OLS in 2 dimensions - correlation

Steven Brawer
11/2017

introduction

In this document we present some results for OLS (Ordinary Least Squares) for two correlated features. We consider the same variances as in a companion document *OLS in one dimension-2,* referred to as **OLS1D**. The approach and notation here are similar to that document, so it is probably a good idea to at least look at the first few pages of OLS1D. The results here are similar but generalized to two correlated features.

Of particular interest is the case of fairly large correlation coefficient, greater than 0.7 or so in magnitude. In this case, the two features are so highly correlated that, in some situations, the question of which is most important becomes an invalid question. In addition, for larger correlation coefficient, estimates of slopes can be unstable (in the sense of higher variances of the estimated slopes).

We present several changes of independent variables to orthogonal features (ie, zero correlation coefficient).

The case of two correlated features is the simplest introduction to correlation. In a later paper, we will extend the analysis to any number of strongly correlated features.

This document presents theoretical results, and a numerical study at the end.

notation

First assume that we have p features. We will specialize to p = 2 later. This document makes use of matrix and vector notation, in contrast to OLS1D.

Let **N** be the number of samples in the training set. **Samples from the training set** are labeled by letters i,j,k = 1,2,...,N. The p features for sample i are indicated by the p-dimensional vector

$$x_i = (x_i^1, x_i^2, .., x_i^p) \qquad , \quad \text{i = 1,2,...,N} \tag{1}$$

so that x_i^q is the numerical value of the qth feature of the ith training-set sample. We assume the values form a continuum. Similarly we can write the N-dimensional vector of the qth feature

$$x^q = (x_1^q, x_2^q, ..., x_N^q) \tag{2}$$

All p features of N training set samples may be represented by a matrix X (matrices will be indicated by capital letters). The quantity

$$
\begin{aligned}
X \equiv [& \\
& [x_1^1, x_1^2, ..., x_1^p], \\
& [x_2^1, x_2^2, ..., x_2^p] \\
& ..... \\
& [x_N^1, x_N^2, ..., x_N^p] \\
&]
\end{aligned}
\tag{3}
$$

is the matrix of features, where

$$X_{i,q} = x_i^q \tag{4}$$

The training set consists of N samples out of a possible infinity of samples. It is useful to have a notation indicating a single sample which may or may not be in the training set. Such a sample is labeled by Greek letters, so

$$x_\mu = (x_\mu^1, x_\mu^2, ..., x_\mu^p) \tag{5}$$

indicates a sample which may or may not be in the training set. If this sample is in the training set, then

$$x_\mu = x_i \tag{6}$$

for some i between 1 and N.

Note that x^q is always an N-dimensional vector, representing all sample values of the qth feature of the training set. On the other hand, the notation x_μ^q indicates the qth feature of a single sample which may or may not be in the training set, and the same for $X_{\mu,q}$.

The **output vector** y for all training-set samples is an N-dimensional vector

$$y = (y_1, y_2, ..., y_N) \tag{7}$$

where y_i is the output of the ith sample, and is a number. Each value is continuous. Similarly, y_μ is the output for one sample which may or may not be in the training set. So a y without a subscript indicates the vector (7).

It is convenient to have a notation for sums over training-set samples. For any N-dimensional vector

$$v \equiv (v_1, v_2, ..., v_N)$$

define

$$< v >\equiv \frac{1}{N}\sum_{i=1}^{N} v_i \tag{8}$$

For example, from eq (2),

$$< x^q >= \frac{1}{N}\sum_{i=1}^{N} x_i^q$$

The **dot product** is, for two N-dimensional vectors v, w,

$$< vw >\equiv \frac{1}{N}\sum_{i=1}^{N} v_i w_i \tag{9}$$

Similarly for the output

$$< y >\equiv \frac{1}{N}\sum_{i} y_i \tag{10}$$

OLS equations and solution

First we give the equations for any number of features, and then specialize to two features. We assume linearity, so there is no bias. The output for any sample is assumed to be *generated* from the following equation

$$y_\mu = \sum_{q=1}^{p} b_q x_\mu^q + \varepsilon_\mu \tag{11}$$

The *true* **slope b** is a p-dimensional vector, and x_μ^q is a sample which may or may not be in the training set. If it is in the training set, we would have (6) for some i. The residuals ε_μ are iid, all drawn from the same distribution, and all have the same variance. A residual from the training set would be indicated by

$$\varepsilon_\mu = \varepsilon_i$$

for some i.

