# What is Regression?

## Why Regression?

Regression is a commonly used term in many classes, journal articles and social media sites. Being another overly misused term, it seems that the general public has grown to favor the phrase “Just run a regression and solve the issue!”. Unlike the term “statistically proven”, this one is only said by people who do not know what they are actually talking about half the time instead of 100% of the time. In light of posting more entries relating to machine learning and neural networks in the future, as well continuing with the series of “misconceptions about statistics”, I have decided to throw this one in to give readers the appropriate foundations in which most of machine learning relies on.

## So, What Is Covered?

This entry will attempt to explain the meaning of regression, the mathematics and the main assumptions behind it, as well as how to interpret the results.

## What is Regression?

Regression in statistics is simply different techniques that we humans created (models) to try and fulfil 2 main objectives:

* Check for dependencies between different variables
* Predict their future behavior

I will try to avoid using the word 'correlation' for now as it will be explained in more detail in another entry. This is much like how we humans created many different models of cars (some fancier than others), but the main function of a car is basically to get someone from point A to point B in a shorter period.

## There is more than 1 type of regression?

Well of course! If you think back to the analogy of the car, different models of cars would better suit certain terrains even though the main function is still to get a person from point A to point B faster. For example, a specific model such as a Lamborghini would do excellent on a smooth and flat road but would probably breakdown on a rocky and uneven terrain. In this situation a different model such as a Jeep would do far better! If that made perfect sense to you, you are now ready to make the conceptual jump across to the idea of different regression models for different "terrains".

## What the 'Main Function' of Regression?

So, what does "Check for dependencies between different variables and predict their behavior" actually mean? Let's take a very simple example where we have [data](http://college.cengage.com/mathematics/brase/understandable_statistics/7e/students/datasets/slr/frames/frame.html) regarding:

* X = Number of claims for a particular region
* Y = Total amount of claims paid out for that region ($)

If we plot these 2 sets of data against each other, we have the follow:

INSERT GRAPH HERE

Where a single point (red point for example) represents a particular region where there are approximately 53 claims reported and $250 paid out. By simply looking at the location of all the points, we can argue that there is some sort of pattern between the 2 variables. As the number of claims increase, so does the total amount being paid (generally). This 'pattern' is also called a trend and the main goal of regression is to create a line that best fits this trend. Or more accurately, regression **is** fitting the line that best represents behavior between different variables.

## Why bother creating a line?

Although we can see some sort of a trend between those 2 variables, we cannot possible gain anything useful without being able to quantify this trend. For example, if I wanted to know what the approximate payout would be if the number of claims in that region was 80, I could not give you an answer at all because there is no data for the payout at the point of 80 claims! This is the same reason why I cannot tell you the approximate payout when a region has more 300 or more claims! In simple terms, adding a line (doing regression) that fits this trend well will *'fill in the gaps'* between missing data points by using the data points that we **have** information about.

## What kind of line should I use?

Now that we have established a good enough reason for us to go through all this trouble to create a fitting line, we can revisit the analogy of the car to sum this all up conceptually. This pattern or trend that we can observe from the scatterplot can be seen as the 'terrain' and it is our job to find the best 'car' or line (regression model) that would be suitable for the particular terrain. Generally speaking, we can use **any** line we want to.

INSERT GRAPH HERE

The graph above is the same one as before but with 2 trend lines/ regression models fitted. As one can clearly see, the orange line is the automobile equivalent of bringing your Lamborghini to a mountain track. The model of car does not fit the terrain at all! The green trend line on the other hand, is the Jeep model and seems to be doing pretty well in fulfilling the main function of a regression model. Just as there are many different car models and terrains, there are also many different patterns of behaviors and regression models we can utilize such as, Simple Linear Regression, Generalized Linear Models, Polynomial Regression, Poisson Regression, Time-series Regression and many others! Do not be overwhelmed with all the complicated terminology though. Just remember that each kind of regression model is created to suit a different kind 'terrain' with the ultimate goal of making more accurate predictions!

## Why not just join up all the points with a line?

If we were to just join up all the lines wouldn't we have gotten a perfectly accurate model?

