Machine Learning

LVC 1: Introduction to Supervised Learning and Regression

Machine learning has found its extensive uses in solving real-life problems in today's world. For different categories of problems, there are different suitable techniques to solve them. One such prominent technique in machine learning is **Supervised Learning**. In supervised learning, we have a target variable whose value we need to predict. The algorithm is provided with some **labeled records** to supervise the existing relation between the target variable and the non-target variables. A supervised learning problem is comprised of the **following steps**:

1. **Formulation**: In this step, the output feature and the input features are identified. This step is very important in order to define the problem appropriately.

2. **Solution**: In this step, the relation between the output feature and the input features is established using the given algorithm and the available data.

3. **Performance Assessment**: In this step, the performance of the model is assessed. This is required to see if the model can perform better and compare the performance with other models. In machine learning, there are multiple metrics available to give a clear idea about the performance of the model.

4. **Interpretation**: Finally, the solution has to be interpreted in such a way that it can be applied to the business. This is helpful in decision-making while solving real-life problems.

Before going ahead, let's go through some basic definitions that will be useful in understanding the topics further.

- **Record**: A record is a single data point from the given dataset. It is also known as an **instance**, **sample, observation**, etc.

- **Feature:** A feature is an attribute associated with the record that describes the record. For example, height, weight, age, etc. are the features of a person.

- **Model:** A model is a **mathematical relation** between the output feature and the input features. It is an anticipation of the actual relation existing between the set of features. But a model is never the truth. It is never a perfect relationship and always has a scope of improvement in itself. The utilization of the model is high because it is well designed and problem-specific.

- **Independent features:** These are the features that are independent of each other. A change in one cannot assure a certain change in another.

- **Dependent features:** These are the features that depend on the independent features. The relation between the dependent and the independent variables is established to build the model.

The big picture

The aim of the entire process is to make relevant predictions on the new or unseen data. To do so, a reliable relationship between the independent and dependent variables is developed. This relation is called the **model** in machine learning. Using this model, predictions are made for the new data points. So, if X is the set of input/independent variables and Y is the output/dependent variable, then for a new data point, Y has to be predicted using X.

In supervised learning, the outcome can be a continuous variable or a categorical variable. In case the output is a categorical variable, it is known as a **classification problem**, whereas, if the output variable is continuous, it is known as a **regression problem**. For example, the life expectancy of a person is a continuous feature. Hence, it will be predicted using regression algorithms. On the other hand, whether it will rain or not today, is a categorical feature, hence it will be solved by classification algorithms.

In general, there are two approaches to solve a data science problem:

a. **Data-ML-Prediction**: Here, using the data, a formula or a relation between the output and the input variables is developed. This formula is called the model. The objective is to get a model with good performance and make final predictions on the unseen data. This type of approach is called the **machine learning approach.** In this approach, the theory does not always explain the success of the model as the emphasis is on applying the algorithms.

b. **Data-Stats-Model-Prediction**: In the statistical approach, using statistical techniques, an **empirical formula** or **statistical model** is prepared. Using that model, predictions over unseen data are made. Such an approach is called **Statistical approach**. This approach needs a deep understanding of simple models/methods and relies on theory to give interpretable results.

In today's world statistics and machine learning are two inseparable fields. An expert in one has to be an expert in the other to function better while solving real-life problems. Nowadays, being only a perfect statistician or a perfect machine learning expert will not function well.

Now, let's understand what a basic statistical framework is.

A basic statistical framework

As mentioned earlier, there are majorly two types of features in supervised learning, categorical and continuous, in machine learning. A categorical feature is a feature that can take on one of a limited, and usually fixed, number of possible values. While a continuous feature can take any value from an infinite number of values within a certain range. In real scenarios, the dependent and independent variables might be of any type among the above-mentioned classes. To be able to make the predictions, there has to be a mapping between the independent variable to the dependent variable. This mapping is done using a function called an **estimator**.

