OLS) solution: $w = (X^\top X)^{-1} X^\top y$. The OLS solution is optimal: there is no better solution for all of these assumptions.

For certain datasets, it may not be computationally efficient to invert $X^\top X$. We can instead use a Cholesky decomposition, which represents $X^\top X = R^\top R$ using an upper-triangular matrix $R \in \mathcal{R}^{m \times m}$. We can then use a two-step procedure to find w: first we perform a forward substitution and solve for z: $R^\top z = X^\top y$, then we perform a backward substitution to solve for w: $Rw = z$.

ORDINARY LEAST SQUARES REGRESSION

GEOMETRIC INTERPRETATION

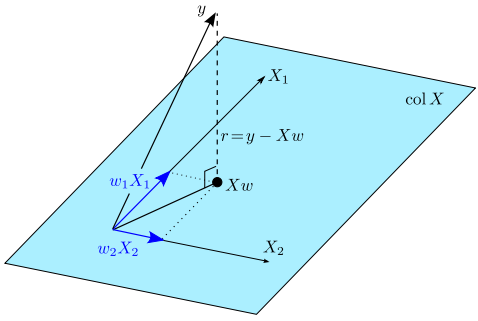


We have a geometric interpretation for the ordinary least-squares solution $w = (X^\top X)^{-1} X^\top y$. The vector $y \in \mathcal{R}^n$ of dependent variables is a single point in an n-dimensional space. If we vary the value of the parameter $w \in \mathcal{R}^m$, the product Xw describes an m-dimensional subspace of $\mathcal{R}^n$. This defines an m-dimensional hyperplane through the origin.

In the plot on the left, we consider the two-dimensional case of this column space.

ORDINARY LEAST SQUARES REGRESSION

GEOMETRIC INTERPRETATION



An obvious way to estimate w^* is to make Xw minimize y on this hyperplane. Choosing the L_2 norm to yield our metric in $\mathcal{R}^n$ implies that we are performing a projection of y onto this hyperplane. In particular, the ordinary least-squares estimator is characterized by the property that the vector of residuals $r = y - Xw$ is orthogonal to the column space of X: $r^\top X = 0$. This means that the residuals are minimizers for linear regression.

Intuitively, any non-orthogonal projection would increase the residual magnitude (or distance to the hyperplane). Therefore, the parameters would be non-optimal.

WIENER-HOPF LINEAR REGRESSION

DERIVATION

As we noted, regression can also be applied through time, wherein we predict the future from the previous observations. We can arrive at an equivalent formulation to the ordinary least squares solution, which is referred to as the Wiener-Hopf solution.

Suppose that we are considering a time-varying signal, where x_i represents an input response, y_i is the desired output, and e_i is the error between the output and desired response at time $i = 1, \dots, n$. If x_i, y_i, and e_i are stationary random processes, i.e., their values are random across time, but with unchanging mean and variance, then the average squared error is:

$$\begin{aligned} E[e_i^2] &= E[y_i^2] - 2E[y_i x_i^\top]w + w^\top E[x_i x_i^\top]w \\ &= E[y_i^2] - 2p^\top w + w^\top R w. \end{aligned}$$

Here, $p = E[y_i x_i]$ and $R = E[x_i x_i^\top]$. We can see clearly now that $E[e_i^2]$ is a quadratic function of the weights when the input is stationary.

WIENER-HOPF LINEAR REGRESSION

DERIVATION

How can we find the minimum of $E[y_i^2] - 2p^\top w + w^\top R w$? We simply compute the gradient as before and set it to zero:

$$\frac{\partial}{\partial w}\left(E[y_i^2] - 2p^\top w + w^\top R w \right) = -2p + Rw + w^\top R = 0.$$

Here, we have that $w^\top R = Rw$ due to properties of the inner product. We therefore arrive at $-2p + 2Rw = 0$. Solving for w yields: $w = R^{-1}p$. As before, this solution for w is optimal for the assumptions that we have made.

WIENER-HOPF LINEAR REGRESSION

INSIGHTS

There is an interesting interpretation for the variables $R = E[x_i x_i^\top]$ and $p = E[y_i x_i]$ that appear in the parameter solution. The matrix R can be viewed as an auto-correlation matrix. It measures the similarity of the input signal over time and defines the error surface shape. Diagonal entries of the auto-correlation matrix are the input signal variances: they are the sum of the squares of the input samples. The off-diagonal entries are cross-product sums for every possible combination of input observations.

The vector p is the cross-correlation of the input and the desired response. The entries of this vector are measures of similarity between the input and the desired response as a function of the lag of one relative to the other.

