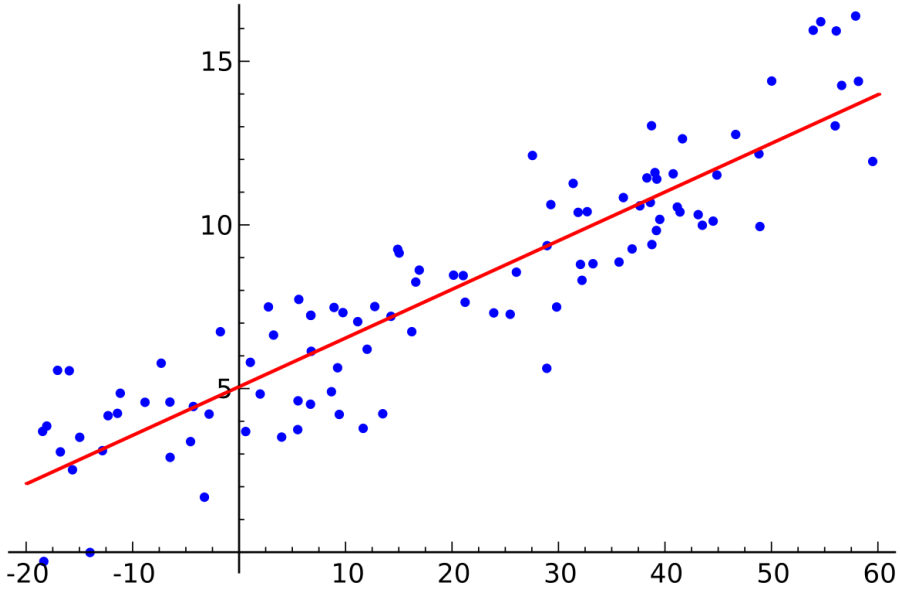
1. Explain the linear regression algorithm in detail.



Linear Regression

Simple linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable. The red line in the above graph is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best. The line can be modelled based on the linear equation shown below.

y = a_0 + a_1 * x ## Linear Equation

The motive of the linear regression algorithm is to find the best values for a_0 and a_1. Before moving on to the algorithm, let's have a look at two important concepts you must know to better understand linear regression.

Cost Function

The cost function helps us to figure out the best possible values for a_0 and a_1 which would provide the best fit line for the data points. Since we want the best values for a_0 and a_1, we convert this search problem into a minimization problem where we would like to minimize the error between the predicted value and the actual value.

$$minimize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

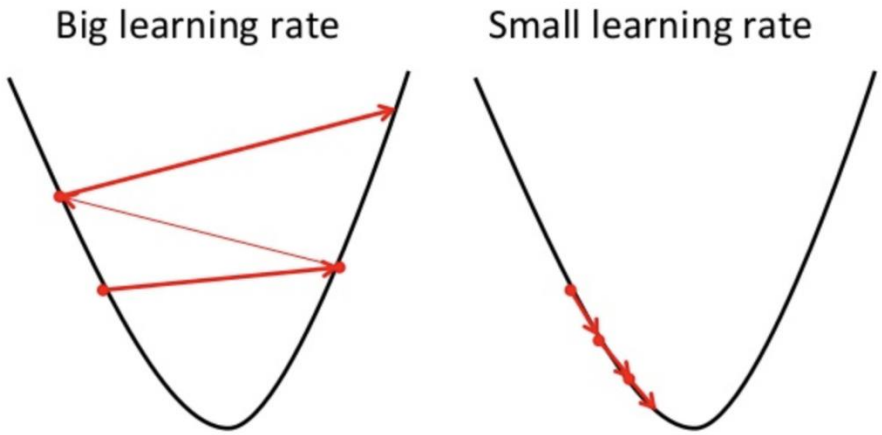
$$J = \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2$$

Minimization and Cost Function

We choose the above function to minimize. The difference between the predicted values and ground truth measures the error difference. We square the error difference and sum over all data points and divide that value by the total number of data points. This provides the average squared error over all the data points. Therefore, this cost function is also known as the Mean Squared Error(MSE) function. Now, using this MSE function we are going to change the values of a_0 and a_1 such that the MSE value settles at the minima.