An **estimator** is a function that is developed by using the available data. It gets trained on the available data and makes predictions on the unseen data. In general, this estimator is also called a model, or a mathematical relation, or a rule-based relation between the output and the input variables. Let x_1, x_2, ... , x_n be the n independent variables while y is the dependent variable. Let g be the estimator function, then the mathematical relation can be denoted as follows:

$$y = g(x_1, x_2, x_3, \ldots\ldots x_n)$$

In statistics, there are many methods to come up with the estimator. A few of them are listed below:

1. **Plugin Estimators:** It is related to directly using the function that depicts the relation between the output and the input features. The **plugin principle** says that a feature of a given distribution can be approximated by the same feature of the empirical distribution of a sample of observations drawn from the given distribution. The feature of the **empirical distribution** is called a **plugin** estimate of the feature of the given distribution. For example, a quantile of a given distribution can be approximated by the analogous quantile of the empirical distribution of a sample of draws from the given distribution.

2. **Maximum likelihood:** It is one of the important ways of getting the correct relation between the output and the input features. It is used to construct the loss or cost function of an algorithm. In this function, the variables are "the parameters of the model" that are set under a hypothesis on the available data. The likelihood function is then maximized using calculus to get the values of the parameters. These parameters are considered best for the construction of the actual model/hypothesis that depicts the relation between the output and the input features. So on maximization, it leads to giving an empirical relation between the features.

Advertising and Sales Example

In this lecture, we will discuss the techniques involved and concepts involved in solving regression problems. Let's understand regression problems by considering a simple **example of Advertising and Sales.**

Data about the expenditure on advertisement on TV, Radio, and Newspaper across 200 markets is collected. Companies in this field are spending a lot of money in advertising these products in the market. They are trying to enhance sales by doing advertisements for the corresponding product in the market. We are trying to see whether the budget for advertisement is affecting the number of sales or not. As the output feature is the number of sales, which is a continuous feature, it is a regression problem. If the relationship between advertisement and sales exists, then how can we predict sales given the channel's budget on advertisement?

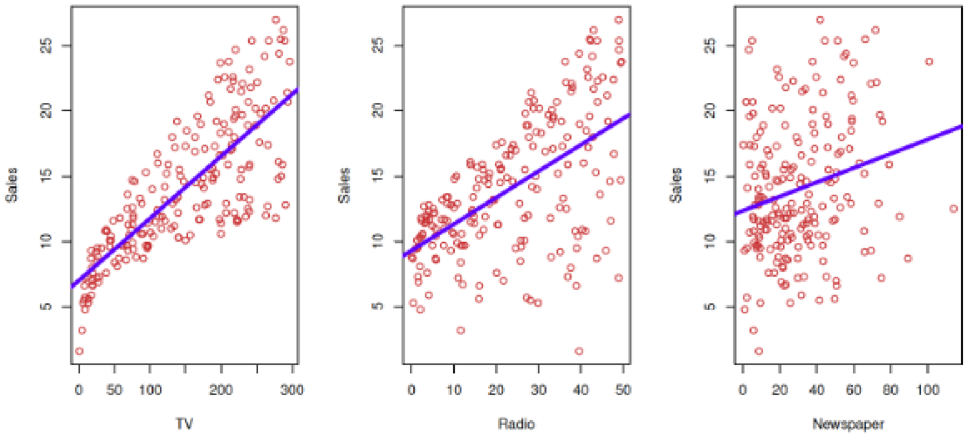
One of the important aspects of regression analysis is to select the features important for the model. To do so, a good starting point is to do the correlation analysis.

Correlation

Correlation gives an idea about the **strength and type** of relationship existing between two numerical features, especially between the output and the input features. Mathematically, it gives the correlation coefficient between two features, denoted by r. It ranges from -1 to +1 (both inclusive). A positive value of r signifies that two features are positively correlated, i.e., if one feature is increased, the other will also increase. The negative value of r shows the inverse relation, i.e., if one feature increases, the other will decrease. Now in terms of magnitude, higher the value (either positive or negative), the stronger is the relation.

However, with a small number of independent features, visualizing the relationship between the independent features and the dependent feature can also give a good idea about the correlation between them.

In our example, we have depicted three **scatter plots** of sales for **three** different channels namely **TV, Radio, and the Newspaper**. The blue line is considered to be the **best fit** line for the available data that can represent the relation between the sales and the advertising budget of the mentioned channel. The plots are depicting whether the relation between the sales and advertisement budget is existing or not.



1. In the first plot for **TV**, it can be seen that there is a **strong dependency** between the output and the input features. For TV, the sales are increasing steeply with an increase in the budget for advertisement.

