Understanding the maths behind linear regression

This lab is not about running a predictive model; it is about understanding the mathematics (algorithms) that goes behind the ready-made methods which we will be using to implement these algorithms. It is about interpreting the results these models spew after the model implementation and making sense of them in the context. Thus, it is of utmost importance to understand the mathematics behind the algorithms and the result parameters of these models.

In this lab, we will discuss a technique called **linear regression**. It is the most basic and generic technique to create a predictive model out of a historical dataset with an output variable.

Before we kick-start the lab, let's discuss what a model means and entails. A mathematical/statistical/predictive model is nothing but a mathematical equation consisting of input variables yielding an output when values of the input variables are provided. For example, let us, for a moment, assume that the price (*P*) of a house is linearly dependent upon its size (*S*), amenities (*A*), and availability of transport (*T*). The equation will look like this:

$$P = a_1 * S + a_2 * A + a_3 * T$$

This is called the **model** and the variables a_1, a_2, and a_3 are called the variable coefficients. The variable *P* is the predicted output while the *S*, *A*, and *T* are input variables. Here, *S*, *A*, and *T* are known but a_1, a_2, and a_3 are not. These parameters are estimated using the historical input and output data. Once, the value of these parameters is found, the equation (model) becomes ready for testing. Now, *S*, *A*, and *T* can be numerical, binary, categorical, and so on; while *P* can also be numerical, binary, or categorical and it is this need to tackle various types of variables that gives rise to a large number of models.

Let us assume that we have a hypothetical dataset containing information about the costs of several houses and their sizes (in square feet):

Size (square feet) X	Cost (lakh INR) Y
1500	45
1200	38
1700	48
800	27

There are two kinds of variables in a model:

- The input or predictor variable, the one which helps predict the value of output variable
- The output variable, the one which is predicted

In this case, cost is the output variable and the size is the input variable. The output and the input variables are generally referred as *Y* and *X* respectively.

In the case of linear regression, we assume that *Y* (*Cost*) is a linear function of *X* (*Size*) and to estimate *Y*, we write:

$$Y_e = \alpha + \beta * X \; or \; \mathrm{Cos}\, t = \alpha + \beta * Size$$

Where *Ye* is the estimated or predicted value of *Y* based on our linear equation.

The purpose of linear regression is to find statistically significant values of *a* and *ß*, which minimize the difference between *Y* and *Y e*. If we are able to determine the values of these two parameters satisfying these conditions, then we will have an equation which we can predict the values of *Y*, given the value of *X*.

So, to summarize, linear regression (like any other supervised algorithm) requires historical data with one output variable and one or more than one input variables to make a model/equation, using which output variables can be calculated/predicted if the input variable is present. In the preceding case, if we find the value of *a =2* and *ß=.3*, then the equation will be:

$$Y_e = 2 + .03X$$

Using this equation, we can find the cost of a home of any size. For a 900 square feet house, the cost will be:

$$Y_e = 2 + 900 * .03 = 29 units$$

The next question that we can ask is how do we estimate *a* and *ß*. We use a method called least square sum of the difference between *Y* and Y_e. The difference between the *Y* and Y_e can be represented as *e*:

$$e = (Y - Y_e)$$

Thus, the objective is to minimize $\sum(Y - Ye)^2 = \sum\left(Y - \left(\alpha + \beta * X\right)\right)^2$; the summation is over all the data points.

We can also minimize: $\sum e^2 = e_1^2 + e_2^2 + \ldots\ldots en^2$, where *n* is the number of data points.

Using calculus, we can show that the values of the unknown parameters are as follows:

$$\beta = \sum\left(Xi - Xm\right)\left(Yi - Ym\right) / \sum\left(Xi - Xm\right)^2$$
$$\alpha = Ym - \beta * Xm$$

where Xm – mean of X values and Ym-mean of Y values

We will not go into the derivation of this formula. The steps for this derivation can be found in any good maths book or the internet if you are interested:

Almost all the statistical tools have ready-made programs to calculate the coefficients *a* and *ß*. However, it is still very important to understand how they are calculated behind the curtain

Making sense of result parameters

Apart from the R^2 statistic, there are other statistics and parameters that one needs to look at in order to do the following:

1. Select some variables and discard others for the model.
2. Assess the relationship between the predictor and output variable and check whether a predictor variable is significant in the model or not.
3. Calculate the error in the values predicted by the selected model.

Let us now see some of the statistics which helps to address the issues discussed earlier.

p-values

One thing to realize here is that the calculation of *a* and *β* are estimates and not the exact calculations. Whether their values are significant or not need to be tested using a hypothesis test.