For samples in the training set, eq (11) (for $\mu \to i = 1,2,...,N$) generates **data**.

If E[...] represents the ensemble average (see OLS1D)

$$E\left[\varepsilon_\mu\right]=0$$
$$E\left[\varepsilon_\mu\varepsilon_{\mu'}\right]=\begin{cases}\sigma^2 & if\ \mu=\mu' \\ 0 & otherwise\end{cases} \tag{12}$$

We assume **all training set features are standardized**, which means mean 0 and standard deviation 1:

$$< x^q >= \frac{1}{N}\sum_i x_i^q = 0$$
$$< x^q x^q >= \frac{1}{N}\sum_i \left(x_i^q\right)^2 = 1 \tag{13}$$

The **regression function** for any particular sample is

$$\hat{y}_\mu = f(x_\mu) = \alpha + \sum_q \beta_q x_\mu^q \tag{14}$$

The quantity $\hat{y}_\mu$ is the **estimated output,** while y_μ, eq (11), is the true output.

The **OLS equations** refer to the training set and are

$$\sum_i \left(y_i - \alpha - \sum_{q=1}^{p} \beta_q x_i^q \right) = 0 \tag{15}$$

$$\sum_i x_i^q \left(y_i - \alpha - \sum_{q'=1}^{p} \beta_{q'} x_i^{q'} \right) = 0 \tag{16}$$

Note that eq (16) is in fact p equations, for q = 1,2,...,p.

Now specialize to p = 2. From (11) and (10)

$$< y >= \frac{1}{N}\sum_i \sum_{q=1}^{2} b_q x_i^q + \frac{1}{N}\sum_i \varepsilon_i$$

Because of first of eqs (13), the first term above is 0, so

$$< y >= \frac{1}{N}\sum_i \varepsilon_i \tag{17}$$

The solution to (15) is

$$\alpha =< y > \tag{18}$$

Define the correlation coefficient

$$\rho_{q,q'} \equiv< x^q x^{q'} >= \frac{1}{N}\sum_i x_i^q x_i^{q'} \tag{19}$$

which is the usual definition because of eqs (13). Note that

$$\rho_{q,q} = 1 \tag{20}$$

and since p = 2 we can drop the subscripts to define

$$\rho \equiv \rho_{1,2} \tag{21}$$

Define

$$z_i \equiv y_i - < y > \tag{22}$$

so z without subscript is the vector

$$z = (z_1, z_2, ..., z_N)$$

Then

$$\rho_{z,q} \equiv< z x^q >= \frac{1}{N}\sum_i z_i x_i^q \tag{23}$$

Because of the first of eqs (13)

$$\rho_{z,q} =< y x^q > \tag{24}$$

Eq (16) becomes

$$\rho_{z,q} - \sum_{q'=1}^{p} \beta_{q'} \rho_{q,q'} = 0 \tag{25}$$

Writing out the solutions (simple enough in 2 dimensions)

$$\beta_1 = \frac{\rho_{z,1} - \rho\rho_{z,2}}{1-\rho^2}$$
$$\beta_2 = \frac{\rho_{z,2} - \rho\rho_{z,1}}{1-\rho^2} \tag{26}$$

Writing everything out

$$\rho_{z,1} = \frac{1}{N}\sum_i x_i^1 \varepsilon_i + b_1 + \rho b_2$$
$$\rho_{z,2} = \frac{1}{N}\sum_i x_i^2 \varepsilon_i + \rho b_1 + b_2 \tag{27}$$

Therefore, the estimates for the slopes are

$$\beta_1 = b_1 + \frac{1}{N\left(1-\rho^2\right)}\sum_i \varepsilon_i \left(x_i^1 - \rho x_i^2\right)$$
$$\beta_2 = b_2 + \frac{1}{N\left(1-\rho^2\right)}\sum_i \varepsilon_i \left(x_i^2 - \rho x_i^1\right) \tag{28}$$

Eqs (18) and (28) are the general solutions to the OLS equations in two dimensions, when the samples are standardized (eqs (13)).