2. In the second plot for **Radio**, the relation is still there but it is a **bit weaker** than the first one. The steep increase in the number of sales is less than that in TV vs Sales plot.

3. In the third plot for **Newspaper**, the steepness of the line is almost flat. This implies that it is the **weakest** relation among all the three channels.

This way it can be concluded that the advertisement budget has an impact on the sales of the product. In every case, adding an extra budget to the advertisement is increasing the sales of the product. This is how visualization helps in selecting features and having an idea about the existing patterns and relationships between features. But it is not saying anything about the expression between the output and input features.

After getting this useful visual insight from the above plots, let us get into finding the expression of the actual relation between the output and the input features.

Regression

While we look into the dataset, each record has a dimension n, that is the number of features available. Using the set of input features X, we need to come up with a relation g between the output y and the input set X. Once this is done, we use g to predict y for a new data point.

To find g, it requires optimization of a certain function called the **loss function** or the **cost function**. For regression, the loss function is the **sum of squares of the residual terms**. A residual can be defined as the **difference** between the **actual target value** and the **predicted target value** by the model. This function has to be optimized to estimate the parameters of the hypothesis function.

Now, let us have a look at one of the regression algorithms, called linear regression. The mathematical expression of the linear regression model is given as follows:

$$y = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_3 + ... + a_n x_n$$

Here, $a_0, a_1, a_2, ..., a_n$ are the **coefficients** (also known as **weights or parameters**) of the model, $x_1, x_2, x_3, ..., x_n$ are the input features available in the data, and y is the output variable.
For linear regression, the loss function is the sum of squared residual terms. A residual can be expressed mathematically as follows:

$$Residual = (y_i - (a_0 + a_1 x^i_1 + a_2 x^i_2 + a_3 x^i_3 + ... + a_n x^i_n))$$

Where, y_i is the actual target value for the i^{th} observation.

The loss function is formed by adding the square of residuals for all m observations. It can be given as follows:

$$Loss\ Function = \sum_{i=1}^{m} (y_i - (a_0 + a_1 x^i_1 + a_2 x^i_2 + a_3 x^i_3 + ... + a_n x^i_n))^2$$

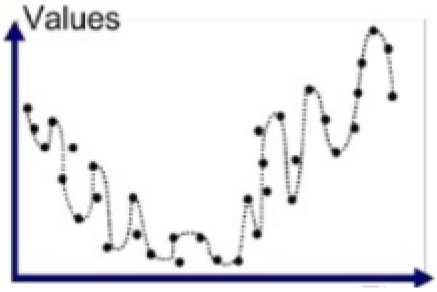
This loss function represents the cost of making wrong predictions. It adds the residual corresponding to each and every training data point. Minimizing such functions will lead to a better model. Here, squares of each residual term in **the training data** are taken to remove the impact of the sign in the algebraic expression. To minimize, it is differentiated to converge to a certain point. Doing so, will fetch n different linear equations, and solving them we will get the corresponding values of $a_1, a_2, a_3, a_4, ... a_n$.

Now that the coefficients are known and hence the relationship is established to make predictions on the unseen data, the equation of linear regression can be used to make predictions for the output feature. If we have to predict multiple dependent variables, then we need to train multiple models corresponding to each output feature.

A good model gives a **generalized performance**, i.e., being able to reproduce the results from the training data on unseen data. It can happen in machine learning that the model performs very well on the training data but cannot perform well on the test data or the model performs badly on both the training and the test data. These two cases are explained below:

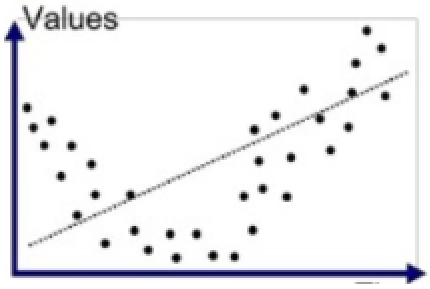
- **Overfitting of the model**: When a machine learning model learns each and every nuance of the training data, including the noise, it becomes **very accurate on the training set** but unable to generalize over the unseen data. Such a model is called an overfit model. An overfit model is too complex for the corresponding data and has a low bias and a high variance.

The below figure shows an overfit model. It can be observed that the model is fitting each and every training example too closely. It is such a complex model that it is very hard to even interpret.



