

From Linear Regression to Principal Component Regression/Analysis & Partial Least Squares
 -- an introductory tutorial to some of the most important ideas in multivariate regression.
 Instructor: Nam Sun Wang

Background. We are given a set of data consisted of a series of n pairs of x and y values, where x is the independent variable and y is the dependent variable.

Raw data: x_i and y_i $i=0..n$

We want to describe y with a model $f(x,a)$, where a are the adjustable model parameters.

$$y=f(x, a)$$

In reality, because of noise or mismatch between the data and the model, there is an error e .

$$y=f(x, a) + \text{error}$$

Or, equivalently in a non-vector notation,

$$y_0=f(x_0, a) + \text{error}_0$$

$$y_1=f(x_1, a) + \text{error}_1$$

:

$$y_n=f(x_n, a) + \text{error}_n$$

Or, equivalently in an index notation,

$$y_i=f(x_i, a) + \text{error}_i$$

In **regression** analysis, whether linear or nonlinear, our task is to find the regression coefficient a such that the given model function $f(x,a)$ passes through the given data (x,y) as closely as possible. That is to say that we try to minimize some sort of measure of error between y and $f(x,a)$. The error can be the (vertical) difference between the given dependent data points and the regression model. For certain applications, it is not clear which variable is an independent one and which is a dependent one. In that case, we may want to minimize the shortest distance between the given data points and the regression line. In general, this distance is neither vertical nor horizontal.

$$\text{Minimize}_{a} \text{ error}$$

The above form is not strictly valid because the error can sometimes be positive if the regression model underestimates y , or it can be negative at other values of x if the model overestimates y . There are many measures of this error. One way is to take the absolute value.

$$\text{Minimize}_{a} |\text{error}| = |y - f(x, a)|$$

Because of the ease with which mathematics can be done, the sum of squared error objective function is the most common. The algorithms that minimize the squared error are called **least squares** or **maximum likelihood estimate**. Because the error is squared, we pay a much higher penalty for an outlier in the least squares method than in the absolute error method.

$$\begin{aligned} \text{Minimize}_{\mathbf{a}} \quad \text{sse} &= \sum_{i=0}^n (\text{error}_i)^2 = \sum_{i=0}^n (y_i - f(x_i, \mathbf{a}))^2 \\ &= \text{error}^T \cdot \text{error} = (\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a}))^T \cdot (\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a})) \quad \dots \text{1} \times \text{1 vector format} \\ &= (|\text{error}|)^2 = \text{error} \cdot \text{error} = (|\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a})|)^2 = \overbrace{(\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a}))^2} \quad \dots \text{purely scalar format} \end{aligned}$$

In **linear regression**, the model function is a linear combination of m basis functions $f_j(x)$. Note that the individual basis functions are not necessarily linear with respect to the independent variable x . We only require that the combination be linear with respect to \mathbf{a} .

$$f(x) = a_0 \cdot f(x)_0 + a_1 \cdot f(x)_1 + a_2 \cdot f(x)_2 + \dots + a_m \cdot f(x)_m = \sum_{j=0}^m a_j \cdot f_{\text{basis}}(x)_m = f_{\text{basis}}(x) \cdot \mathbf{a} \quad \text{for } j=0 \dots m$$

Typically, we choose a series of **independent** basis functions based on our knowledge of the underlying mechanism. If we know *a priori* the response is periodic with respect to the independent variable x , we may naturally choose sine and cosine functions, e.g., $\sin(\omega_0 x)$, $\sin(\omega_1 x)$, etc., where $\omega_0 = 2\pi/T$, $\omega_1 = 2 * 2\pi/T$, ... are the zeros of the sine function. If the theory says the basis functions should follow Bessel functions, then we let f_j be these functions, e.g., $J_0(\lambda_0 x)$, $J_0(\lambda_1 x)$, etc. Usually, $\lambda_0, \lambda_1, \dots$ are the zeros of the Bessel functions, which is analogous to the sine functions. If we do not know much about the underlying mechanism, we may simply choose power series $1, x, x^2, x^3, \dots$, etc. to be the basis functions. Although the last approach is a very common practice among college students -- and most of the time such simple-minded regression does a good job -- it is not mathematically very rigorous, because the basis functions $1, x, x^2, x^3, \dots$, etc. are not **orthogonal** to each other. Once we learn the concept of orthogonality and the various advantages of having orthogonal basis functions over non-orthogonal ones, we tend to choose a set of orthogonal basis functions (e.g., Legendre polynomials, Laguerre polynomials, or Hermite polynomials) instead of the plain power series expansion in regression. The least squares estimate of the model parameter \mathbf{a} is given by the normal equation. Note that we can calculate the regression coefficient \mathbf{a} deterministically. There is no iteration in linear regression.