The hypothesis tests whether the value of *β* is non-zero or not; in other words whether there is a sufficient correlation between *X* and `yact`. If there is, the *β* will be non-zero.

In the equation, *y= a +β*x*, if we put *β=0*, there will be no relation between *y* and *x*. Hence the hypothesis test is defined, as shown in the following:

$$Null\ hypothesis - Ho : \beta = 0$$
$$Alternate\ Hypothesis - Ha : \beta \Diamond 0$$

So, whenever a regression task is performed and *β* is calculated, there will be an accompanying t-statistic and a p-value corresponding to this hypothesis test, calculated automatically by the program. Our task is to assume a significance level of our choice and compare this with the p-value. It will be a two-tailed test and if the p-value is less than the chosen significance level, then the null hypothesis that *β=0* is rejected.

If p-value for the t-statistic is less than the significance level, then the null-hypothesis is rejected and *β* is taken to be significant and non-zero. The values of p-value larger than the significance level demonstrate that *β* is not very significant in explaining the relationship between the two variables. As we see in the case of multiple regression (multiple input variables/predictors), this fact can be used to weed out unwanted columns from the model. The higher the p-value, the less significant they are to the model and the less significant ones can be weeded out first.

F-statistics

When one moves from a simple linear regression to a multiple regression, there will be multiple *β*s and each of them will be an estimate. In such a case, apart from testing the significance of the individual variables in the model by checking the p-values associated with their estimation, it is also required to check whether, as a group all the predictors are significant or not. This can be done using the following hypothesis:

$$\begin{aligned}&Null\ hypothesis - Ho: \beta_1 = \beta_2 = \beta_3 = \ldots\ldots = \beta_n = 0\\&Alternate\ Hypothesis - Ha: One\ of\ the\ \beta_i\ is\ not\ equal\ to\ 0\end{aligned}$$

The statistic that is used to test this hypothesis is called the **F-statistic** and is defined as follows:

$$F - statistic = \frac{(SST - SSD)/p}{SSD/(n-p-1)}$$

Where SST and SSD have been defined earlier as:

$$SST = \sum(yact - yavg)2 \qquad SSD = \sum(yact - ypred)2$$

where n=number of rows in the dataset; p- number of predictor variables in the model

The F-statistics follows the F-distribution. There will be a p-value that is associated with this F-statistic. If this p-value is small enough (smaller than the significance level chosen), the null hypothesis can be rejected.

The significance of F-statistic is as follows:

- p-values are about individual relationships between one predictor and one outcome variable. In case of more than one predictor variable, one predictor's relationship with the output might get changed due to the presence of other variables. The F-statistics provides us with a way to look at the partial change in the associated p-value because of the addition of a new variable.
- When the number of predictors in the model is very large and all the $\beta\ i$ are very close to `0`, the individual p-values associated with the predictors might give very small values. In such a case, if we rely solely on individual p-values, we might incorrectly conclude that there is a relationship between the predictors and the outcome, when it is not there actually. In such cases, we should look at the p-value associated with the F-statistic.

Residual Standard Error

Another concept to learn is the concept of **Residual Standard Error** (**RSE**). It is defined as:

$$RSE = \sqrt{\frac{1}{n-2} * \Sigma_{i=1}^{n}\left(yact - y\,\mathrm{mod}\,el\right)^2} \text{ and } SSD = \Sigma_{i=1}^{n}\left(yact - y\,\mathrm{mod}\,el\right)^2$$

So, RSE can be written as for a simple linear regression model.

$$RSE = \sqrt{\frac{1}{n-2} * SSD}$$

Where *n=number* of data points. In general, where *p=number* of predictor variables in the model.

$$RSE = \sqrt{\frac{1}{n-p-1} * SSD}$$

The RSE is an estimate of the standard deviation of the error term (*res*). This is the error that is inevitable even if the model coefficients are known correctly. This may be the case because the model lacks something else, or may be another variable in the model (we have just looked at one variable regression till now, but in most of the practical scenarios we have to deal with multiple regression, where there would be more than one input variable. In multiple regressions, values of the RSE generally go down, as we add more variables that are more significant predictors of the output variable).

The RSE for a model can be calculated using the following code snippet. Here, we are calculating the RSE for the data frame we have used for the model, `df`:

```
df['RSE']=(df['Actual_Output(yact)']-df['ymodel'])**2
RSEd=df.sum()['RSE']
RSE=np.sqrt(RSEd)/98
RSE
```

The value of the RSE comes out to be 0.46 in this case. As you might have guessed, the smaller the RSE, the better the model is. Again, the benchmark to compare this error is the mean of the actual values, `yact`. As we have seen earlier, this value is *ymean=2.53*. So, we will observe an error of 0.46 over 2.53 that amounts to around an 18% error.