- **Underfitting of the model**: If a model is **performing badly on the training and test datasets**, then it is called an underfit model. Such a model is less complex for the corresponding data and has a high bias and a low variance.

  The below figure shows an example of an underfit model. It can be seen that the model line is far away from the actual data points.



Overfitting and underfitting are not considered suitable for a machine learning model. In an ideal case, the model should have a low bias and a low variance. Such a model is supposed to generalize better over the unseen data.

The machine learning model can have different perspectives to look upon. If we consider weights of the linear regression model as one vector say, θ and the input features X as another vector, then the output y can be represented by the product of these two quantities. Mathematically, it can be given as follows:

$$y = \theta^T X$$

Where, X is the vector of input features $1, x_1, x_2, x_3, ..., x_n$ and θ is the weight vector $a_0, a_1, a_2, a_3, ..., a_n$.

Weights are estimated in such a way that the loss function is minimum, i.e., the total sum of squared errors is minimized.

Now, it is important to understand how the equation is established between the output and the input variables. The process begins with minimizing the loss function.

The solution to the regression problem

Once the loss function is established, it is crucial to find the set of weights/parameters using the training data. Let us look at the loss function once again:

$$Loss\ Function\ =\ \sum_{i=1}^{m}(y_i\ -\ (a_0\ +\ a_1x^i_1\ +\ a_2x^i_2\ +\ a_3x^i_3 + \ ....\ +\ a_nx^i_n))^2$$

Since the objective is to minimize the loss function, we will differentiate the loss function with respect to each unknown weight, and equate it to 0.

$$\frac{d(Loss\ function)}{d(a_j)} = 2\ \times\ \sum_{i=1}^{m}(y_i\ -\ (a_0\ +\ a_1x^i_1\ +\ a_2x^i_2\ +\ a_3x^i_3 + \ ....\ +\ a_nx^i_n))\ \times x^i_j\ =\ 0$$

Doing so for each j from 0 to n, will give $n\ +\ 1$ linear equations which can be solved to get the value of the weights. Using these weights, the final equation between the output and the input features will be set.

In vector form, the solution to the regression problem can be given as:

$$\widehat{\theta} = (X^TX)^{-1}X^TY$$

Where, $\widehat{\theta}$ is the vector with estimated values of the weights.

Nowadays, solving such linear equations is a very easy task for computers. It is much faster than manual computations. Let us have a look at the results for the Advertising and Sales example we began with.

Results for the Advertising and Sales example

Doing all the requisite processings, we found the weights for the final model. These weights are associated with the corresponding features to get the final predictions. The equation of the model is shown below:

$$\widehat{Sales}\ =\ 2.94\ +\ 0.046\ \times\ TV\ +\ 0.19\ \times\ Radio\ -\ 0.001\ \times\ Newspaper$$

A **simple linear regression model** is one where only one **independent** feature is used to make a prediction of the output feature. The below equation is the equation of the simple regression model with only one independent feature - Newspaper.

$$\widehat{Sales} = 12.35 + 0.055 \times Newspaper$$

Interpretation and Justification

While working with the linear regression algorithm, one of the crucial things is to interpret the working of the algorithm. To find the best possible model, there are certain processes that the algorithm follows. It is quite necessary to develop an understanding of how the algorithm reaches the final model. In this section, let's understand the concepts of empirical risk minimization and maximum likelihood.

Empirical Risk Minimization

There are a lot of factors that influence a machine learning model. One such factor is the quality of the samples taken from the population. When we consider the entire population as our training data, then the training becomes very complex due to the computational effort involved in processing a huge amount of data. So, in general we take **a small sample** from the population.

The term empirical implies that we minimize our **error based on a sample set**. The empirical error is also sometimes called the generalization error. The reason is that, in most problems, we don't have access to the whole population, but only the training subset. We want to generalize based on that subset. This error is also called the risk, hence the term risk in empirical risk minimization.

Different samples will fetch different linear regression models. As the size of the sample increases, the estimates become closer to the estimates on the population itself. Since the size of sample might be a limitation, we need to make sure that the sample is a good representation of the population. The predictions of the model trained on a sample that is a good representative of the population are trustworthy than a model built on a sample which is not a representative of the population.

Maximum Likelihood

Maximum Likelihood Estimation is a probabilistic framework for solving the problem of density estimation. It involves maximizing a likelihood function in order to find the probability distribution and parameters that best explain the observed data.