WIENER-HOPF LINEAR REGRESSION

DERIVATION

The value of the error at the optimal solution w^* can be easily computed:

$$\begin{aligned} E[y_i^2] + (w^*)^\top R w^* - 2p^\top w^* &= E[y_i^2] + (R^{-1}p)^\top R R^{-1} p - 2p^\top R^{-1} p \\ &= E[y_i^2] + p^\top R^{-1} p - 2p^\top R^{-1} p \\ &= E[y_i^2] - p^\top R^{-1} p \end{aligned}$$

which is nothing more than $E[y_i^2] - p^\top w^*$. Therefore, the solution depends on the energy of y_i along with the cross-correlation and auto-correlation of the input signal x_i.

WIENER-HOPF LINEAR REGRESSION

INSIGHTS

If w is not exactly w^*, e.g., $w = w^* + v$, then we can derive the penalty in performance

$$E[y_i^2] - (w^* + v)^\top R(w^* + v) - 2p^\top(w^* + v) = E[y_i^2] - p^\top R^{-1} p + v^\top R v.$$

This means that the excess square error is a quadratic function of the deviation of the weight parameters and depends only on the input signal statistics.

From the error, since it is non-negative, we can conclude that $v^\top R v \geq 0$. This means that R must be positive semi-definite. If R does not have full rank, then it is singular. Therefore, R^{-1} does not exist and we have more than one solution. The existence of more than one solution indicates that error surface minimum forms a hyperline or hyperplane.

GRADIENT-BASED LEAST SQUARES REGRESSION

OVERVIEW

The ordinary least squares solution is analytic, which is beneficial whenever the auto-correlation matrix is non-singular. That is, we have a simple, single-step operation that can be used to determine the optimal solution for the regression weights given the stated assumptions.

However, when the auto-correlation matrix is singular, we cannot rely on the analytic expression. We can instead use gradient-based techniques that iteratively construct solutions:

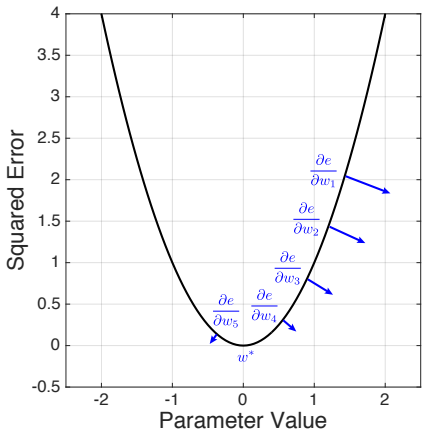
Steepest Descent: When performing minimization, the solution iterates are in the direction of the negative gradient of the performance surface. For a parabolic error surface, the negative gradient points to the minimum only along the principal axis of the parabola.

Newton's Method: A second-order version of steepest descent. It utilizes curvature information about the parabola (or locally parabolic neighborhood) to ensure that the changes are always in the direction of the minimum of the error surface.

These techniques avoid direct knowledge of the inverse auto-correlation matrix.

GRADIENT-BASED LEAST SQUARES REGRESSION

BASIC IDEA

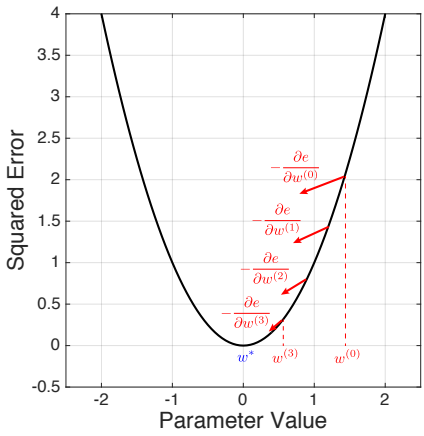


Recall that the gradient of a function $e : \mathcal{R} \rightarrow \mathcal{R}$ at a point $w_1, w_2, \ldots$ gives a direction that maximizes that function. In one dimension, it is the slope of the tangent line of the graph of the function. We can plot the gradient against the function to get a sense for its direction and magnitude.

Along the steeper portions of function, the gradient will have a higher magnitude, as the slope is greater. Flatter regions of the function will have a gradient with a lower magnitude, as the slope approaches zero.

GRADIENT-BASED LEAST SQUARES REGRESSION

BASIC IDEA



If we want to minimize a function, we will need to consider the negative gradient. We will then need an update equation that modifies the current solution iterate $w^{(i)}$, $i = 0, 1, ...$, by the error surface gradient at that point. If we update the solution in this fashion, it will converge, under some relatively mild conditions, to the optimal solution w^* for this parabolic function.