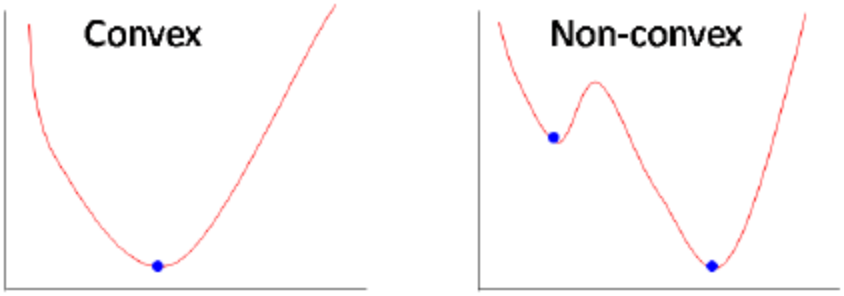
Gradient Descent

The next important concept needed to understand linear regression is gradient descent. Gradient descent is a method of updating a_0 and a_1 to reduce the cost function(MSE). The idea is that we start with some values for a_0 and a_1 and then we change these values iteratively to reduce the cost. Gradient descent helps us on how to change the values.



Gradient Descent

To draw an analogy, imagine a pit in the shape of U and you are standing at the topmost point in the pit and your objective is to reach the bottom of the pit. There is a catch, you can only take a discrete number of steps to reach the bottom. If you decide to take one step at a time you would eventually reach the bottom of the pit but this would take a longer time. If you choose to take longer steps each time, you would reach sooner but, there is a chance that you could overshoot the bottom of the pit and not exactly at the bottom. In the gradient descent algorithm, the number of steps you take is the learning rate. This decides on how fast the algorithm converges to the minima.



Convex vs Non-convex function

Sometimes the cost function can be a non-convex function where you could settle at a local minima but for linear regression, it is always a convex function.

Formula for linear regression equation is given by:

$$y = a + bx$$

a and *b* are given by the following formulas:

$$b\,(slope) = \frac{n\sum xy - (\sum x)(\sum y)}{n\sum x^2 - (\sum x)^2}$$

$$a\,(intercept) = \frac{n\sum y - b(\sum x)}{n}$$

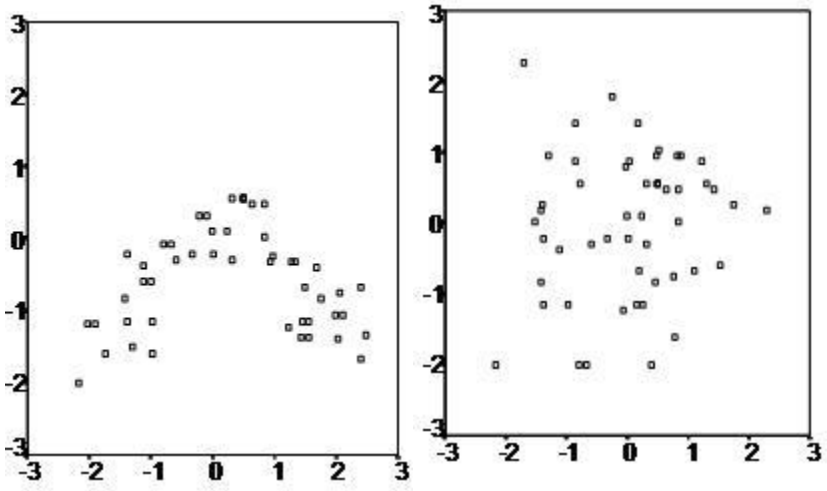
Where,
x and *y* are two variables on regression line.
b = Slope of the line.
a = *y*-intercept of the line.
x = Values of first data set.
y = Values of second data set.

2. What are the assumptions of linear regression regarding residuals?

Linear regression analysis makes several key assumptions:
There must be a linear relationship between the outcome variable and the independent variables. Scatterplots can show whether there is a linear or curvilinear relationship.
Multivariate Normality–Multiple regression assumes that the residuals are normally distributed.
No Multicollinearity—Multiple regression assumes that the independent variables are not highly correlated with each other. This assumption is tested using Variance Inflation Factor (VIF) values.
Homoscedasticity–This assumption states that the variance of error terms are similar across the values of the independent variables. A plot of standardized residuals versus predicted values can show whether points are equally distributed across all values of the independent variables.

Linear regression analysis makes several key assumptions:
There must be a linear relationship between the outcome variable and the independent variables. Scatterplots can show whether there is a linear or curvilinear relationship.
Multivariate Normality–Multiple regression assumes that the residuals are normally distributed.
No Multicollinearity—Multiple regression assumes that the independent variables are not highly correlated with each other. This assumption is tested using Variance Inflation Factor (VIF) values.
Homoscedasticity–This assumption states that the variance of error terms are similar across the values of the independent variables. A plot of standardized residuals versus predicted values can show whether points are equally distributed across all values of the independent variables.
Intellectus Statistics automatically includes the assumption tests and plots when conducting a regression.
Multiple linear regression requires at least two independent variables, which can be nominal, ordinal, or interval/ratio level variables. A rule of thumb for the sample size is that regression analysis requires at least 20 cases per independent variable in the analysis.