Let (X_i, Y_i) be a sample in a certain dataset with n records. Consider $Y_i|X_i$ is an event that shows the occurrence of Y_i provided X_i has already occurred, i.e., given the input features X_i , the occurrence of Y_i.

In the entire sample, there are n such events. The likelihood of the event $Y_i|X_i$ is the probability of occurrence of this event.

In the context of linear regression, let us consider the relationship between Y_i and X_i which is as follows:

$$Y_i \; = \; \theta_0^* + \theta_1^* X_i + W_i$$

Where θ_0^* is the constant term in the equation, θ_1^* is the coefficient, and W_i is the error term (noise).

It is assumed that the events in the sample are independent of each other and the error terms follow a normal distribution, with a mean equal to the zero and the standard deviation σ. Keeping in mind the normal distribution, mathematically, the likelihood function is defined as follows:

$$max_\theta \, P(Y|X;\ \theta) \; = \; \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} exp\{-\frac{(Y_i - \theta_0^* - \theta_1^* X_i)^2}{2\sigma^2}\}$$

The above equation shows the likelihood of a single event. To estimate the collective likelihood of the entire sample of events, we need to combine the likelihood of all the individual events in the sample. As we assume that the events in the sample are independent of each other, we can apply the multiplication rule of probability. Hence, the likelihood of the entire sample can be given as follows:

$$L(\theta_0^*, \theta_i^*) \; = \prod_{i=1}^{n} P(Y_i \,|\, X_i) \; = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \; exp\{-\frac{(Y_i - \theta_0^* - \theta_i^* X_i)^2}{2\sigma^2}\}$$

The above equation shows the combined probability of all the events to occur in the sample. To estimate the parameters, the above expression has to be maximized and the maximum value of the above expression is called the maximum likelihood.

After training the model, it is required to do the **performance assessment** of the model.

Performance Assessment

Performance assessment is the process of assessing the performance of a machine learning model. It is required because it tells us about the possible scope of changes and improvements to be done to make the model perform better. In machine learning, there are a certain number of methods to do the performance assessment. Such methods ensure the best possible fit of the model.

For example, in regression problems, R^2 value (read as R-squared) is an important performance metric that tells us about the quality of the fit. It indicates the variance explained in the dependent variable by the independent variables. Mathematically, it is given as follows:

$$R^2 = 1 - \left(\frac{RSS}{TSS}\right)$$

Here, RSS is the residual sum of squares of the model and TSS is the total sum of squares of the model. TSS is calculated by taking the mean of the dependent variable as the prediction and then calculating the residual sum of squares. This is the maximum possible value of error a model can show. So, the term RSS/TSS is always less than or equal to 1. A lesser RSS means a better model, which will lead to R^2 tending to 1. While a higher RSS means a bad model, which will lead to R^2 tending to 0.

$R^2 = 1$ means that the $RSS = 0$ and the model is perfectly fitting to the training data. Such models are fitting best to the training data but might not be an overfit model.

$R^2 = 0$ means that the RSS and TSS are equal. The model is as bad as a **naive** model where all the predictions are guessed to be the mean of the existing output values.

Remark: R^2 can be negative in the extreme case where the model's predictions are even worse than the naive prediction. This will make the RSS greater than the TSS which, in turn, will make the R^2 negative.

Take an example of the model

$$\widehat{Sales} = 2.94 + 0.046 \times TV + 0.19 \times Radio - 0.001 \times Newspaper$$

Here, R^2 is 0.897 which is a high value. Hence, all the budgets together explain a lot about the sales.

Considering the simple linear regression models:

- **For newspaper alone:** $R^2 = 0.05$ which implies that the newspaper budget explains very little about the sales.
- **For tv alone:** $R^2 = 0.61$
- **For radio alone:** $R^2 = 0.33$

Adjusted R-Squared

Adjusted R^2 is a corrected goodness-of-fit measure for regression models. It uses the R^2 value to get a more appropriate metric which takes into account the number of data points and the number of independent variables. Mathematically, it is given as:

$$R^2_{adj} = 1 - [(1 - R^2)(n - 1)/(n - k - 1)]$$

Where, n is the number of points in the data, k is the number of independent regressors, i.e., the number of variables in your model, excluding the constant.