Eqs (28) give the **estimated slopes**. We should really write $\hat{\beta}$, but since β is always used for the estimated slope, no confusion should result.

One curious aspect of eqs (28) is that

$$< x^2 (x^1 - \rho x^2) >= 0$$
$$< x^1 (x^2 - \rho x^1) >= 0 \tag{29}$$

We will see that these relations appear elsewhere (eqs (58) and (60) below).

one-feature fit to the data

A preliminary approach to evaluating the importance of a feature might be to solve two one-dimensional problems, one for each feature. That is, instead of eq (14), we solve the one-dimension problem with regression function

$$\hat{y}_\mu = g(x_\mu) = \alpha + \gamma_q x_\mu^q \tag{30}$$

for q = 1 and, separately, q = 2. One might try this approach to determine which feature is most important in determining the output. We might suspect that if the estimated slope in one case

is large and the other small, than the feature with the large slope might be interpreted as being more important

The solution to (30) is simply

$$\gamma_q = \rho_{z,q}$$

From eq (27) we see that

$$\begin{aligned} E[\gamma_1] &= b_1 + \rho b_2 \\ E[\gamma_2] &= b_2 + \rho b_1 \end{aligned} \tag{31}$$

Each of (31) is a separate solution to a one-feature assumption (30). We see that for larger correlation, the slope combines both $b_1 \; and \; b_2$ almost equally weighted. So, for example, if q = 1, and suppose $b_1 \ll b_2$, then if the correlation coefficient is large, the estimated slope for x^1 (for the one-feature approach) is still large, similar to that for x^2 . So the result (31) may be misleading.

variance of slopes

From (28) we see that

$$E[\beta_q] = b_q \tag{32}$$

In other words, the average slope is just the true slope in this simple case where the data itself , eq (11), is linear. Defining

$$\Delta\beta_q \equiv \beta_q - b_q \tag{33}$$

It is simple enough to calculate, from eq (28):

$$E[\Delta\beta_q{}^2] = \frac{\sigma^2}{N\left(1-\rho^2\right)} \tag{34}$$

The correlation coefficient of the two slopes is

$$\frac{E[\Delta\beta_1 \Delta\beta_2]}{\sqrt{E[\Delta\beta_1{}^2]E[\Delta\beta_2{}^2]}} = -\rho \tag{35}$$

If the correlation is large and N not too large, then the variance eq (34) can be fairly large even if σ^2 itself is not too big. In other words, estimates of the slopes can be unstable and unreliable. Because of eq (35), for large correlation, the large variance affects both slopes at once.

variance of yhat - E[yhat]

The estimated value for the output, eq (14), is

$$\widehat{y_\mu} =< y > +\beta_1 x_\mu^1 + \beta_2 x_\mu^2 \tag{36}$$

From eq (17), E[<y>]=0, so

$$E[\widehat{y_\mu}] = b_1 x_\mu^1 + b_2 x_\mu^2 \tag{37}$$

Defining

$$D\widehat{y_\mu} \equiv \widehat{y_\mu} - E[\widehat{y_\mu}] \tag{38}$$

We have

$$D\widehat{y_\mu} =< y > +\Delta\beta_1 x_\mu^1 + \Delta\beta_2 x_\mu^2 \tag{39}$$

$$E[D\widehat{y_\mu}^2] = E[< y >^2] + \left(x_\mu^1\right)^2 E[\Delta\beta_1{}^2] + \left(x_\mu^2\right)^2 E[\Delta\beta_2{}^2] + 2x_\mu^1 x_\mu^2 E[\Delta\beta_1 \Delta\beta_2] \tag{40}$$

where the other cross-terms vanish. Using previous results, we find

$$E[D\widehat{y_\mu}^2] = \frac{\sigma^2}{N}\left[1 + \frac{1}{1-\rho^2}\left(\left(x_\mu^1\right)^2 + \left(x_\mu^2\right)^2 - 2\rho x_\mu^1 x_\mu^2\right)\right] \tag{41}$$

As in the case of variance of the estimated slopes, if N is not too large and the correlation is large, this variance can be large.