First, multiple linear regression requires the relationship between the independent and dependent variables to be linear. The linearity assumption can best be tested with scatterplots. The following two examples depict a curvilinear relationship (left) and a linear relationship (right).



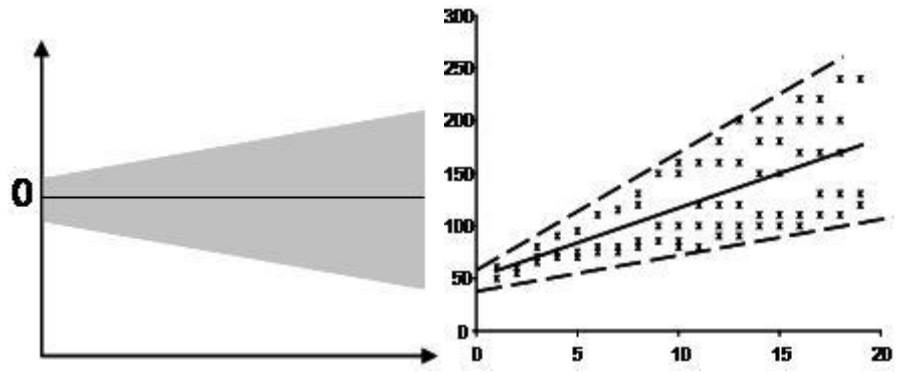
Second, the multiple linear regression analysis requires that the errors between observed and predicted values (i.e., the residuals of the regression) should be normally distributed. This assumption may be checked by looking at a histogram or a Q-Q-Plot. Normality can also be checked with a goodness of fit test (e.g., the Kolmogorov-Smirnov test), though this test must be conducted on the residuals themselves.

Third, multiple linear regression assumes that there is no multicollinearity in the data. Multicollinearity occurs when the independent variables are too highly correlated with each other.

Fourthly, linear regression analysis requires that there is little or no autocorrelation in the data. Autocorrelation occurs when the residuals are not independent from each other. In other words when the value of y(x+1) is not independent from the value of y(x).

While a scatterplot allows you to check for autocorrelations, you can test the linear regression model for autocorrelation with the Durbin-Watson test. Durbin-Watson's d tests the null hypothesis that the residuals are not linearly auto-correlated. While d can assume values between 0 and 4, values around 2 indicate no autocorrelation. As a rule of thumb values of $1.5 < d < 2.5$ show that there is no auto-correlation in the data. However, the Durbin-Watson test only analyses linear autocorrelation and only between direct neighbors, which are first order effects.

The last assumption of the linear regression analysis is homoscedasticity. The scatter plot is good way to check whether the data are homoscedastic (meaning the residuals are equal across the regression line). The following scatter plots show examples of data that are not homoscedastic (i.e., heteroscedastic):



3. What is the coefficient of correlation and the coefficient of determination?

Coefficient of correlation is "R" value which is given in the summary table in the Regression output. R square is also called coefficient of determination. Multiply R times R to get the R square value. In other words Coefficient of Determination is the square of Coefficeint of Correlation.

R square or coeff. of determination shows percentage variation in y which is explained by all the x variables together. Higher the better. It is always between 0 and 1. It can never be negative – since it is a squared value.

It is easy to explain the R square in terms of regression. It is not so easy to explain the R in terms of regression.

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.850[a]	.723	.690	4.57996

a. Predictors: (Constant), weight, horsepower

b. Dependent Variable: mpg

Coefficient of Correlation is the R value i.e. .850 (or 85%). Coefficient of Determination is the R square value i.e. .723 (or 72.3%). R square is simply square of R i.e. R times R.

Coefficient of Correlation: is the degree of relationship between two variables say x and y. It can go between -1 and 1. 1 indicates that the two variables are moving in unison. They rise and fall together and have perfect correlation. -1 means that the two variables are in perfect opposites. One goes up and

other goes down, in perfect negative way. Any two variables in this universe can be argued to have a correlation value. If they are not correlated, then the correlation value can still be computed which would be 0. The correlation value always lies between -1 and 1 (going thru 0 – which means no correlation at all – perfectly not related). Correlation can be rightfully explained for simple linear regression – because you only have one x and one y variable. For multiple linear regression R is computed, but then it is difficult to explain because we have multiple variables involved here. That's why R square is a better term. You can explain R square for both simple linear regressions and also for multiple linear regressions.

