Unit-1

Introduction:

**ARTIFICIAL INTELIGENCE**:

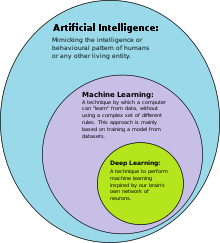
Artificial intelligence (AI) is the ability of a [computer](https://www.britannica.com/technology/computer) or a [robot](https://www.britannica.com/technology/robot-technology) controlled by a computer to do tasks that are usually done by humans because they require human intelligence and discernment. Although there are no AIs that can perform the wide variety of tasks an ordinary human can do, some AIs can match humans in specific tasks.

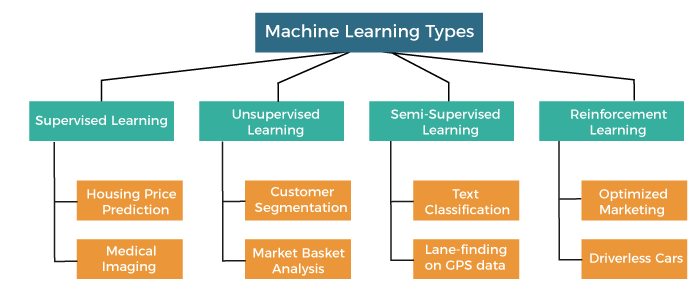
**MACHINE LEARNING**:

machine learning is the concept that a computer program can learn and adapt to new data without human intervention. Machine learning is a field of [artificial intelligence](https://www.investopedia.com/alternative-investments-4427781) (AI) that keeps a computer’s built-in [algorithms](https://www.investopedia.com/terms/a/algorithm.asp) current regardless of changes in the worldwide economy.

Machine learning can be applied in a variety of areas, such as in investing, advertising, lending, organizing news, fraud detection, and more.

**DEEP LEARNING**:

Deep learning is a subset of [machine learning (ML)](https://www.hpe.com/in/en/what-is/machine-learning.html), which is itself a subset of [artificial intelligence (AI)](https://www.hpe.com/in/en/what-is/artificial-intelligence.html). The concept of AI has been around since the 1950s, with the goal of making computers able to think and reason in a way similar to humans. As part of making machines able to think, ML is focused on how to make them learn without being explicitly programmed. Deep learning goes beyond ML by creating more complex hierarchical models that are meant to mimic how humans learn new information  
**TYPES OF MECHINE LEARNING**:

Machine learning is a subset of AI, which enables the machine to automatically learn from data, improve performance from past experiences, and make predictions. Machine learning contains a set of algorithms that work on a huge amount of data. Data is fed to these algorithms to train them, and on the basis of training, they build the model & perform a specific task 

1. Supervised Machine Learning

As its name suggests, [Supervised machine learning](https://www.javatpoint.com/supervised-machine-learning) is based on supervision. It means in the supervised learning technique, we train the machines using the "labelled" dataset, and based on the training, the machine predicts the output. Here, the labelled data specifies that some of the inputs are already mapped to the output. More preciously, we can say; first, we train the machine with the input and corresponding output, and then we ask the machine to predict the output using the test dataset.

Supervised machine learning can be classified into two types of problems, which are given below:

* **Classification**
* **Regression**

## 2. Unsupervised Machine Learning

[Unsupervised learnin](https://www.javatpoint.com/unsupervised-machine-learning)g is different from the Supervised learning technique; as its name suggests, there is no need for supervision. It means, in unsupervised machine learning, the machine is trained using the unlabeled dataset, and the machine predicts the output without any supervision.

In unsupervised learning, the models are trained with the data that is neither classified nor labelled, and the model acts on that data without any supervision.

Unsupervised Learning can be further classified into two types, which are given below:

* **Clustering**
* **Association**

## 3. Semi-Supervised Learning

Semi-Supervised learning is a type of Machine Learning algorithm that lies between Supervised and Unsupervised machine learning**.** It represents the intermediate ground between Supervised (With Labelled training data) and Unsupervised learning (with no labelled training data) algorithms and uses the combination of labelled and unlabeled datasets during the training period.

**A**lthough Semi-supervised learning is the middle ground between supervised and unsupervised learning and operates on the data that consists of a few labels, it mostly consists of unlabeled data. As labels are costly, but for corporate purposes, they may have few labels. It is completely different from supervised and unsupervised learning as they are based on the presence & absence of labels.

To overcome the drawbacks of supervised learning and unsupervised learning algorithms, the concept of Semi-supervised learning is introduced. The main aim of [semi-supervised learning](https://www.javatpoint.com/semi-supervised-learning) is to effectively use all the available data, rather than only labelled data like in supervised learning. Initially, similar data is clustered along with an unsupervised learning algorithm, and further, it helps to label the unlabeled data into labelled data. It is because labelled data is a comparatively more expensive acquisition than unlabeled data.