$$\mathbf{a} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}$$

where the elements of the $n \times m$ matrix \mathbf{X} (actually $(n+1) \times (m+1)$ if the index starts from 0) have the value of f_j evaluated at x_i , i.e., $X_{ij} = f_j(x_i)$.

$$X_{i,j} = f(x_i)_j$$

In summary, the steps for linear regression are:

- Step 1. Provide \mathbf{x} and \mathbf{y} vectors.
- Step 2. Provide the basis function $f_j(x)$.
- Step 3. Form the \mathbf{X} matrix: $X_{ij} = f_j(x_i)$.
- Step 4. Apply the normal equation to find \mathbf{a} : $\mathbf{a} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}$.
- Step 5. The regression equation is: $y_{\text{regress}} = f(x) \cdot \mathbf{a}$.
- Step 6. Examine the goodness of fit.

Optimal Number of Terms to Use. In practice, we may not be given the number of basis functions.

Finding out the number of terms to use (which is called the **order** or **degree**) is part of the problem to be solved in linear regression. When there are not enough terms, the regression model cannot adequately capture the general trend in the dependent variable y , and the model's predictive capability is low. This is a problem of **under-fitting**. Conversely, when there are too many terms, the regression model captures *everything* in y , including *noise* and blemish that is specific to that particular data set. This is a problem of **over-fitting**. Neither is desirable. To determine how many terms to use optimally, we start the X matrix with just one basis function, i.e., the X matrix's dimension is $n \times 1$. Carry out regression, then record some measures of goodness of fit, e.g., the sum of squared error (sse), the standard error of calibration/prediction (sec, sep), or the coefficient of correlation (the r^2 value). Add one additional term, i.e., expand the X matrix to $n \times 2$, and repeat the regression process. Continue adding one term at a time until the sse value, the sep value, or the r^2 value starts to level off. When this happens, stop. If we plot the sse value versus the number of terms employed, we typically see a sharp drop in the sse value for the first few terms. Then, the curve flattens out. Choose the number of terms near the corner of the bend in this curve.

Alternatively, in a process called **cross-validation**, we randomly split the original sample into two sets. We use the first set to build the regression model and the second set to independently test the soundness of the model. The first set is called the **training data** or **calibration data**; the second set is called the **test data** or **prediction data**. We reach a peak in the predictive power, when the regression model yields the lowest standard error of prediction. Choose the number of terms that corresponds to the lowest sep value.

In linear regression with power series ($1, x, x^2, x^3$, etc.) as the basis functions, we first try to capture the part in the original data that can be explained or are contained in the first basis function. If the first basis function is 1, we effectively take away from the original data y as much constant component as possible. The amount to take away is $a_0 \cdot 1$ or $a_0 \cdot f_0(x)$. The residual $e^{<0>}$, computed as $e^{<0>} = y - a_0 \cdot f_0(x)$, now should contain no more constant component. In the next step, we then take away as much linear component x as possible from $e^{<0>}$. A large value of a_1 associated with the basis function $f_1(x) = x$ takes away more x -dependent component from y than what the original data y contains, and a small value of a_1 will leave some x -dependent component behind. Of course, neither is desirable. The regression coefficient a_1 calculated by the normal equation tells us just how much to take away so that the residual $e^{<1>}$, computed as $e^{<1>} = e^{<0>} - a_1 \cdot f_1(x)$, should contain no more x -dependent component. We then repeat the process to take away one mathematical component at a time from y until what is left is just noise. Unfortunately, if we were to continue with the straight x^2 term, we find that some additional 1-component (i.e., constant component) may be further taken away during the process of taking away the x^2 -dependent component. This is because x^2 and 1 are not mutually orthogonal; being a symmetric function, x^2 contains some constant components. This is not a problem if we stay with an orthogonal set of basis functions. An orthogonal basis has the advantage that we can successively take away one component at a time from the residual without affecting what has been done up to that point.