4. Explain the Anscombe's quartet in detail.

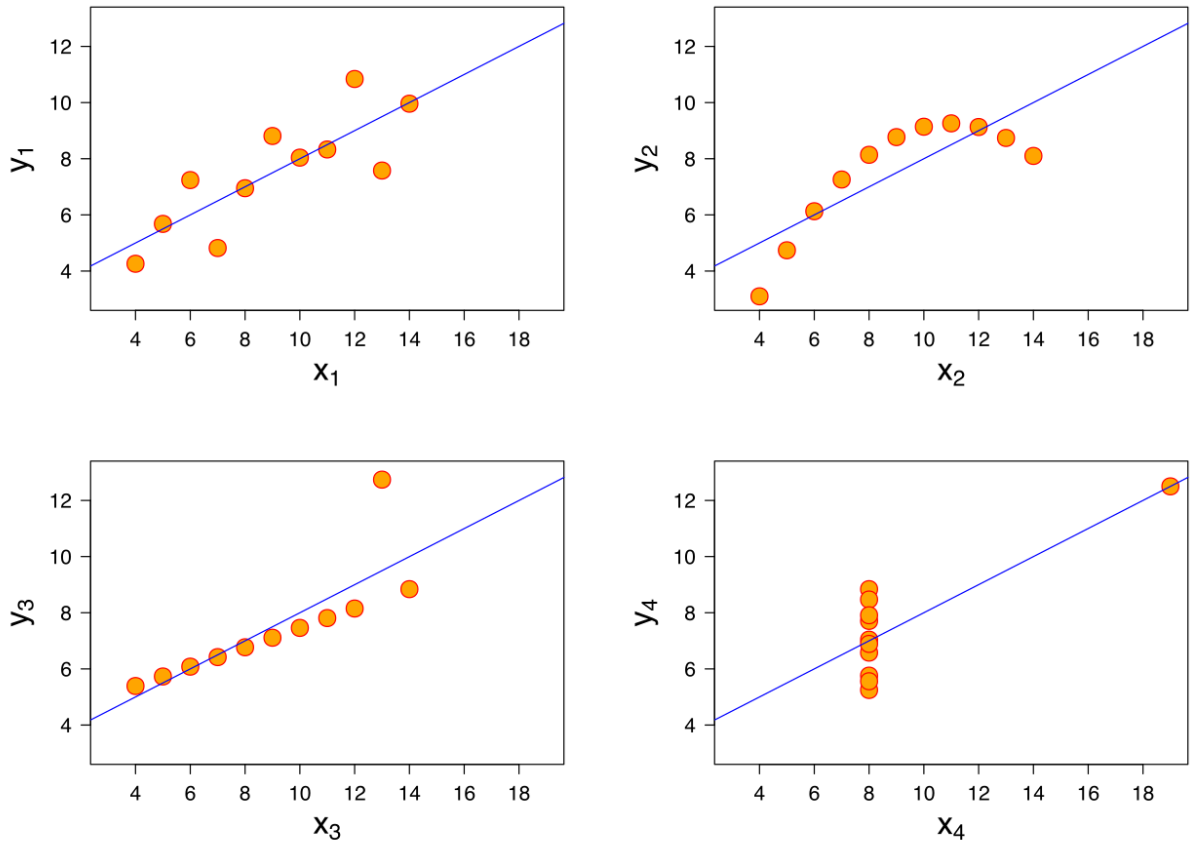
Anscombe's Quartet was developed by statistician Francis Anscombe. It comprises four datasets, each containing eleven (x,y) pairs. The essential thing to note about these datasets is that they share the same descriptive statistics. But things change completely, when they are graphed. Each graph tells a different story irrespective of their similar summary statistics.

	I		II		III		IV	
	x	y	x	y	x	y	x	y
	10	8,04	10	9,14	10	7,46	8	6,58
	8	6,95	8	8,14	8	6,77	8	5,76
	13	7,58	13	8,74	13	12,74	8	7,71
	9	8,81	9	8,77	9	7,11	8	8,84
	11	8,33	11	9,26	11	7,81	8	8,47
	14	9,96	14	8,1	14	8,84	8	7,04
	6	7,24	6	6,13	6	6,08	8	5,25
	4	4,26	4	3,1	4	5,39	19	12,5
	12	10,84	12	9,13	12	8,15	8	5,56
	7	4,82	7	7,26	7	6,42	8	7,91
	5	5,68	5	4,74	5	5,73	8	6,89
SUM	99,00	82,51	99,00	82,51	99,00	82,50	99,00	82,51
AVG	9,00	7,50	9,00	7,50	9,00	7,50	9,00	7,50
STDEV	3,32	2,03	3,32	2,03	3,32	2,03	3,32	2,03

Quartet's Summary Stats
The summary statistics show that the means and the variances were identical for x and y across the groups :

- Mean of x is 9 and mean of y is 7.50 for each dataset.
- Similarly, the variance of x is 11 and variance of y is 4.13 for each dataset
- The correlation coefficient (how strong a relationship is between two variables) between x and y is 0.816 for each dataset

When we plot these four datasets on an x/y coordinate plane, we can observe that they show the same regression lines as well but each dataset is telling a different story :



- Dataset I appears to have clean and well-fitting linear models.
- Dataset II is not distributed normally.
- In Dataset III the distribution is linear, but the calculated regression is thrown off by an outlier.
- Dataset IV shows that one outlier is enough to produce a high correlation coefficient.