We can imagine these algorithms with an example. Supervised learning is where a student is under the supervision of an instructor at home and college. Further, if that student is self-analysing the same concept without any help from the instructor, it comes under unsupervised learning. Under semi-supervised learning, the student has to revise himself after analyzing the same concept under the guidance of an instructor at college.

### Advantages and disadvantages of Semi-supervised Learning

**Advantages:**

* It is simple and easy to understand the algorithm.
* It is highly efficient.
* It is used to solve drawbacks of Supervised and Unsupervised Learning algorithms.

**Disadvantages:**

* Iterations results may not be stable.
* We cannot apply these algorithms to network-level data.
* Accuracy is low.

## 4. Reinforcement Learning

Reinforcement learning works on a feedback-based process, in which an AI agent (A software component) automatically explore its surrounding by hitting & trail, taking action, learning from experiences, and improving its performance**.** Agent gets rewarded for each good action and get punished for each bad action; hence the goal of reinforcement learning agent is to maximize the rewards.

In reinforcement learning, there is no labelled data like supervised learning, and agents learn from their experiences only.

The [reinforcement learning](https://www.javatpoint.com/reinforcement-learning) process is similar to a human being; for example, a child learns various things by experiences in his day-to-day life. An example of reinforcement learning is to play a game, where the Game is the environment, moves of an agent at each step define states, and the goal of the agent is to get a high score. Agent receives feedback in terms of punishment and rewards.

Due to its way of working, reinforcement learning is employed in different fields such as Game theory, Operation Research, Information theory, multi-agent systems.

A reinforcement learning problem can be formalized using **Markov Decision Process(MDP).** In MDP, the agent constantly interacts with the environment and performs actions; at each action, the environment responds and generates a new state.

### Categories of Reinforcement Learning

Reinforcement learning is categorized mainly into two types of methods/algorithms:

* **Positive Reinforcement Learning:** Positive reinforcement learning specifies increasing the tendency that the required behaviour would occur again by adding something. It enhances the strength of the behaviour of the agent and positively impacts it.
* **Negative Reinforcement Learning:** Negative reinforcement learning works exactly opposite to the positive RL. It increases the tendency that the specific behaviour would occur again by avoiding the negative condition

. **Advantages:**

* It helps in solving complex real-world problems which are difficult to be solved by general techniques.
* The learning model of RL is similar to the learning of human beings; hence most accurate results can be found.
* Helps in achieving long term results.

**Disadvantage**

* RL algorithms are not preferred for simple problems.
* RL algorithms require huge data and computations.
* Too much reinforcement learning can lead to an overload of states which can weaken the results.

**MAIN CHALLENGES OF MACHINE LEARNING**:

* Not enough training data.
* Poor Quality of data.
* Irrelevant features.
* Nonrepresentative training data.
* Overfitting and Underfitting.

## 1. Not enough training data :

Let’s say for a child, to make him learn what an apple is, all it takes for you to point to an apple and say apple repeatedly. Now the child can recognize all sorts of apples.

Well, machine learning is still not up to that level yet; it takes a lot of data for most of the algorithms to function properly. For a simple task, it needs thousands of examples to make something out of it, and for advanced tasks like image or speech recognition, it may need lakhs(millions) of examples.

## 2. Poor Quality of data:

Obviously, if your training data has lots of errors, outliers, and noise, it will make it impossible for your machine learning model to detect a proper underlying pattern. Hence, it will not perform well.

So put in **every ounce of effort**in cleaning up your training data. No matter how good you are in selecting and hyper tuning the model, this part plays a major role in helping us make an accurate machine learning model.

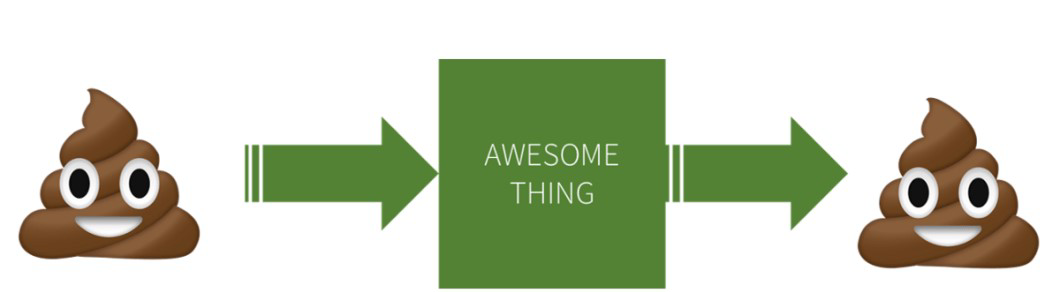
“Most Data Scientists spend a significant part of their time in cleaning data”.