INSERT GRAPH HERE

Yes, you would! But the catch is that the model is only perfectly accurate for the data you already have. In other words, it would not generalize well to new data. If we go back to the car analogy, this would be the equivalent of having an extremely customized Jeep model which has tires that can only travel on 1 particular kind of rock (granite for example). If there was even a little bit of sand or soil on the ground, the whole Jeep model would break down. Doesn't seem very useful to be too accurate now does it? This brings up the concept of 'Parsimony' for a model. We have to trade some accuracy for simplicity when building regression models in order for them to be useful for predictions. If the model cannot generalize to new data, there is no point predicting anything as the basis of prediction is adding new data to the model! Now that you understand the basic objectives of regression, we can now delve deeper into the underlying mechanics/ mathematics of how it works and how to interpret the results. For simplicity, this post will limit the technical discussions to the simplest class of regression models: Simple Linear Regression (SLR), where the fitted line is always straight, and the underlying concepts can then be extended to other more complicated models.

## How do I know what is a 'Good Fit'?

Just like any other any terms in mathematics/ statistics, 'Good Fit' is only useful when it is quantifiable. Since we are doing SLR, we must first recognize that our resulting model will be in the form:

Which is the general equation of a straight line. Here, Y is the predicted result (total claim payouts in the case of the previous example) and X is the variable that is said to affect Y (number of claims). The slope of the line is represented by 'm' and the intercept is 'b' (not every line starts from the 0 point). Since we have X and Y fixed from the data given, our job in finding the 'best-fit line' is to find the optimal values of 'm' and 'b'. The next thing we must then recognize is that, for our **predicted line** (whatever it is), if we have any number of points in our dataset, every single fitted ith point will satisfy the equation:

Simply because we are doing a SLR and our predicted line will always be straight. Once you now understand how our fitted line will look and how it is written mathematically, we can start to talk about errors. In general, an error would intuitively mean the difference between our fitted line and the actual data point, which is exactly what SLR is based on! To begin to put this in a pseudo equation let's define a single error for a data point by:

INSERT GRAPH HERE

From the example graph above, all the Error are represented by the black lines leading from the fitted line to the actual data points. Now that we have defined the error for all the individual points, how do we actually combine all these errors together to get the 'total error for the regression line'? Can we just add them up? Although simply adding all the errors up would be the intuitive thing to do, there is actually a slight issue when doing this. From the graph above, we can observe that some black lines (errors) are above the blue line and some are below. Those that are above the blue line display a positive value (see the Error formula above) and those lines that are below will display a negative one. Adding all of them up will cause everything to eventually cancel out, giving us an inaccurate way of measuring error! Since this is the case, we humans have created several ways to counteract this problem and one of them is to square (multiplied by itself) the error values. Keeping this in mind, the new 'benchmark' for each individual error will now be:

Now that we don't face the problem of errors cancelling each other out, we can go on to defining the total error of a fitted line by simply summing over all the data points!

Another interpretation of the Sum Squared Error or SSE is the total area of imaginary squares created by using the error lines as one of the sides. (this is basically what you are doing when squaring a number)

INSERT PICTURE HERE

Okay, now that we have properly defined both conceptually and mathematically the definition of what we want as an 'error' for the trend line, we can basically end of this section by claiming that a 'Good Fit' just means a line that produces the **least** total error when plotted through a bunch of points!

## How do we get the 'Least Squared Error'?

This section of the entry is more technical than the rest and involves some basic calculus which will be briefly explained here (but there will be an in-depth explanation in another post). This brings us to the topic of **'Optimization/ Minimization'**. It is the main reason why we even study calculus in school, and it is being used in almost every single background program or system you can think of. Now, it is also going to help us get the line with the least squared errors. The basic idea behind optimization can be better explained with a simple graph such as:

INSERT PICTURE HERE

Just by looking at the graph, we can roughly tell that point A would be the highest point on this line. As usual, in mathematics, our statements are only useful when they are quantifiable. That can be achieved by realizing that the slope of the graph is only equals to 0 at point A. If we generalize this idea, we can also claim that the slope of any graph will be 0 at a maximum or a minimum point! If we recall from basic calculus, the slope of any function at a particular point is just the derivative of that function with respect to the variable. So, if we have a function:

Now, this idea can be extended to our problem where:

And if we let Yactual = a, and we recognized from before that each Yfitted value is just mXi + b, we can rewrite our function that needs to be optimized as:

Here, our function (whatever shape it has) output would change with different values of 'm' and 'b' and it is our job to find the values of 'm' and 'b' that give the smallest output, or the smallest error. This can be done with the concept of the zero-slope point discussed above. We first take the partial derivatives with respect to both 'm' and 'b' and set that equal to 0 to find the points on our function where 'm' gives the lowest output, and 'b' gives the lowest output individually.