This is analogous to picking out all the lettuce from a salad, then all the carrots, then all the spinach, then all the chicken meat, and so on, until we are left with just the minor ingredient or just the dust particles in a salad bowl. Another analogy is extracting different flavor or taste from a dish. We start by extracting the sugar that imparts the sweetness, then take away the salt that imparts the saltiness, then the organic acids that contribute to the bitterness, then the vinegar that causes the

sourness, and so on until what is left behind is just a mixture of bland, tasteless, odorless stuff. Another analogy is the color in a print. A printer typically makes three masks of primary subtractive colors: yellow, cyan, and magenta. In a four-color print, a fourth color, black, is used to save ink. (A light addition apparatus, such as the TV screen, is based on three additive colors: red, green, and blue.) We first find how much yellow component there is in a given color; take this component out. We then find how much cyan there is; take this out. Then magenta, until we are left with just some residual that cannot be captured with any of these three primary colors. Using a non-orthogonal set of basis functions is like picking out a brown color, which contains other colors that we have already taken out. Why should there be any residual color left? Contrary to what most of us are commonly told, these three primary colors cannot capture all the different intricate shades of a color. Thus, a print with just three primary colors can never exactly duplicate the colors in an original painting. In mathematical terms, a painting is spanned by more than just three independent basis vectors/functions.

We can tell how much has been taken out by measuring what is left behind (the residual) that have not been captured by any of the basis functions included in the model. The more components we employ, the more of the original dependent vector y we should capture. Thus, the saying "given enough components, one should be able to fit an elephant." One important consideration in regression analysis is to decide which components to employ to explain the given y . Mathematically, we can estimate the goodness of fit by examining the amount of residual, the sum of squared error (sse), the standard error or calibration/prediction (sec, sep), or the correlation coefficient. These measures of goodness of fit are closely related to each other.

Example.**Step 1.** Generate some artificial (x,y) data to demonstrate linear regression.

$$f_{\text{true}}(x) := x \cdot e^{-x} + \frac{\sin(x)}{x}$$

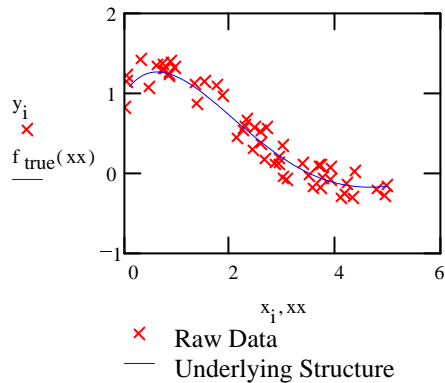
Number of points: $N := 50$ $i := 0 \dots N$

$$x_i := \text{rnd}(5) \quad y := \overrightarrow{f_{\text{true}}(x)} + (\text{rnd}(0.5) - 0.25) \quad y_i := f_{\text{true}}(x_i) + (\text{rnd}(0.5) - 0.25)$$

xx := 0.1, 0.2 .. 5

↑ Add about 10% noise to y.

↑ This form adds to y a different random number each time.

**Step 2.** Choose the power series to be the basis functions.

$$f_0(x) := 1 \quad f_1(x) := x \quad f_2(x) := x^2 \quad f_3(x) := x^3 \quad f_4(x) := x^4 \quad f_5(x) := x^5$$

$$f_6(x) := x^6 \quad f_7(x) := x^7 \quad f_8(x) := x^8 \quad f_9(x) := x^9 \quad f_{10}(x) := x^{10} \quad f_{11}(x) := x^{11}$$

$$f(x) := (f_0(x) \ f_1(x) \ f_2(x) \ f_3(x) \ f_4(x) \ f_5(x) \ f_6(x) \ f_7(x) \ f_8(x) \ f_9(x) \ f_{10}(x) \ f_{11}(x))^T$$

Step 3. Form X **Step 4.** Regression **Step 5.** Regression model.