For linear regression, we are fortunate that we are dealing with a convex error surface. Many problems have highly non-convex error surfaces, which complicates the search process for the best solution.

GRADIENT-BASED LEAST SQUARES REGRESSION

DERIVATION

Our aim is to find the best solution w^*. To do this, we will start with an initial guess $w^{(0)}$ of the solution. This guess can be virtually anything that makes sense for the application. In the univariate case, we might set $w^{(0)}$ to be a random number. In the multi-variate case, we might set $w^{(0)}$ to be a vector of random numbers. Each initial guess starts the search in a different portion of the parameter search space; some starting locations may be closer to w^* than others.

We then measure the gradient at $w^{(0)}$. The gradient points in the direction of increasing error. We therefore move in the surface by a quantity proportional to the negative gradient, which should decrease the error. This yields a new solution iterate $w^{(1)}$. This process is then repeated until the solution iterates $w^{(2)}$, $w^{(3)}$, ... reach a small neighborhood around w^*.

GRADIENT-BASED LEAST SQUARES REGRESSION

DERIVATION

Formally, our aim for multivariate linear regression is to perform the following process:

$$w^{(t+1)} = w^{(t)} - \alpha \frac{\partial}{\partial w} \sum_{i=1}^{n} \left\| y_i - \sum_{j=1}^{m} w_j^{(t)} x_{i,j} \right\|_2^2$$

Here, the parameter $\alpha \in \mathcal{R}_+$ is the step-size. We may have $\alpha \in \mathcal{R}_+^m$ if we want different step-sizes for each parameter component. In either case, it controls how much the solutions actually move in the direction opposite of the gradient. Setting α to a small number means that the solutions take small steps in the search space, which may require a great many iterations to arrive at a good final result. Setting α to a large number means that we take large steps, which may lead to intermediate solutions that diverge. Finding good values for α may require significant trial and error.

CHOOSING THE LINEAR REGRESSOR

GRADIENT-BASED LEAST SQUARES

We now can take the derivative of the error expression $(w^\top x_i - y_i)^2$ with respect to w, which yields the following gradient descent update:

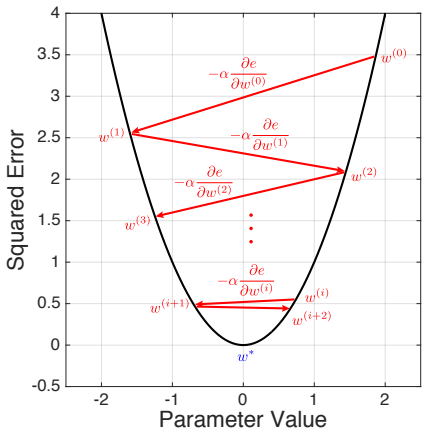
$$w^{(t+1)} = w^{(t)} - \frac{\alpha}{2}\left(\sum_{i=1}^{n}((w^{(t)})^\top x_i - y_i)x_i\right).$$

In the above expression, the gradient is found over all input-output response pairs across each iteration. This assumes that we have a complete set of input-output response pairs for which we want to estimate parameters.

It is important to restate that we can use the gradient-based update in instances when we cannot compute the ordinary least squares solution.

GRADIENT-BASED LEAST SQUARES REGRESSION

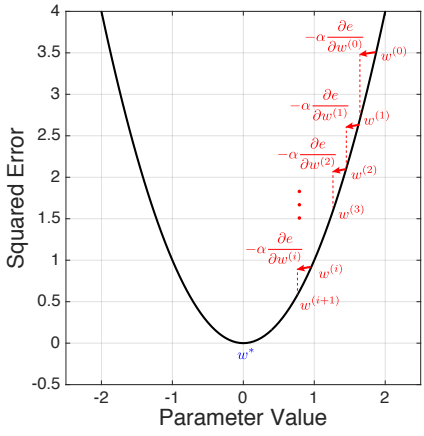
STEP SIZE INSIGHTS



Ideally we would like to use a large value for the step size to reach the solution more quickly. However, when the solution iterates $w^{(i)}$ are close to w^*, where the gradient is small but not zero, the iterative process continues to wander around a neighborhood of w^* without ever stabilizing. This phenomenon is called rattling, and the rattling basin increases proportionally to the step size. This means that when the adaptive process is stopped by an external influence (e.g., a threshold on the number of iterations), the weights may not exactly be at w^*.