Now, let's answer the question **why we need Adjusted R^2?**

R^2 tends to optimistically estimate the fit of the linear regression. It always increases or stays the same as the number of independent variables in the model is increased. It doesn't penalize addition of insignificant variables to the model. Adjusted R^2 attempts to correct for this overestimation. It might decrease if a specific variable does not improve the model. Hence, Adjusted R^2 is always less than or equal to R^2. In general, it also lies between 0 and 1.

How noisy/reliable are my estimates of weights?

Estimating weights is one of the aspects of the model preparation process. Along with this, we also need to ensure how reliable it is. There is randomness in the weights because it is dependent on the **input features** which contain noise. Due to this, the weights are random variables following approximately normal distribution. Weights also have their ground truth values.

Remark: The weights will exactly follow the normal distribution if we assume the noise to be normally distributed.

Let the ground truth value be θ^* and the estimated value be $\widehat{\theta}$. Then, the expected value is given as $E(\widehat{\theta} - \theta^*)^2$. It can be broken into two components namely **bias** and **variance**.

$$E[(\widehat{\theta} - \theta^*)^2] = (E[\widehat{\theta}] - \theta^*)^2 + var(\widehat{\theta})$$

In the above equation, the first term is the **bias** term while the second term is the **variance**.

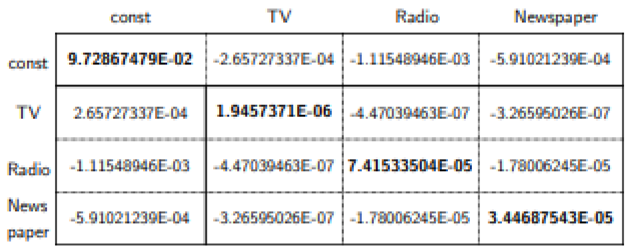
If the estimator is unbiased, i.e., $E[\widehat{\theta}] = \theta^*$, the bias is zero and only the variance component contributes to the error. Fortunately, the estimator is unbiased in linear regression, hence we focus on the variance of $\widehat{\theta}$.

The distribution of weights

Weights are random variables but they are almost normally distributed. The lesser the standard error of these random variables, the more trustworthy the model is. If it is high, the model is less trustworthy because it is supposed to give erroneous outcomes when deployed to unseen data.

Covariance Matrix

It is a matrix that represents the variation of a set of variables with each other. To understand it more clearly, let us look into the covariance matrix for our Advertising and Sales example.



	const	TV	Radio	Newspaper
const	**9.72867479E-02**	-2.65727337E-04	-1.11548946E-03	-5.91021239E-04
TV	2.65727337E-04	**1.9457371E-06**	-4.47039463E-07	-3.26595026E-07
Radio	-1.11548946E-03	-4.47039463E-07	**7.41533504E-05**	-1.78006245E-05
News paper	-5.91021239E-04	-3.26595026E-07	-1.78006245E-05	**3.44687543E-05**

The table contains four features namely const, TV, Radio, and Newspaper where, TV, Radio, and Newspaper are the features available in the model and the feature const is the constant term appeared in the model. The entire table is the covariance matrix while the individual entries are the covariance of the two variables belonging to its row and column. The diagonal entries are the covariance of a variable with itself which is the same as the variance.

Mathematically, it is given as follows:

$$Cov(\widehat{\theta},\ \theta^{*}) = E[(\widehat{\theta} - \theta^{*})\,(\widehat{\theta} - \theta^{*})^{T}]$$

Where, $\widehat{\theta}$ is the estimated value of the weight vector θ and θ^{*} is the actual value of the weight vector θ

Confidence Interval

As we know, the regression weights are estimated using samples and are subject to uncertainty. Therefore, we will never exactly estimate the true value of these parameters from sample data in an empirical application. However, we can construct confidence intervals for the weights so that this range will contain the population parameter with a certain degree of confidence, expressed in percentage terms.

Since $\widehat{\theta}$ follows the normal distribution, the 95% confidence interval is two standard deviations away from the estimated value. Mathematically, it is given as follows:

$$95\%\ CI:\ [\widehat{\theta} - 2\widehat{\sigma},\ \widehat{\theta} + 2\widehat{\sigma}\,]$$

The above expression implies that out of 100 samples (data sets) where each sample has a certain number of records, 95% of samples (data sets) are expected to result in confidence intervals that contain the true value of regression weights.