First, employ only 1 term.

$$X_{i,0} := f(x_i)_0 \quad a := (X^T \cdot X)^{-1} \cdot X^T \cdot y \quad y_{\text{regress}}(x) := \sum_{j=0}^0 a_j \cdot f(x)_j \quad a = 0.495$$

$$sse_0 := \left(y - \overrightarrow{y_{\text{regress}}(x)} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}}(x)} \right) \quad sse_0 = 16.786$$

2 terms:

$$X_{i,1} := f(x_i)_1 \quad a := (X^T \cdot X)^{-1} \cdot X^T \cdot y \quad y_{\text{regress}}(x) := \sum_{j=0}^1 a_j \cdot f(x)_j \quad a = \begin{pmatrix} 1.434 \\ -0.379 \end{pmatrix}$$

$$sse_1 := \left(y - \overrightarrow{y_{\text{regress}}(x)} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}}(x)} \right) \quad sse_1 = 2.067$$

3 terms:

$$X_{1,2} := f(x_i)_2 \quad a := (X^T \cdot X)^{-1} \cdot X^T \cdot y \quad y_{\text{regress}(x)} := \sum_{j=0}^2 a_j \cdot f(x)_j \quad a = \begin{pmatrix} 1.456 \\ -0.408 \\ 0.006 \end{pmatrix}$$

$$sse_2 := \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \quad sse_2 = 2.06$$

4 terms:

$$X_{1,3} := f(x_i)_3 \quad a := (X^T \cdot X)^{-1} \cdot X^T \cdot y \quad y_{\text{regress}(x)} := \sum_{j=0}^3 a_j \cdot f(x)_j \quad a = \begin{bmatrix} 1.154 \\ 0.402 \\ -0.404 \\ 0.055 \end{bmatrix}$$

$$sse_3 := \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \quad sse_3 = 1.1$$

5 terms:

$$X_{1,4} := f(x_i)_4 \quad a := (X^T \cdot X)^{-1} \cdot X^T \cdot y \quad y_{\text{regress}(x)} := \sum_{j=0}^4 a_j \cdot f(x)_j \quad a = \begin{bmatrix} 1.038 \\ 0.949 \\ -0.917 \\ 0.217 \\ -0.016 \end{bmatrix}$$

$$sse_4 := \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \quad sse_4 = 0.923$$

6 terms:

$$X_{1,5} := f(x_i)_5 \quad a := (X^T \cdot X)^{-1} \cdot X^T \cdot y \quad y_{\text{regress}(x)} := \sum_{j=0}^5 a_j \cdot f(x)_j \quad a = \begin{bmatrix} 1.045 \\ 0.878 \\ -0.802 \\ 0.153 \\ -0.001 \\ -0.001 \end{bmatrix}$$

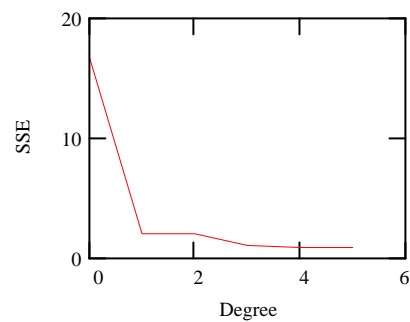
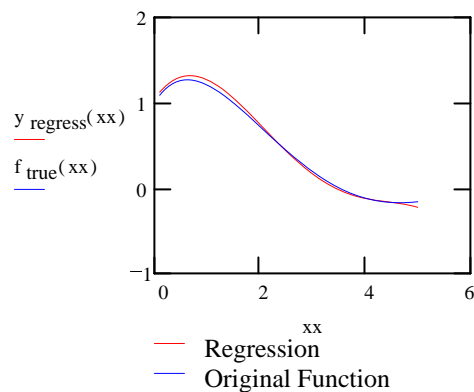
$$sse_5 := \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}(x)}} \right) \quad sse_5 = 0.922$$

Step 6. Examine sse. There is not much change in the sse value after about 4 terms.

