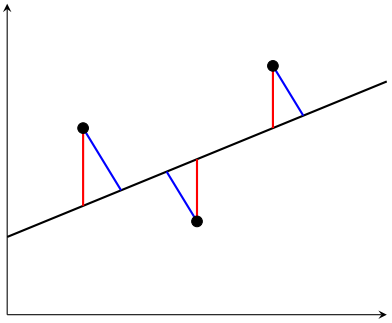
CS 189 Introduction to Machine Learning
Fall 2017 Note 8

1 Total Least Squares

Previously, we have covered **Ordinary Least Squares (OLS)**. which assumes that the dependent variable y is noisy but the independent variables x are noise-free. We now discuss **Total Least Squares** (**TLS**), which arises in the case where we assume that our x data is also corrupted by noise. Both LS methods want to get a model that produce an approximation closest to all the points, but they measure the distance differently. OLS tries to minimize the vertical distance between the fitted line and data points, while TLS tries to minimize the perpendicular distance.



The **red** line represents **vertical distance**, which OLS aims to minimize. The **blue** line represents **perpendicular distance**, which TLS aims to minimize. Note that all blue lines are perpendicular to the black line (hypothesis model), while all red lines are perpendicular to the x axis.

We might begin with a probabilistic formulation and fit the parameters via maximum likelihood estimation, as before. Suppose on the plane, we have a true model that we want to recover from some data points:

$$y_i = ax_i \tag{1}$$

and we observe data points in the form

$$(x_i + z_{xi}, y_i + z_{yi}) \tag{2}$$

where the noise terms are normally distributed, i.e. $z_{xi}, z_{yi} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$.

Applying the equation (1) that comes from the true model mentioned above, we rewrite (2) in the form $(x_i + z_{xi}, ax_i + z_{yi})$, giving

$$y = ax + \underbrace{-az_x + z_y}_{\sim\mathcal{N}(0,a^2+1)} \tag{3}$$

Given these assumptions, we can derive the likelihood for just 1 point under hypothesis a:

$$P(x_i, y_i; a) = \frac{1}{\sqrt{2\pi(a^2+1)}} \exp\left(-\frac{1}{2}\frac{(y_i - ax_i)^2}{a^2+1}\right) \tag{4}$$

Thus the log likelihood is

$$\log P(x_i, y_i; a) = \text{constant} - \frac{1}{2}\log(a^2+1) - \frac{1}{2}\frac{(y_i - ax_i)^2}{a^2+1} \tag{5}$$

Observe that a shows up in three places, unlike the form that we are familiar with, where a only appears in the quadratic term. Our usual strategy of setting the derivative equal to zero to find a maximizer will not yield a nice system of linear equations in this case, so we'll try a different approach.

1.1 Solution

To solve the TLS problem, we develop another formulation that can be solved using the singular value decomposition.

Assume we have n data points, $\vec{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}$, and we stack them to get

$$(X + \vec{e})\vec{w} = \vec{y} + \vec{f} \tag{6}$$

where $\vec{w} \in \mathbb{R}^d$ is the weight, and E and $\vec{f}$ are noise terms that we add to explain the error in the model. Our goal is to minimize the Frobenius norm[1] of the matrix composed of these error vectors. Recall that from a probabilistic perspective, finding the most likely value of a Gaussian corresponds to minimizing the squared distance from the mean. Since we assume the noise is 0-centered, we want to minimize the sum of squares of each entry in the error matrix, which corresponds exactly to minimizing the Frobenius norm. Thus we arrive at the following constrained optimization problem:

$$\min_{E,\vec{f}} \left\| \begin{bmatrix} E & \vec{f} \end{bmatrix} \right\|_F^2 \quad \text{subject to} \quad (X+E)\vec{w} = \vec{y} + \vec{f}$$

In order to separate out the term being minimized, we rearrange the constraint equation as

$$\underbrace{\left(\begin{bmatrix} X & \vec{y} \end{bmatrix} + \begin{bmatrix} E & \vec{f} \end{bmatrix}\right)}_{\in \mathbb{R}^{n \times (d+1)}} \begin{bmatrix} \vec{w} \\ -1 \end{bmatrix} = 0 \tag{7}$$

