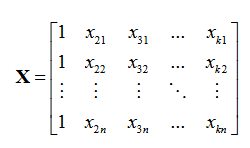
**LAB3 - Multiple Linear Regression**

In this lab, we will perform multiple linear regression in Python. To do this, we will need the .csv provided in the LAB2 folder.

Again, this is the same .csv file we used in the previous labs. This time, we will investigate the effects of age, experience and power to the player’s salary and try to find which metric has greater importance over the others.

In order to implement multiple linear regression, we need to have our independent variables as a matrix (X). Therefore, you will need to extract each independent variable column (“Age”, “Experience” and “Power” in this case) and concatenate them together along with a ones column to form the correct matrix, which should look like this:

[](http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html#sklearn.linear_model.LinearRegression.predict)

Here, instead of having one single variable as a vector, we have multiple vectors forming a matrix. Concatenating vectors can be done in many ways, an easy one is to call numpy’s “vstack” function, which *vertically* stacks each vector. You then need to take the *transpose* of the resulting matrix.

For each column in the matrix, we obtain a different coefficient. The aim is to find coefficients such that:

y = β0 + β1x1 + β2x2 + ……. + βnxn, where

xi, i = 1,2,3,..,n is our set of input variables, and



βj, j = 0,1,2,...,n is the set of estimated coefficients, which is also shown as

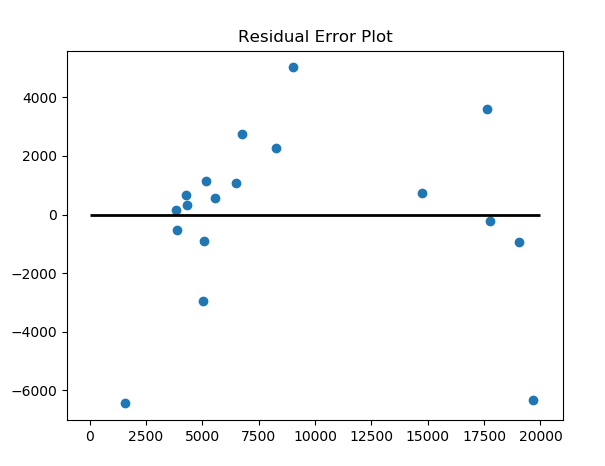
The formula to manually calculate the coefficients is given below:



After calculating the coefficients, we need to plot the results. Unlike single linear regression, we don’t have a regression *line*, but a regression *hyperplane* (with more than 3 dimensions), which unfortunately cannot be plotted. We can, however, plot the *errors* for each observation. We can plot our predicted Y values against each one’s error margin. The formulas are (for predicted Y values and error margins, respectively):



The output of the plot should look like this:



Please use numpy arrays for any numerical arrays or matrices. Similar to the previous labs, we will use matplotlib for plotting purposes. Aside from these (with an exception of “pandas”), no other packages are permitted for use.