This quartet emphasizes the importance of visualization in Data Analysis. Looking at the data reveals a lot of the structure and a clear picture of the dataset.

5. What is Pearson's R?

Pearson's correlation coefficient is the covariance of the two variables divided by the product of their standard deviations. The form of the definition involves a "product moment", that is, the mean (the first moment about the origin) of the product of the mean-adjusted random variables; hence the modifier product-moment in the name.

A Pearson's correlation can determine if two numeric variables are significantly linearly related. A correlation analysis provides information on the strength and direction of the linear relationship between two variables.

The Pearson's correlation coefficient varies between -1 and +1 where:

r = 1 means the data is perfectly linear with a positive slope (i.e., both variables tend to change in the same direction)
r = -1 means the data is perfectly linear with a negative slope (i.e., both variables tend to change in different directions)
r = 0 means there is no linear association
r > 0 < 5 means there is a weak association
r > 5 < 8 means there is a moderate association
r > 8 means there is a strong association

a Pearson correlation between variables X and Y is calculated by

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

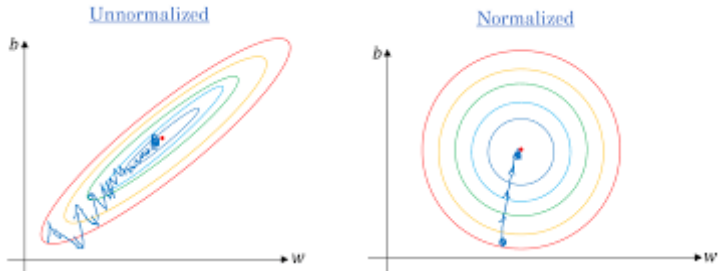
The formula basically comes down to dividing the covariance by the product of the standard deviations. Since a coefficient is a number divided by some other number our formula shows why we speak of a correlation coefficient.

6. What is scaling? Why is scaling performed? What is the difference between normalized scaling and standardized scaling?

This means that you're transforming your data so that it fits within a specific scale, like 0–100 or 0–1. You want to scale data when you're using methods based on measures of how far apart data points, like support vector machines, or SVM or k-nearest neighbors, or KNN. With these algorithms, a change of "1" in any numeric feature is given the same importance.

Scaling is a technique used to apply to optimization problems. It means normalization of features so that its values lie under a particular range like between (0,1). If you train a model without feature scaling, then it takes time to find global minima. That's why normalized data is used to train the model faster.

For example:



Advantages of Feature scaling:

1. It helps to train machine learning algorithms faster

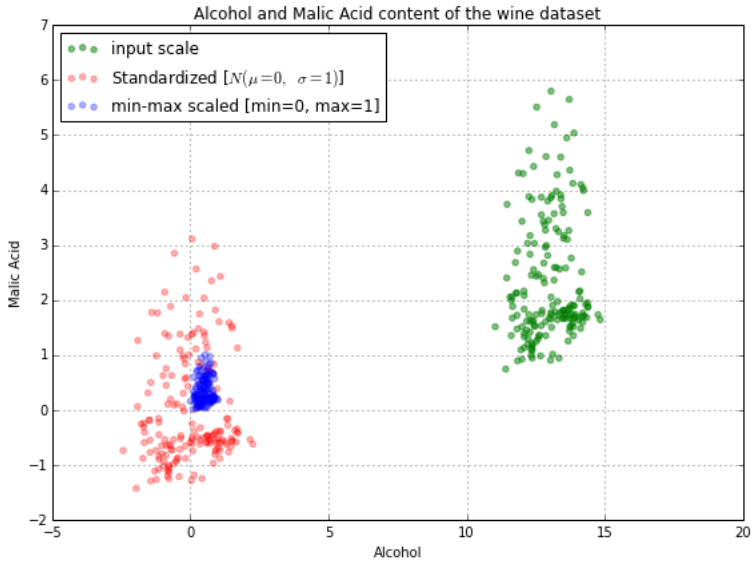
2. It prevents machine learning algorithm to get stuck in local minima
3. It gives better error surface shape
4. Weight decay and bayes optimization can be achieved efficiently