While reporting the outcome of a model, it is a good practice to add the confidence interval. It becomes more impactful.

To test the compatibility of the data with the weights, let us test the null hypothesis that $\theta^{*} = 0$.

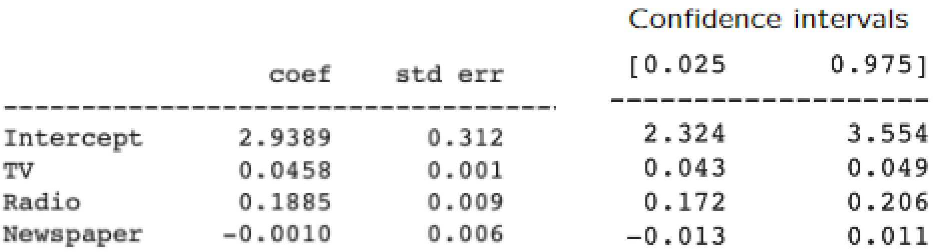
Testing the hypothesis

In linear regression, we have the null hypothesis that a particular variable does not add value to the model and the corresponding weight is zero. If the estimated value of weights deviates a lot from 0, then the null hypothesis will be rejected. It would be considered that enough evidence is found against the null hypothesis.

If the estimated value is close to 0, it is considered that enough evidence could not be found against the null hypothesis. In such a case, the null hypothesis is not rejected and is considered to be the true statement.

Apart from correlation test, to understand the significance of the features for the model, **Wald's test** is used in machine learning. The higher the significance of the feature, the more important it is to the model and should be included in it.

For our Advertising and Sales example, the Wald's test showed that the Newspaper coefficient is not significant, hence we fail to reject the null hypothesis that $\theta^{*} = 0$. The below figures also show that the coefficient of Newspaper is very close to 0 and the confidence interval of the coefficient also contains 0 while the confidence intervals for all other coefficients do not contain 0.



```
                                              Confidence intervals
                     coef      std err        [0.025        0.975]
------------------------------------------    --------------------
Intercept          2.9389       0.312          2.324         3.554
TV                 0.0458       0.001          0.043         0.049
Radio              0.1885       0.009          0.172         0.206
Newspaper         -0.0010       0.006         -0.013         0.011
```

In general, while reporting, we use the p-value corresponding to the hypothesis test. If it is less than the **significance level,** then the **null hypothesis** is rejected, and if it is more than the significance value, then we fail to reject the **null hypothesis.**

Sources of Error

Now, let us understand the different sources of errors while building the regression model. For example, suppose the following model is built:

$$y = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + ... + a_nx_n$$

Corresponding to such models there are two sources of errors:

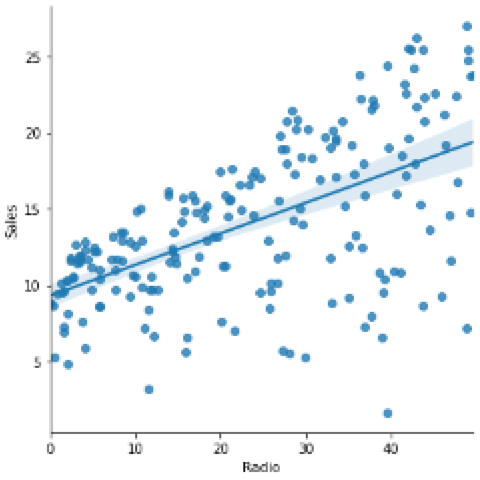
1. **The noise of the data:** It is an inherent source of error in the data. It occurs while the data is acquired either by some tool or a person. This error can not be treated because it is inherent inside the data.

2. **Variance**: It is the error associated with the model. It can be reduced and modified. It is related to the inaccuracies of the weights estimated during the training of the model.

Confidence Bands

Since we don't know the true regression line and only estimate the best fit line in linear regression, the estimates might be noisy. The gray (light shaded) bands around the regression line represent the range which would contain the true regression line with a certain level of confidence. The width of the confidence band indicates how accurate the estimate is for those values of the independent variable. Confidence bands are fatter at both ends because as you get farther from the mean of data points the uncertainty increases.

The below figure shows the 95% confidence band of the simple linear regression line for the variable Radio.


Sales
Radio