GRADIENT-BASED LEAST SQUARES REGRESSION

STEP SIZE INSIGHTS



When setting the step size too small, iterates will move only a slight amount on the error surface. It can take a great many iterations before the iterates even reach w^*. However, due to the small step size, they will oscillate in a much smaller neighborhood around w^* than in the previous case. The quality of the parameters will likely be better as a consequence.

We therefore see that there is an intrinsic compromise between the accuracy of the final solution and the speed of convergence when using steepest descent.

GRADIENT-BASED LEAST SQUARES REGRESSION

CONVERGENCE INSIGHTS: UNIVARIATE CASE

We can show that, under ideal cases, steepest descent will converge to w^* in the limit. Suppose that we have a solution iterate w which is offset from w^* as follows: $w = w^* + v$. As we saw previously, the excess square error is a quadratic function of the iterates: $v^\top R v$, or, rather, $(w - w^*)^\top R (w - w^*)$. In the univariate case, R contains a single element, which implies that the error is given by $\lambda(w - w^*)^2$, where $\lambda = r_{1,1}$.

Taking the gradient of $\lambda(w - w^*)^2$ allows us to see the effect of each iterate on the error:

$$\begin{aligned} w^{(t+1)} &= w^{(t)} - 2\lambda\alpha(w^{(t)} - w^*) \\ &= (1 - 2\lambda\alpha) w^{(t)} + 2\lambda\alpha w^*. \end{aligned}$$

This is a linear, first-order, constant-coefficient difference equation. We can therefore analyze the effect recursively.

GRADIENT-BASED LEAST SQUARES REGRESSION

CONVERGENCE INSIGHTS: UNIVARIATE CASE

Expanding out the iterate updates, we find that

$$
\begin{aligned}
w^{(1)} &= (1-2\lambda\alpha)w^{(0)} + 2\lambda\alpha w^* \\
w^{(2)} &= (1-2\lambda\alpha)((1-2\lambda\alpha)w^{(0)} + 2\lambda\alpha w^*) + 2\lambda\alpha w^* \\
&= (1-2\lambda\alpha)^2 w^{(0)} + 2\lambda\alpha w^*(1 + (1-2\lambda\alpha)) \\
w^{(k)} &= (1-2\lambda\alpha)^k w^{(0)} + \textstyle\sum_{i=1}^{k-1}(1-2\lambda\alpha)^i 2\lambda\alpha w^* \\
&= (1-2\lambda\alpha)^k w^{(0)} + 2\lambda\alpha w^*(1-(1-2\lambda\alpha)^k)/(1-(1-2\lambda\alpha)) \\
&= (1-2\lambda\alpha)^k (w^{(0)} - w^*) + w^*.
\end{aligned}
$$

We therefore see that when $k \to \infty$, $(1-2\lambda\alpha)^k(w^{(0)} - w^*) \to 0$. Therefore, $w^{(k)}$ converges to w^*. An analogous result holds in the multivariate case.

GRADIENT-BASED LEAST SQUARES REGRESSION

CONVERGENCE INSIGHTS: UNIVARIATE CASE

To meet this condition, we must have the following conditions: $|1-2\lambda\alpha| < 1$ and $0 < \alpha < \lambda^{-1}$. Here, λ has the dual interpretation as being the eigenvalue of the auto-correlation matrix, as the matrix is just a scalar.

The rate of convergence, that is, the rate at which $w^{(i)}$ approaches w^*, is dependent on the geometric ratio $\gamma = 1-2\lambda\alpha$. When $|\gamma| < 1$, the rate of convergence increases for smaller values of γ. In the instance when $\gamma = 0$, the minimum is reached in a single step. For positive values of γ, there is no oscillation (overdamped). For negative values of γ, the iterates will overshoot the minimum (underdamped). When $|\gamma| \geq 1$, there will be no convergence.

CHOOSING THE LINEAR REGRESSOR

CONVERGENCE INSIGHTS: MULTIVARIATE CASE

The multivariate case is straightforward. In this instance, the gradient is given by $2R(w^{(k)} - w^*)$, which implies that

$$w^{(t+1)} = (I - 2\alpha R)w^{(t)} + 2\alpha R w^*$$

where I is the identity matrix. We have that $w^{(t+1)}$ will dependent on $w^{(t)}$ and all other coefficients through R. In the principal coordinate system,

$$V^{(t+1)} = V^{(t)} - 2\alpha R V^{(t)} = (I - 2\alpha R) V^{(t)}$$
$$Q(V^{(t+1)})^\top = (I - 2\alpha R) Q(V^{(t)})^\top = (I - 2\alpha\Lambda)(V^{(t)})^\top.$$

Since Λ is diagonal, we can talk about individual solutions along the principal axis: $(V^{k+1})^\top = (I - 2\mu\Lambda)^k (V^{(0)})^\top$. Each of these solutions can be written separately in a vector form: $(v_i^{t+1})^\top = (1 - 2\alpha\lambda_i)(v_i^{(t)})^\top$.