Two methods are usually well known for rescaling data. Normalization, which scales all numeric variables in the range [0,1]. One possible formula is given below:

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

On the other hand, you can use standardization on your data set. It will then transform it to have zero mean and unit variance, for example using the equation below:

$$x_{new} = \frac{x - \mu}{\sigma}$$



About standardization

Standardizing the features so that they are centered around 0 with a standard deviation of 1 is not only important if we are comparing measurements that have different units, but it is also a general requirement for many machine learning algorithms. Intuitively, we can think of gradient descent as a prominent example (an optimization algorithm often used in logistic regression, SVMs, perceptrons,

neural networks etc.); with features being on different scales, certain weights may update faster than others since the feature values play a role in the weight updates
Some examples of algorithms where feature scaling matters are:

- k-nearest neighbors with an Euclidean distance measure if want all features to contribute equally
- k-means (see k-nearest neighbors)
- logistic regression, SVMs, perceptrons, neural networks etc. if you are using gradient descent/ascent-based optimization, otherwise some weights will update much faster than others
- linear discriminant analysis, principal component analysis, kernel principal component analysis since you want to find directions of maximizing the variance (under the constraints that those directions/eigenvectors/principal components are orthogonal); you want to have features on the same scale since you'd emphasize variables on "larger measurement scales" more. There are many more cases than I can possibly list here … I always recommend you to think about the algorithm and what it's doing, and then it typically becomes obvious whether we want to scale your features or not.

In addition, we'd also want to think about whether we want to "standardize" or "normalize" (here: scaling to [0, 1] range) our data. Some algorithms assume that our data is centered at 0. For example, if we initialize the weights of a small multi-layer perceptron with tanh activation units to 0 or small random values centered around zero, we want to update the model weights "equally." As a rule of thumb I'd say: When in doubt, just standardize the data, it shouldn't hurt.

About Min-Max scaling

An alternative approach to Z-score normalization (or standardization) is the so-called Min-Max scaling (often also simply called "normalization" - a common cause for ambiguities).
In this approach, the data is scaled to a fixed range - usually 0 to 1.
The cost of having this bounded range - in contrast to standardization - is that we will end up with smaller standard deviations, which can suppress the effect of outliers.

A Min-Max scaling is typically done via the following equation:

7. You might have observed that sometimes the value of VIF is infinite. Why does this happen?

Variance inflation factors show the degree to which a regression coefficient will be affected because of the variable's redundancy with other independent variables. As the squared multiple correlation of any predictor variable with the other predictors approaches unity, the corresponding VIF becomes infinite.

Multicollinearity is a state of very high intercorrelations or inter-associations among the independent variables. It is therefore a type of disturbance in the data, and if present in the data the statistical inferences made about the data may not be reliable.

The variance inflation factor (VIF) identifies correlation between independent variables and the strength of that correlation. Statistical software calculates a VIF for each independent variable. VIFs start at 1 and have no upper limit. A value of 1 indicates that there is no correlation between this independent variable and any others. VIFs between 1 and 5 suggest that there is a moderate correlation, but it is not

severe enough to warrant corrective measures. VIFs greater than 5 represent critical levels of multicollinearity where the coefficients are poorly estimated, and the p-values are questionable.

If there is perfect correlation between independent variables, then VIF = infinity.

In Geely Auto assignment, enginetype_rotor column and cylindernumber_two have exact same pair of values for cylindernumber = 2 and enginetype = rotor. So, they are highly correlated with each other, resulting in VIF values as 1.

If VIF is large and multicollinearity affects your analysis results, then you need to take some corrective actions before you can use multiple regression. Here are the various options:

- One approach is to review your independent variables and eliminate terms that are duplicates or not adding value to explain the variation in the model. For example, if your inputs are measuring the weight in kgs and lbs then just keep one of these variables in the model and drop the other one. Dropping the term with a large value of VIF will hopefully, fix the VIF for the remaining terms and now all the VIF factors are within the threshold limits. If dropping one term is not enough, then you may need to drop more terms as required.
- A second approach is to use principal component analysis and determine the optimal set of principal components that best describe your independent variables. Using this approach will get rid of your multicollinearity problem but it may be hard for you to interpret the meaning of these "new" independent variables.
- The third approach is to increase the sample size. By adding more data points to our model, hopefully, the confidence intervals for the model coefficients are narrower to overcome the problems associated with multicollinearity.
- The fourth approach is to transform the data to a different space like using a log transformation so that the independent variables are no longer correlated as strongly with each other.
- Finally, you can use a different type of model call ridge regression that better handles multicollinearity.

