**Regression**

Now we’ll extend the results of the previous file to cover multiple independent variables. Note that we can visually ascertain whether a linear regression is appropriate by separately graphing the dependent variable against each of the independent variables. If *all* the separate plots are linear, then linear regression should work.

**Multivariate Regression**

When we’re doing single variable regression, we make the fit:



where i enumeraes the rows of data. If we have multiple outputs (say two), then we could write that as:



Here, i enumerates the rows of daa, and α the column, the component. So can see we take the same row of data, **x**, and map it to two different outputs. Could write this out in matrix form:



(first column of x’s would just be 1’s in this context) Sklearn’s linear regression can handle this setup. It will return the matrix of coefficients. Let’s look at coupled difference equations. A coupled difference equation might look like,



where **x** = (xn-1, xn-2, …, xn-k), and **m** = (mn-1, mn-2, …, mn-k). To get the regression coefficients, I guess we can just regress, successively, xn vs. all previous k x, y, and z values, and then regress yn against all previous k x, y, and z values, and then do the same for zn. To avail ourselves of sklearn’s regression algorithm, we’d probably want to write this as:



Again, we can do sklearn’s linear regression thing to get the matrix of fit coefficients.