CHOOSING THE LINEAR REGRESSOR

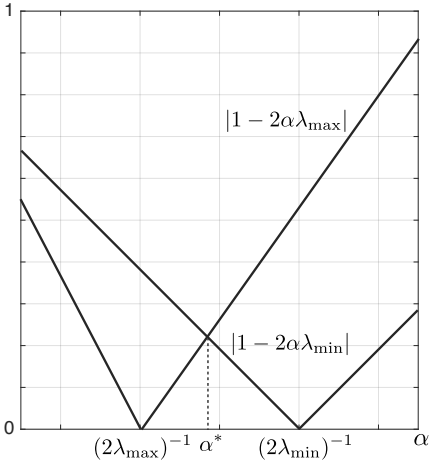
CONVERGENCE INSIGHTS: MULTIVARIATE CASE

For convergence, each $(v_i^{t+1})^\top = (1-2\alpha\lambda_i)(v_i^{(t)})^\top$ must satisfy $|1-2\alpha\lambda_i| < 1$, which implies that $0 < \alpha < \lambda_i^{-1}$. So, to have convergence in a multi-variate system, it is necessary and sufficient that $0 < \alpha < \lambda_{\text{max}}^{-1}$, where λ_{max} is the largest eigenvalue of the auto-correlation matrix. In practice, for smooth convergence, α should be two orders of magnitude smaller than $\lambda_{\max}$.

It can be shown that each $(v_i^{t+1})^\top = (1-2\alpha\lambda_i)(v_i^{(t)})^\top$ converges at different rates according to the step size α and the eigenvalue λ_i. The convergence of the entire system of equations will be determined by the smallest eigenvalue $\lambda_{\min}$ of the auto-correlation matrix: $\max_i (2\alpha\lambda_i)^{-1} = (\min_i 2\alpha\lambda_i)^{-1} = (2\alpha\lambda_{\min})^{-1}$.

CHOOSING THE LINEAR REGRESSOR

CONVERGENCE INSIGHTS: MULTIVARIATE CASE



So how do we choose the step size that provides the fastest convergence? We should look at the smallest eigenvalue of the auto-correlation matrix and add it to the largest eigenvalue: $\alpha^* = (\lambda_{\max} + \lambda_{\min})^{-1}$. For this choice, the slowest and fastest modes will converge at the same rate.

We therefore have that

$$1 - 2\alpha^*\lambda_{\min} = \frac{\lambda_{\max}/\lambda_{\min} - 1}{\lambda_{\max}/\lambda_{\min} + 1}.$$

The ratio $\lambda_{\max}/\lambda_{\min} \geq 1$ is the eigenvalue spread or condition number. The larger the spread, the slower the overall convergence.

LEAST MEAN SQUARES REGRESSION

DERIVATION

Sometimes, though, we receive observations in a streaming manner. We would like a means of estimating the instantaneous gradient for each sample. We can do this by removing the summation, then taking the derivative with respect to w:

$$\frac{\partial}{\partial w}\frac{1}{2n}\sum_{i=1}^{n}\left\| y_i - \sum_{j=1}^{m} w_j^{(t)} x_{i,j} \right\|_2^2 \approx \frac{1}{2}\frac{\partial}{\partial w}\left(y_i - \sum_{j=1}^{m} w_j^{(t)} x_{i,j} \right)^2$$

which is $-(y_i - (w^{(t)})^\top x_i)x_i$. This expression tells us that the instantaneous estimate of the gradient is simply the product of the input x_i to the weight times the error $(y_i - (w^{(t)})^\top x_i)$ at iteration t. This estimate led to the famous least means square (LMS) algorithm.

LEAST MEAN SQUARES REGRESSION

DERIVATION

When the instantaneous gradient is used in place of the total gradient, the LMS update equation for the parameters becomes

$$w^{(t+1)} = w^{(t)} + \alpha(y_i - (w^{(t)})^\top x_i)x_i.$$

It is important to note that the estimate of the gradient will be noisy, since we are using a single sample instead of summing the error for each independent variable in the dataset. Normally, many iterations are required to find the minimum in the parameter search space. The gradient noise is being averaged out during this process. Phrased another way, it is an iterative process that is improving the gradient estimate.