Note that, for samples in the training set

$$\frac{1}{N}\sum_i E[D\widehat{y_i}^2] = \frac{3\ \sigma^2}{N} \tag{42}$$

This relation generalizes to greater number of features (higher dimensions), so that instead of 3 there would be the factor p+1.

variance of yhat - ytrue

Define

$$\Delta y_\mu \equiv \widehat{y_\mu} - y_\mu \tag{43}$$

Note that

$$E[\widehat{y_\mu}] = E[y_\mu] = b_1 x_\mu^1 + b_2 x_\mu^2 \tag{44}$$

which holds both for training set samples or non-training set samples. Then

$$\Delta y_\mu = < y > + \Delta\beta_1 x_\mu^1 + \Delta\beta_2 x_\mu^2 - \varepsilon_\mu \tag{45}$$

which is, from (28)

$$\Delta y_\mu = \frac{1}{N}\sum_i \left(\varepsilon_i - \varepsilon_\mu\right) + \frac{1}{N\left(1-\rho^2\right)}\left[x_\mu^1 \sum_i \left(x_i^1 - \rho x_i^2\right) + x_\mu^2 \sum_i \left(x_i^2 - \rho x_i^1\right) \right] \tag{46}$$

From this it is easily seen that, if $\mu = k$, so the **sample is in the training set,**

$$E[\Delta y_k{}^2] = \sigma^2\left(1 - \frac{1}{N}\right) - \frac{\sigma^2}{N\left(1-\rho^2\right)}\left[\left(x_k^1\right)^2 + \left(x_k^2\right)^2 - 2\rho x_k^1 x_k^2\right] \tag{47}$$

$$\frac{1}{N}\sum_k E[\Delta y_k{}^2] = \sigma^2\left(1 - \frac{3}{N}\right) \tag{48}$$

Because the variance must be non-negative, we see that at least 3 samples are required. The reason is that the OLS equations are in fact 3 equations: eq (15) is one equation, but eq (16) is actually two equations, for q = 1,2.

For a **sample not in the training set,** ie, $\mu \neq any\ k$, eq (46) gives

$$E[\Delta y_\mu{}^2] = \sigma^2\left(1 + \frac{1}{N}\right) + \frac{\sigma^2}{N\left(1-\rho^2\right)}\left[\left(x_\mu^1\right)^2 + \left(x_\mu^2\right)^2 - 2\rho x_\mu^1 x_\mu^2\right] \tag{49}$$

Note the difference in signs between eqs (47) and (49). This type of result is discussed extensively in OLS1D, where the difference in signs between (47) and (49) is due to the independence of residuals, eqs (12). Note also the effect of a large correlation.

Of special interest here is the case where x^1_μ has a different sign than x^2_μ. This gives the largest effect. The reason is that the slope of the line is affected most by large values of |x|, as discussed in OLS1D. If the correlation is large and positive, both ends of the line are affected in a correlated manner.

In the next sections, we get back to trying to learn **which feature (if any) is most important**.

change of variables - symmetric

We have the identity

$$b_1 x^1 + b_2 x^2 = \frac{1}{2}(b_1 + b_2)(x^1 + x^2) + \frac{1}{2}(b_1 - b_2)(x^1 - x^2) \tag{50}$$

If x^1, x^2 are highly positively correlated, then the first term in (50) should be large and the second term smaller, especially if b_1 and b_2 are of comparable magnitude. This suggests that we can **change variables in the regression function:**

$$\begin{aligned} p^1 &\equiv \frac{1}{C}(x^1 + x^2) \qquad C = \sqrt{2(1+\rho)} \\ p^2 &\equiv \frac{1}{D}(x^1 - x^2) \qquad D = \sqrt{2(1-\rho)} \end{aligned} \tag{51}$$

Each p^q is an N-dimensional vector. We see that

$$< p^q p^q >= 1, \quad < p^1 p^2 >= 0 \tag{52}$$

Then the regression function (14) becomes ($p_\mu \equiv (p^1_\mu, p^1_\mu)$)

$$\hat{y}_\mu = f(p_\mu) = \alpha + \sum_q \omega_q p^q_\mu \tag{53}$$

where in (53), the estimated slopes β_q are different from (28). (Note that eq (11), which simply generates data, is unchanged.) We find from (26) (where ρ in that equation is 0, but in below it is non-zero) that

$$\begin{aligned} E[\omega_1] &= \frac{\sqrt{2(1+\rho)}}{2}(b_1 + b_2) \\ E[\omega_2] &= \frac{\sqrt{2(1-\rho)}}{2}(b_1 - b_2) \end{aligned} \tag{54}$$