8. What is the Gauss-Markov theorem?

The Gauss Markov theorem says that, under certain conditions, the ordinary least squares (OLS) estimator of the coefficients of a linear regression model is the best linear unbiased estimator (BLUE), that is, the estimator that has the smallest variance among those that are unbiased and linear in the observed output variables.

The Gauss–Markov theorem states that the ordinary least squares (OLS) estimator has the lowest sampling variance within the class of linear unbiased estimators, if the errors in the linear regression model are uncorrelated, have equal variances and expectation value of zero. The errors do not need to be normal, nor do they need to be independent and identically distributed (only uncorrelated with mean zero and homoscedastic with finite variance).

There are five Gauss Markov assumptions (also called conditions):

1. Linearity: the parameters we are estimating using the OLS method must be themselves linear.

2. Random: our data must have been randomly sampled from the population.
3. Non-Collinearity: the regressors being calculated aren't perfectly correlated with each other.
4. Exogeneity: the regressors aren't correlated with the error term.
5. Homoscedasticity: no matter what the values of our regressors might be, the error of the variance is constant.

9. Explain the gradient descent algorithm in detail.

Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. It is basically used for updating the parameters of the learning model.

The primary goal of machine learning algorithms is always to build a model, which is basically a hypothesis which can be used to find an estimation for Y based on X. Let us consider an example of a model based on certain housing data which comprises of the sale price of the house, the size of the house etc. Suppose we want to predict the pricing of the house based on its size. It is clearly a regression problem where given some inputs, we would like to predict a continuous output.

The hypothesis is usually presented as

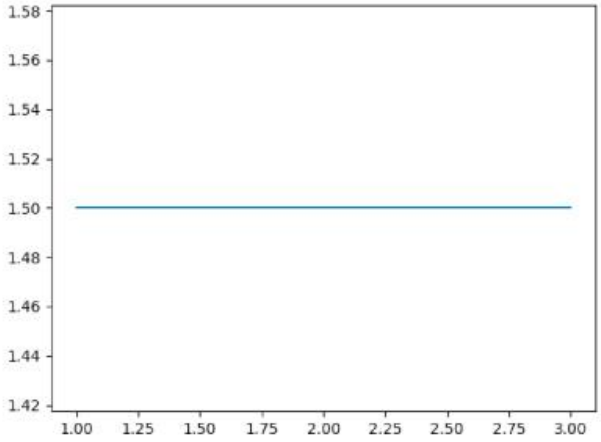
$$h_\theta(x) = \theta_0 + \theta_1 x$$

where the theta values are the *parameters*.

Let us look into some examples and visualize the hypothesis:

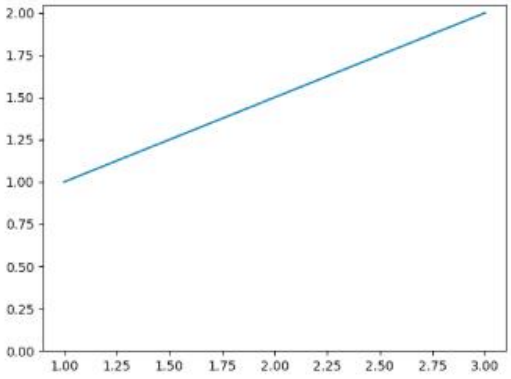
$$\theta_0 = 1.5$$
$$\theta_1 = 0$$

This yields h(x) = 1.5 + 0x. 0x means no slope, and y will always be the constant 1.5. This looks like:



Now let us consider,

$$\theta_0 = 1$$
$$\theta_1 = 0.5$$



Where, h(x) = 1 + 0.5x

Gradient descent is used to minimize a cost function J(W) parameterized by a model parameters W. The gradient (or derivative) tells us the incline or slope of the cost function. Hence, to minimize the cost function, we move in the direction opposite to the gradient.

1. Initialize the weights W randomly.
2. Calculate the gradients G of cost function w.r.t parameters. This is done using partial differentiation: G = ∂J(W)/∂W. The value of the gradient G depends on the inputs, the current values of the model parameters, and the cost function. You might need to revisit the topic of differentiation if you are calculating the gradient by hand.
3. Update the weights by an amount proportional to G, i.e. W = W - ηG
4. Repeat until the cost J(w) stops reducing, or some other pre-defined termination criteria is met.

Types of gradient Descent:

Batch Gradient Descent: This is a type of gradient descent which processes all the training examples for each iteration of gradient descent. But if the number of training examples is large, then batch gradient descent is computationally very expensive. Hence if the number of training examples is large, then batch gradient descent is not preferred. Instead, we prefer to use stochastic gradient descent or mini-batch gradient descent.