In linear algebraic terms, this expression says that the vector $(\vec{w}, -1)$ lies in the nullspace of the matrix on the left. Note that $\begin{bmatrix} X & \vec{y} \end{bmatrix}$ would not be full rank if we observe the data with no noise, since we would have $\vec{y} = X\vec{w}$, which implies that the columns are linearly dependent. But the observations we get have noise, which makes the training data matrix full rank. To compensate, we must add something to it so that it loses rank, since otherwise the nullspace is just $\{\vec{0}\}$ and the

[1] Recall that the Frobenius norm is like the standard Euclidean norm but applied to the elements of a matrix instead of a vector:

$$\|A\|_F^2 = \sum_i \sum_j A_{ij}^2$$

equation cannot be solved. We use the SVD coordinate system to achieve this:

$$\begin{bmatrix} X & \vec{y} \end{bmatrix} = \begin{bmatrix} \vec{u}_1 & \dots & \vec{u}_{d+1} & | & U_{rest} \end{bmatrix} \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{d+1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \vec{v}_1^\top \\ \vdots \\ \vec{v}_{d+1}^\top \end{bmatrix} \tag{8}$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{d+1} > 0$, and U and V are orthogonal matrices. Recall that this implies that multiplication by U or V does not change the Frobenius norm, so minimizing $\| \begin{bmatrix} E & \vec{f} \end{bmatrix} \|_F^2$ is equivalent to minimizing $\| \begin{bmatrix} E' & \vec{f}' \end{bmatrix} \|_F^2$ where $E', \vec{f}'$ are $E, \vec{f}$ expressed in the SVD coordinates. Now our problem reduces to finding $E', \vec{f}'$ such that

$$\begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_{d+1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} E' & \vec{f}' \end{bmatrix} \tag{9}$$

is not full rank and $\| \begin{bmatrix} E' & \vec{f}' \end{bmatrix} \|_F^2$ is as small as possible. Since the matrix on the left is diagonal, we can reduce its rank by simply zeroing out one of its diagonal elements. Therefore our perturbation $\begin{bmatrix} E' & \vec{f}' \end{bmatrix}$ will have $-\sigma_j$ in the (j, j) position for some j, and zeros everywhere else. To minimize the size of the perturbation, we decide to eliminate the smallest σ_j by taking

$$\begin{bmatrix} E' & \vec{f}' \end{bmatrix} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & -\sigma_{d+1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix}$$

Such a perturbation in SVD coordinates corresponds to a perturbation of

$$\begin{bmatrix} E & \vec{f} \end{bmatrix} = \begin{bmatrix} \vec{u}_1 & \dots & \vec{u}_{d+1} & | & U_{rest} \end{bmatrix} \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & -\sigma_{d+1} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \vec{v}_1^\top \\ \vdots \\ \vec{v}_{d+1}^\top \end{bmatrix} = -\sigma_{d+1} \vec{u}_{d+1} \vec{v}_{d+1}^\top$$

in the original coordinate system. It turns out that this choice is optimal, as guaranteed by the Eckart-Young theorem, which is stated at the end for reference.

The nullspace of our resulting matrix is then

$$\text{null}\left(\begin{bmatrix} X & \vec{y} \end{bmatrix} + \begin{bmatrix} E & \vec{f} \end{bmatrix} \right) = \text{null}\left(\sum_{j=1}^{d} \sigma_j \vec{u}_j \vec{v}_j^\top \right) = \text{span}\{\vec{v}_{d+1}\}$$

where the last equality holds because $\{\vec{v}_1, \ldots, \vec{v}_{d+1}\}$ form an orthogonal basis for $\mathbb{R}^{d+1}$. To get the weight $\vec{w}$, we find a scaling α such that $(\vec{w}, -1)$ is in the nullspace, i.e.