Because the p's are uncorrelated (eq (52)), the variance of each is given by the same formula as in OLS1D

$$E[\Delta\omega_q{}^2] = \frac{\sigma^2}{N} \tag{55}$$

For larger correlation coefficient, this variance is much less than eq (34), meaning OLS estimates when computed using (51) and (53) will be relatively much more stable for larger correlations.

change of variables - asymmetric

There are an infinite number of ways to change variables to orthogonal coordinates p^q, such that eqs (52) hold. If we let

$$\begin{aligned} p^1 &= G(x^1 + ax^2) \\ p^2 &= H(bx^1 + x^2) \end{aligned} \tag{56}$$

where G, H, a, b are constants, then to make $< p^1 p^2 >= 0$ we solve

$$\rho(ab+1) = -(a+b) \tag{57}$$

This is one equation in two unknowns, so there are an infinite number of solutions. Once a and b are decided, then G and H can be computed to make the first of eqs (52) hold.

In the previous section, we chose a = 1, b = -1. Another solution is to set $a = 0,\ b = -\rho$, to give standardized features

$$\begin{aligned} p^1 &= x^1 \\ p^2 &= \frac{1}{\sqrt{1-\rho^2}}(\rho x^1 - x^2) \end{aligned} \tag{58}$$

The regression function is still given by eq (53), so the estimated slopes are

$$\begin{aligned} E[\omega_1] &= b_1 + \rho b_2 \\ E[\omega_2] &= -b_2\sqrt{1-\rho^2} \end{aligned} \tag{59}$$

The transformation (58) would be useful for large correlation if b_2 were small. It says that x^1 is most important if b_2 is small and ρ large.

There is a symmetric alternative to eqs (58) for standardized features

$$p^2 = x^2$$
$$p^1 = \frac{1}{\sqrt{1-\rho^2}}(\rho x^2 - x^1) \tag{60}$$

so

$$E[\omega_1] = -b_1\sqrt{1-\rho^2}$$
$$E[\omega_2] = \rho b_1 + b_2 \tag{61}$$

In both these cases, since the variables are orthogonal, the variance in the slope is given by eq (55).

a numerical study

In this example, N = 20. The standard deviation of all ε_μ, eq (11), is 0.3. The true slopes in eq (11) are $b_1 = 1, b_2 = 0.1$. The feature x^1 is generated from a square distribution and is standardized (eqs (13)). The feature x^2 is correlated to x^1 and is generated by

$$x^2 = x^1 + e$$

where e (a random variable) is an N-dimensional vector generated from a square distribution with standard deviation

$$< ee >= \sqrt{\frac{1}{\rho^2} - 1}$$

We take $\rho = 0.9$.

The value of the correlation coefficient used in calculations is

$$\rho_{calc} = \frac{< x1x2 >}{| x1 || x2 |}$$

The feature x^2 is also standardized.

We have the following results from the calculation:

rhoCalc: 0.876 exact: 0.9

The result of the OLS for x^1, x^2 *is*

$E[\beta_1] = 1.19, exact = 1.0$

$E[\beta_2] = .066, exact = 0.1$

The calculation for the reduced one-dimensional cases, eqs (30) and (31):

$E[\gamma_1] = 1.25, exact = 1.09$

$E[\gamma_2] = 1.11, exact = 1.0$

change vars to x1+x2
slope for p1: 1.22 exact: 1.066
slope for p2: 0.28 exact: 0.22

*change vars to x1, (rho*x1-x2)/sqrt(1-rho**2)*
slope for p1: 1.25 exact: 1.088
slope for p2: -0.032 exact: -0.048

*change vars to x2, (rho*x2-x1)/sqrt(1-rho**2)*
slope for p1: -0.57 exact: -0.48
slope for p2: 1.11 exact: 0.98