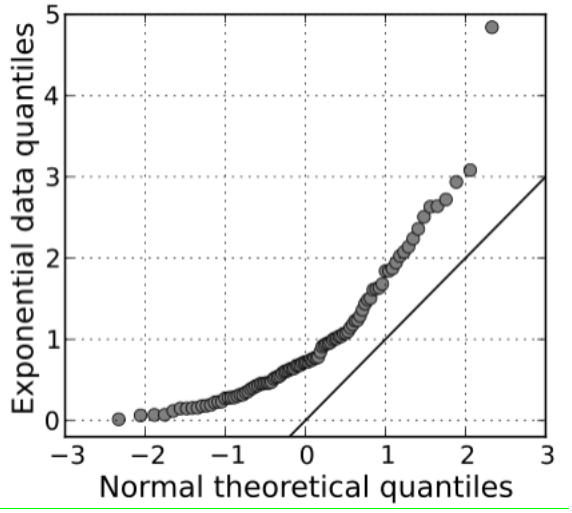
Stochastic Gradient Descent: This is a type of gradient descent which processes 1 training example per iteration. Hence, the parameters are being updated even after one iteration in which only a single example has been processed. Hence this is quite faster than batch gradient descent. But again, when the number of training examples is large, even then it processes only one example which can be additional overhead for the system as the number of iterations will be quite large.

Mini Batch gradient descent: This is a type of gradient descent which works faster than both batch gradient descent and stochastic gradient descent. Here b examples where b<m are processed per iteration. So even if the number of training examples is large, it is processed in batches of b training examples in one go. Thus, it works for larger training examples and that too with lesser number of iterations.

10. What is a Q-Q plot? Explain the use and importance of a Q-Q plot in linear regression.

A Q–Q plot is a plot of the quantiles of two distributions against each other, or a plot based on estimates of the quantiles. The pattern of points in the plot is used to compare the two distributions.

The main step in constructing a Q–Q plot is calculating or estimating the quantiles to be plotted. If one or both of the axes in a Q–Q plot is based on a theoretical distribution with a continuous cumulative distribution function (CDF), all quantiles are uniquely defined and can be obtained by inverting the CDF. If a theoretical probability distribution with a discontinuous CDF is one of the two distributions being compared, some of the quantiles may not be defined, so an interpolated quantile may be plotted. If the Q–Q plot is based on data, there are multiple quantile estimators in use. Rules for forming Q–Q plots when quantiles must be estimated or interpolated are called plotting positions.



The points plotted in a Q–Q plot are always non-decreasing when viewed from left to right. If the two distributions being compared are identical, the Q–Q plot follows the 45° line y = x. If the two distributions agree after linearly transforming the values in one of the distributions, then the Q–Q plot follows some line, but not necessarily the line y = x. If the general trend of the Q–Q plot is flatter than the line y = x, the distribution plotted on the horizontal axis is more dispersed than the distribution plotted on the vertical axis. Conversely, if the general trend of the Q–Q plot is steeper than the line y = x, the distribution plotted on the vertical axis is more dispersed than the distribution plotted on the horizontal axis. Q–Q plots are often arced, or "S" shaped, indicating that one of the distributions is more skewed than the other, or that one of the distributions has heavier tails than the other.

Although a Q–Q plot is based on quantiles, in a standard Q–Q plot it is not possible to determine which point in the Q–Q plot determines a given quantile. For example, it is not possible to determine the median of either of the two distributions being compared by inspecting the Q–Q plot. Some Q–Q plots indicate the deciles to make determinations such as this possible.

The intercept and slope of a linear regression between the quantiles gives a measure of the relative location and relative scale of the samples. If the median of the distribution plotted on the horizontal axis is 0, the intercept of a regression line is a measure of location, and the slope is a measure of scale. The distance between medians is another measure of relative location reflected in a Q–Q plot. The

"probability plot correlation coefficient" (PPCC plot) is the correlation coefficient between the paired sample quantiles. The closer the correlation coefficient is to one, the closer the distributions are to being shifted, scaled versions of each other. For distributions with a single shape parameter, the probability plot correlation coefficient plot provides a method for estimating the shape parameter – one simply computes the correlation coefficient for different values of the shape parameter, and uses the one with the best fit, just as if one were comparing distributions of different types.

Another common use of Q–Q plots is to compare the distribution of a sample to a theoretical distribution, such as the standard normal distribution N(0,1), as in a normal probability plot. As in the case when comparing two samples of data, one orders the data (formally, computes the order statistics), then plots them against certain quantiles of the theoretical distribution.