There are a couple of examples when you’d want to clean up the data :

* If you see some of the instances are clear outliers just discard them or fix them manually.
* If some of the instances are missing a feature like (E.g., 2% of user did not specify their age), you can either ignore these instances, or fill the missing values by median age, or train one model with the feature and train one without it to come up with a conclusion.

## 3. Irrelevant Features:

“Garbage in, garbage out (GIGO).”



[*Image Source*](https://www.google.com/url?sa=i&url=https%3A%2F%2Fcirco.io%2Fgigo%2F&psig=AOvVaw3FIiG07ExC0wxdc14cHczU&ust=1624633398091000&source=images&cd=vfe&ved=0CAoQjRxqFwoTCODm5Z7FsPECFQAAAAAdAAAAABAD)

In the above image, we can see that even if our model is “AWESOME” and we feed it with garbage data, the result will also be garbage(output). Our training data must always contain **more relevant**and **less to none irrelevant features.**

The credit for a successful machine learning project goes to coming up with a good set of features on which it has been trained (often referred to as **feature engineering** ), which includes feature selection, extraction, and creating new features which are other interesting topics to be covered in upcoming blogs.

## 4. Nonrepresentative training data:

To make sure that our model generalizes well, we have to make sure that our training data should be representative of the new cases that we want to generalize to.

If train our model by using a nonrepresentative training set, it won’t be accurate in predictions it will be **biased against one** class or a group.

For E.G., Let us say you are trying to build a model that recognizes the genre of music. One way to build your training set is to search it on youtube and use the resulting data. Here we assume that youtube’s search engine is providing representative data but in reality, the search will be biased towards popular artists and maybe even the artists that are popular in your location(if you live in India you will be getting the music of Arijit Singh, Sonu Nigam or etc).

So use representative data during training, so your model won’t be biased among one or two classes when it works on testing data.

## 5. Overfitting and Underfitting :

Let’s start with an example, say one day you are walking down a street to buy something, a dog comes out of nowhere you offer him something to eat but instead of eating he starts barking and chasing you but somehow you are safe. After this particular incident, you might think all dogs are not worth treating nicely.

So this**overgeneralization**is what we humans do most of the time, and unfortunately machine learning model also does the same if not paid attention. In machine learning, we call this overfitting i.e model performs well on training data but fails to generalize well.

Overfitting happens when our model is too complex.

Things which we can do to overcome this problem:

1. Simplify the model by selecting one with fewer parameters.
2. By reducing the number of attributes in training data.
3. Constraining the model.
4. Gather more training data.

**STATISTICAL LEARNING**:

**INTRODUCTION**:

*An Introduction to Statistical Learning* provides a broad and less technical treatment of key topics in statistical learning. Each chapter includes an R lab. This book is appropriate for anyone who wishes to use contemporary tools for data analysis.

**SUPERVISED LEARNING**:

As its name suggests, [Supervised machine learning](https://www.javatpoint.com/supervised-machine-learning) is based on supervision. It means in the supervised learning technique, we train the machines using the "labelled" dataset, and based on the training, the machine predicts the output. Here, the labelled data specifies that some of the inputs are already mapped to the output. More preciously, we can say; first, we train the machine with the input and corresponding output, and then we ask the machine to predict the output using the test dataset.

Let's understand supervised learning with an example. Suppose we have an input dataset of cats and dog images. So, first, we will provide the training to the machine to understand the images, such as the **shape & size of the tail of cat and dog, Shape of eyes, colour, height (dogs are taller, cats are smaller), etc.** After completion of training, we input the picture of a cat and ask the machine to identify the object and predict the output. Now, the machine is well trained, so it will check all the features of the object, such as height, shape, colour, eyes, ears, tail, etc., and find that it's a cat. So, it will put it in the Cat category. This is the process of how the machine identifies the objects in Supervised Learning.

**The main goal of the supervised learning technique is to map the input variable(x) with the output variable(y).** Some real-world applications of supervised learning are **Risk Assessment, Fraud Detection, Spam filtering,** etc.