$sse_{\text{old}} := y \cdot y$... variation in the original y

$$r2 := \frac{sse_{\text{old}} - sse_4}{sse_{\text{old}}} \quad r2 = 96.846\% \quad \sqrt{r2} = 98.41\%$$

$j := 0..5$



Re-work the problem without having to do regression step-by-step manually.

Number of basis functions: $m := 12$ $j := 0..11$

Step 3. Form X

$$X_{i,j} := f(x_i)_j$$

Step 4. Regression

$$a := 0 \quad a^{<j>} := \left(\text{submatrix}(X, 0, N, 0, j)^T \cdot \text{submatrix}(X, 0, N, 0, j) \right)^{-1} \cdot \text{submatrix}(X, 0, N, 0, j)^T \cdot y$$

Step 5. Regression model.

$$y_{\text{regress}}(x, \text{degree}) := \sum_{j=0}^{\text{degree}} (a^{<\text{degree}>})_j \cdot f(x)_j$$

Step 6. Goodness of fit.

$$\text{sse}(\text{degree}) := \left(y - \overrightarrow{y_{\text{regress}}(x, \text{degree})} \right) \cdot \left(y - \overrightarrow{y_{\text{regress}}(x, \text{degree})} \right)$$

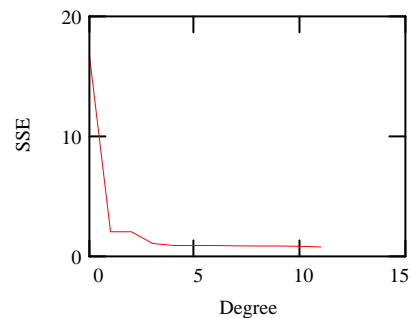
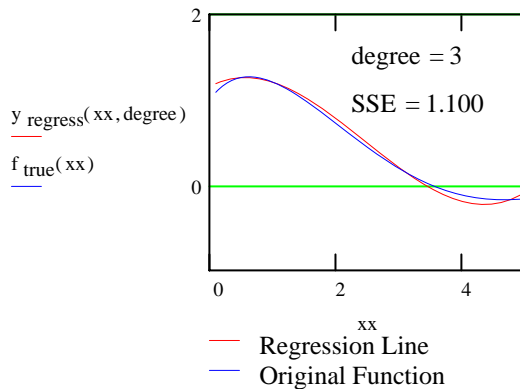
Animation Section

degree := 3 FRAME := degree degree := FRAME
SSE := sse(degree)

Toggle off "FRAME:=degree" and animate for FRAME=0..11. To play a pre-recorded movie, click on the icon.



pca.avi



Weight.

Since different experimental points may have different uncertainties, we try to weigh each point inversely proportional to the uncertainty associated with that point. The problem statement is:

$$\begin{aligned} \text{Minimize}_{\mathbf{a}} \quad \text{sse} &= \sum_{i=0}^n w_i \cdot (\text{error}_i)^2 = \sum_{i=0}^n w_i \cdot (y_i - f(x_i, \mathbf{a}))^2 \\ &= \mathbf{e}^T \cdot \mathbf{W} \cdot \mathbf{e} = (\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a}))^T \cdot \mathbf{W} \cdot (\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a})) \quad \dots \text{1x1 vector format} \\ &= \left(\left| \sqrt{\mathbf{w}} \cdot \mathbf{e} \right| \right)^2 = \mathbf{w} \cdot \mathbf{e} \cdot \mathbf{e} = \left[\left| \sqrt{\mathbf{w}} \cdot (\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a})) \right| \right]^2 = \overrightarrow{\left[\mathbf{w} \cdot (\mathbf{y} - \mathbf{f}(\mathbf{x}, \mathbf{a}))^2 \right]} \quad \dots \text{purely scalar format} \end{aligned}$$

The optimal value of \mathbf{a} is:

$$\mathbf{a} = \left(\mathbf{X}^T \cdot \mathbf{W} \cdot \mathbf{X} \right)^{-1} \cdot \mathbf{X}^T \cdot \mathbf{W} \cdot \mathbf{y}$$