Loess (or Lowess) Scatterplot Smoothing

Basic idea: move a window along the x-axis of a scatterplot, calculating a "fitted value" at each position of the window. The fitted values, when connected, form the loess curve.

Definitions and notation:

Let

$x_i, y_i,\ (i = 1, \ldots, n)$ = a set of n observations of x and y;

λ = degree of the locally fitted polynomial (usually 0, 1, or 2);

α = the proportion of the n observations included in each window;

$q = \text{int}(\alpha \times n)$, the integer value of the product of α and n;

$v_i,\ (i = 1, \ldots, n)$ = m equally spaced values across the range of x (i.e.

$\Delta v = v_{j+1} - v_i = (\max(x) - \min(x))/m$;

$\Delta_i(v_j) = \left| x_i - v_j \right|$ = the distance between the *i-th* value of x and the *j-th* value of v;

$\Delta_{[i]}(v_j) = \Delta_i(v_j)$ = the i-th smallest distance (ranked from smallest to largest; i.e.

$(\Delta_{[1]}(v_1) < \Delta_{[2]}(v_2) < \ldots < \Delta_{[n]}(v_j))$;

$\Delta *_{[i]}(v_j) = (\Delta_{[i]}(v_j) / \Delta_{[q]}(v_j))$ = the distance between the *i-th* value of x and the *jth* value of v divided by $\Delta_{[q]}(v_j)$, the distance between v_j and the *q-th* farthest x_i;

$$w_i(v_j) = \begin{cases} (1 - \left| \Delta *_{[i]} v_j \right|^3)^3, & \text{if } \left| \Delta *_{[i]} v_j \right| < 1 \\ 0, & \text{otherwise} \end{cases},$$

where the $w_i(v_j)$ are known as the "tricube" weights, and

$g(v_j)$ = the loess "fitted value" at v_j.

Calculation of the loess "fitted values"

The $g(v_j)$ are defined as the predicted (or fitted) values of y at v_j, i.e.:

$$g(v_j) = \hat{y} = \sum_{k=0}^{\lambda} \hat{b}_{jk} v_j^k = \hat{b}_{j1} v_j + \hat{b}2 v_j^2 + \ldots + \hat{b}_{j\lambda} v_j^{\lambda},$$

where $\hat{b}_{jk}$, ($k = 1,...,\lambda$) are the weighted least squares (WLS) estimates that minimize the quantity

$$\sum_{i=1}^{n}\left(w_i(v_j)\left(y_i - \sum_{k=0}^{\lambda} \hat{b}_{jk} v_i^k \right)\right)^2 .$$

The optional "robustness" step:

The influence of outliers or unusual points on the loess curve can be mimimized by examining the residuals, e_i, from the fitted curve, and downweighting those that are relatively large.

Let

$e_i = y_i - \sum_{k=0}^{\lambda} \hat{b}_{jk} x_i^k$, where the $\hat{b}_{jk}$'s are those for the target point, v_j, closest to x_i.

Also define

$e_i* = \dfrac{e_i}{6\bullet\text{Median}|e_i|}$, and the associated robustness weights

$$r_i = \begin{cases} (1-|e_i *|^2)^2, \text{ if } |e_i *| < 1 \\ 0, \text{ otherwise} \end{cases}.$$

Then the robustness weights are used to estimate a new set of regression coefficients, $b_{jk}*$, which minimize

$$\sum_{i=1}^{n}\left(r_i w_i(v_j)\left(y_i - \sum_{k=0}^{\lambda} \hat{b}_{jk} * x_i^k \right)\right)^2 .$$

The new fitted values then are given by

$$g(v_j)* = \sum_{k=0}^{\lambda} \hat{b}_{jk} * v_j^k .$$

Although this approach yields discrete fitted values, the lowess "curve" can be depicted by fitting an interpolating spline function to the fitted values. This discussion is based on the one in Jacoby, W.G. (1997) *Statistical Graphics for Univariate and Bivariate Data*, Sage Publications.