$$\begin{bmatrix} \vec{w} \\ -1 \end{bmatrix} = \alpha \vec{v}_{d+1}$$

Once we have $\vec{v}_{d+1}$, or any scalar multiple of it, we simply rescale it so that the second component is -1, and then the first component gives us w. Since $\vec{v}_{d+1}$ is a right-singular vector of $\begin{bmatrix} X & \vec{y} \end{bmatrix}$, it is an eigenvector of the matrix

$$\begin{bmatrix} X & \vec{y} \end{bmatrix}^\top \begin{bmatrix} X & \vec{y} \end{bmatrix} = \begin{bmatrix} X^\top X & X^\top \vec{y} \\ \vec{y}^\top X & \vec{y}^\top \vec{y} \end{bmatrix} \tag{11}$$

So to find it we solve

$$\begin{bmatrix} X^\top X & X^\top \vec{y} \\ \vec{y}^\top X & \vec{y}^\top \vec{y} \end{bmatrix} \begin{bmatrix} \vec{w} \\ -1 \end{bmatrix} = \sigma_{d+1}^2 \begin{bmatrix} \vec{w} \\ -1 \end{bmatrix} \tag{12}$$

To gain an extra perspective, ignore the bottom equation (we can do this because we have an extra degree of freedom) and consider the solution of the top equation:

$$X^\top X \vec{w} - X^\top \vec{y} = \sigma_{d+1}^2 \vec{w} \tag{13}$$

which can be rewritten as

$$(X^\top X - \sigma_{d+1}^2 I)\vec{w} = X^\top \vec{y} \tag{14}$$

This result is like ridge regression, but with a *negative* regularization constant! Why does this make sense? One of the motivations of ridge regression was to ensure that the matrix being inverted is in fact nonsingular, and subtracting a scalar multiple of the identity seems like a step in the opposite direction. We can make sense of this by recalling our original model:

$$X = X_{\text{true}} + Z_x$$

where X_{true} are the actual values before noise corruption, and Z_x is a noise term. Then

$$\begin{aligned}
\mathbb{E}[X^\top X] &= \mathbb{E}[(X_{\text{true}} + Z_x)^\top (X_{\text{true}} + Z_x)] \\
&= \mathbb{E}[X_{\text{true}}^\top X_{\text{true}}] + \mathbb{E}[X_{\text{true}}^\top Z_x] + \mathbb{E}[Z_x^\top X_{\text{true}}] + \mathbb{E}[Z_x^\top Z_x] \\
&= X_{\text{true}}^\top X_{\text{true}} + X_{\text{true}}^\top \underbrace{\mathbb{E}[Z_x]}_{\vec{0}} + \underbrace{\mathbb{E}[Z_x]^\top}_{\vec{0}} X_{\text{true}} + \mathbb{E}[Z_x^\top Z_x] \\
&= X_{\text{true}} X_{\text{true}} + \mathbb{E}[Z_x^\top Z_x]
\end{aligned}$$

Observe that the off-diagonal terms of $\mathbb{E}[Z_x^\top Z_x]$ terms are zero because the ith and jth rows of Z_x are independent for $i \neq j$, and the on-diagonal terms are essentially variances. Thus the $-\sigma_{d+1}^2 I$ term is there to compensate for the extra noise introduced by our assumptions regarding the independent variables.

1.2 Eckart-Young Theorem

The Eckart-Young theorem essentially says that the best low-rank approximation (in terms of the Frobenius norm) is obtained by throwing away the smallest singular values.

Theorem. Suppose $A \in \mathbb{R}^{m \times n}$ has rank $r \leq \min(m,n)$, and let $A = U\Sigma V^\top = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^\top$ be its singular value decomposition. Then

$$A_k = \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^\top = U \begin{bmatrix} \sigma_1 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & 0 & \cdots & 0 \\ 0 & 0 & \sigma_k & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} V^\top$$

where $k \leq r$, is the best rank-k approximation to A in the sense that

$$\|A - A_k\|_F \leq \|A - \tilde{A}\|_F$$

for any $\tilde{A}$ such that $\text{rank}(\tilde{A}) \leq k$.