### Categories of Supervised Machine Learning

Supervised machine learning can be classified into two types of problems, which are given below:

* **Classification**
* **Regression**

### a) Classification

Classification algorithms are used to solve the classification problems in which the output variable is categorical, such as "**Yes" or No, Male or Female, Red or Blue, etc**. The classification algorithms predict the categories present in the dataset. Some real-world examples of classification algorithms are **Spam Detection, Email filtering, etc.**

Some popular classification algorithms are given below:

* **Random Forest Algorithm**
* **Decision Tree Algorithm**
* **Logistic Regression Algorithm**
* **Support Vector Machine Algorithm**

### b) Regression

Regression algorithms are used to solve regression problems in which there is a linear relationship between input and output variables. These are used to predict continuous output variables, such as market trends, weather prediction, etc.

Some popular Regression algorithms are given below:

* **Simple Linear Regression Algorithm**
* **Multivariate Regression Algorithm**
* **Decision Tree Algorithm**
* **Lasso Regression**

### Advantages and Disadvantages of Supervised Learning

**Advantages:**

* Since supervised learning work with the labelled dataset so we can have an exact idea about the classes of objects.
* These algorithms are helpful in predicting the output on the basis of prior experience.

**Disadvantages:**

* These algorithms are not able to solve complex tasks.
* It may predict the wrong output if the test data is different from the training data.
* It requires lots of computational time to train the algorithm.

### Applications of Supervised Learning

Some common applications of Supervised Learning are given below:

* **ImageSegmentation:**  
  Supervised Learning algorithms are used in image segmentation. In this process, image classification is performed on different image data with pre-defined labels.
* **MedicalDiagnosis:**  
  Supervised algorithms are also used in the medical field for diagnosis purposes. It is done by using medical images and past labelled data with labels for disease conditions. With such a process, the machine can identify a disease for the new patients.
* **Fraud Detection -** Supervised Learning classification algorithms are used for identifying fraud transactions, fraud customers, etc. It is done by using historic data to identify the patterns that can lead to possible fraud.
* **Spam detection -** In spam detection & filtering, classification algorithms are used. These algorithms classify an email as spam or not spam. The spam emails are sent to the spam folder.
* **Speech Recognition -** Supervised learning algorithms are also used in speech recognition. The algorithm is trained with voice data, and various identifications can be done using the same, such as voice-activated passwords, voice commands, etc

**UNSUPERVISED LEARNING:**

[Unsupervised learnin](https://www.javatpoint.com/unsupervised-machine-learning) Is different from the Supervised learning technique; as its name suggests, there is no need for supervision. It means, in unsupervised machine learning, the machine is trained using the unlabeled dataset, and the machine predicts the output without any supervision.

In unsupervised learning, the models are trained with the data that is neither classified nor labelled, and the model acts on that data without any supervision.

**The main aim of the unsupervised learning algorithm is to group or categories the unsorted dataset according to the similarities, patterns, and differences.** Machines are instructed to find the hidden patterns from the input dataset.

Let's take an example to understand it more preciously; suppose there is a basket of fruit images, and we input it into the machine learning model. The images are totally unknown to the model, and the task of the machine is to find the patterns and categories of the objects.

So, now the machine will discover its patterns and differences, such as colour difference, shape difference, and predict the output when it is tested with the test dataset.

### Categories of Unsupervised Machine Learning

Unsupervised Learning can be further classified into two types, which are given below:

* **Clustering**
* **Association**

### 1) Clustering

The clustering technique is used when we want to find the inherent groups from the data. It is a way to group the objects into a cluster such that the objects with the most similarities remain in one group and have fewer or no similarities with the objects of other groups. An example of the clustering algorithm is grouping the customers by their purchasing behaviour.

Some of the popular clustering algorithms are given below:

* **K-Means Clustering algorithm**
* **Mean-shift algorithm**
* **DBSCAN Algorithm**
* **Principal Component Analysis**
* **Independent Component Analysis**

### 2) Association

Association rule learning is an unsupervised learning technique, which finds interesting relations among variables within a large dataset. The main aim of this learning algorithm is to find the dependency of one data item on another data item and map those variables accordingly so that it can generate maximum profit. This algorithm is mainly applied in **Market Basket analysis, Web usage mining, continuous production**, etc.

Some popular algorithms of Association rule learning are **Apriori Algorithm, Eclat, FP-growth algorithm.**

### Advantages and Disadvantages of Unsupervised Learning Algorithm

**Advantages:**

* These algorithms can be used for complicated tasks compared to the supervised ones because these algorithms work on the unlabeled dataset.
* Unsupervised algorithms are preferable for various tasks as getting the unlabeled dataset is easier as compared to the labelled dataset.

**Disadvantages:**

* The output of an unsupervised algorithm can be less accurate as the dataset is not labelled, and algorithms are not trained with the exact output in prior.
* Working with Unsupervised learning is more difficult as it works with the unlabelled dataset that does not map with the output.