Now, we have 2 equations and 2 unknowns 'm' and 'b'. We can solve for both variables through various ways and we will not go through the tedious algebraic steps here. We will, however, end up with the 2 equations:

These 2 results may seem somewhat cryptic at first, but the important thing to note is that both these formulae depend on information that we already have! Since both 'X', 'a' and 'n' are given from the dataset, we now have formulae that gives us the optimal values of 'm' and 'b' so that our regression line will produce the least squared error!

## So, what does this mean?

Okay, now we have gotten some mathematical formulae to solve for our slope (m) and intercept (b) of our SLR line. We know that we can just calculate both quantities from our given dataset but what does the output mean? Let's say we are studying the effect of X: Hours spent studying on Y: Exam results. Just think of the dataset as a 2-column list of numbers, where the left column represents the number of hours studying, and the right column represents the corresponding exam results. If we plot the graph of this dataset, we may get something that looks like this:

INSERT GRAPH HERE

Where there is a clear trend between the number of hours studying on the X-axis and the results on the Y-axis. Let's also say we went through the steps in calculate our SLR line and found out that our resulting regression line equation is:

What does this mean? A way to make things clearer is to actually replace the values with their respective names:

Now we can see clearly our regressed relationship between the results and the number of hours put into studying for any given ith student. It seems that firstly, if I do not study at all, I can expect to score 25 marks, where 25 = 0.25 ∗ 0 + 25. This means that for ever extra hour of studying I put in, I can expect to get 0.25 more marks! Now, we actually have what we set out to solve in the first place, which is a model that can tell me the result for every level of input that I want, whether or not there is data at that particular level.

## What about the variation of my prediction?

Firstly, I would like to point out that whatever we have done so far was to get a single value estimate (for example 25 marks) from a regression equation. If we wanted to make some statements about the probability of the actual point being above or below that predicted value, we would need an additional assumption which can be summarized by the addition of:

The extra ϵi symbol represents the errors from the predictions (the black lines from before), and what we are implying is that the distribution of those errors follows a normal distribution with a mean of 0 and some finite variance. For more information on the normal distribution you can refer to my previous post on statistically proving things. Note that there are also other implications of the normal distribution assumption of the errors such as minimum variance but that is outside the scope of this topic and will not be covered here. Just know that as of now, we actually can make statements such as:

*“If I study for 0 hours, I can expect to get 25 marks but there is like a 5% chance that I will either get 5 or 45! “*

Which is pretty useful considering the simplicity of the derivation so far! This method of SLR is more commonly known as 'Ordinary Least Squares' or OLS.

## Does this fitted model make sense?

Although it seems like we have nailed this problem, we must always consider the extreme events in order to test the robustness of our model. Without going into any of the metrics for model validation yet (will be covered in a separate post), we have to understand that one of the properties of a linear model is that there is no upper limit for the variables unless otherwise specified. This means that if the maximum points on the exam is 100, my models tell me that I can get above 100 points just by putting in 300 hours on average, which is actually impossible! So, we must be careful when extrapolating our concept to make sure that it would still make sense even when outside the range of our dataset.

## We are barely scrapping the surface

Although it seems like we have gone through quite a few steps to properly understand what regression, or more specifically 'Ordinary Least Squares' method of regression is actually trying to accomplish, there are a lot more technicalities involved from underlying assumptions, statistical properties of the errors all the way to the method in which we have chosen our data points! I will try to (eventually) cover all the significant topics and answer some questions like 'How do we evaluate our model?', 'What if my errors are not normally distributed' and 'Is there another way to optimize besides OLS?' So next time you hear the word regression hopefully you will now have a better idea of what is actually going on. Also, how is this concept used in real life data and modelling? More on this next time!