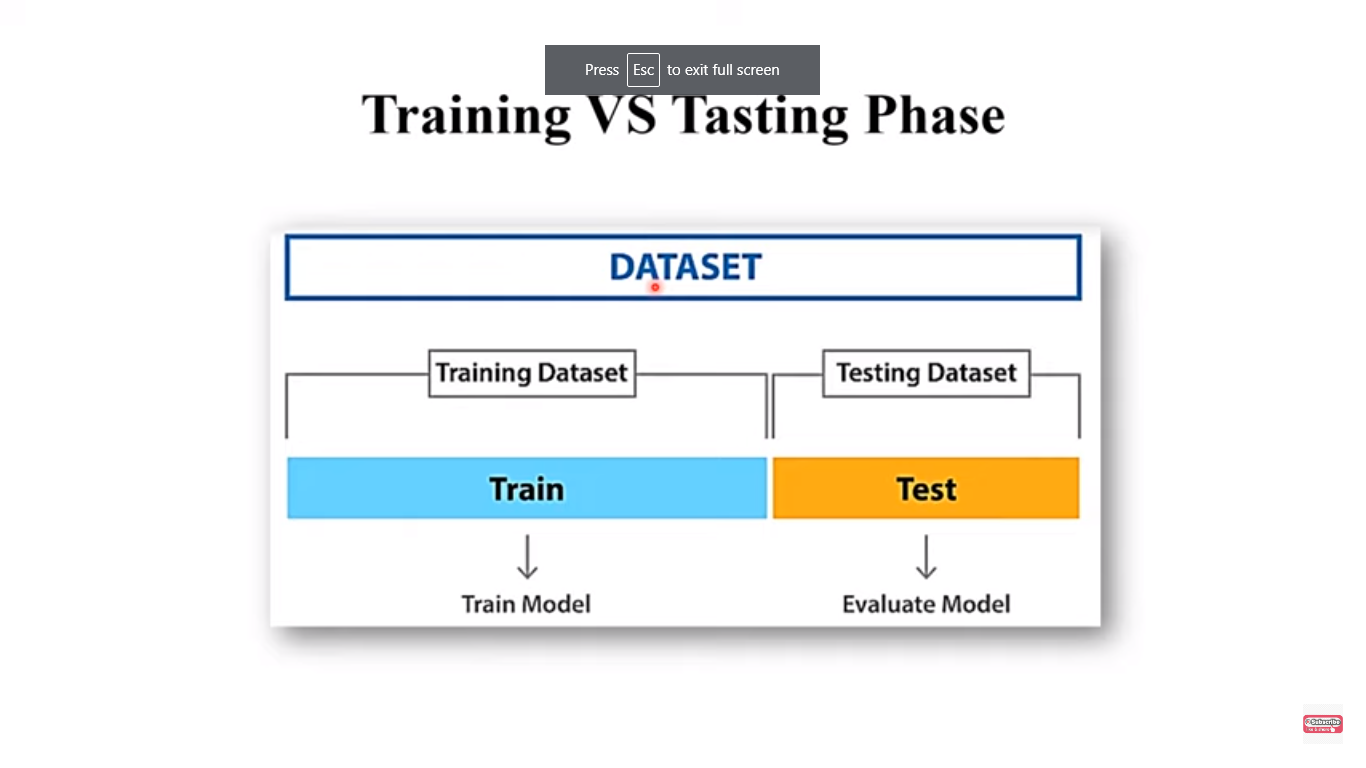
### Applications of Unsupervised Learning

* **Network Analysis:** Unsupervised learning is used for identifying plagiarism and copyright in document network analysis of text data for scholarly articles.
* **Recommendation Systems:** Recommendation systems widely use unsupervised learning techniques for building recommendation applications for different web applications and e-commerce websites.
* **Anomaly Detection:** Anomaly detection is a popular application of unsupervised learning, which can identify unusual data points within the dataset. It is used to discover fraudulent transactions.
* **Singular Value Decomposition:** Singular Value Decomposition or SVD is used to extract particular information from the database. For example, extracting information of each user located at a particular location

**TRAINING AND TESTING:**

What is Training Dataset?

The ***training data is the biggest (in -size) subset of the original dataset, which is used to train or fit the machine learning model***. Firstly, the training data is fed to the ML algorithms, which lets them learn how to make predictions for the given task.



For **Unsupervised learning**, the training data contains unlabeled data points, i.e., inputs are not tagged with the corresponding outputs. Models are required to find the patterns from the given training datasets in order to make predictions.

On the other hand, for supervised learning, the training data contains labels in order to train the model and make predictions.

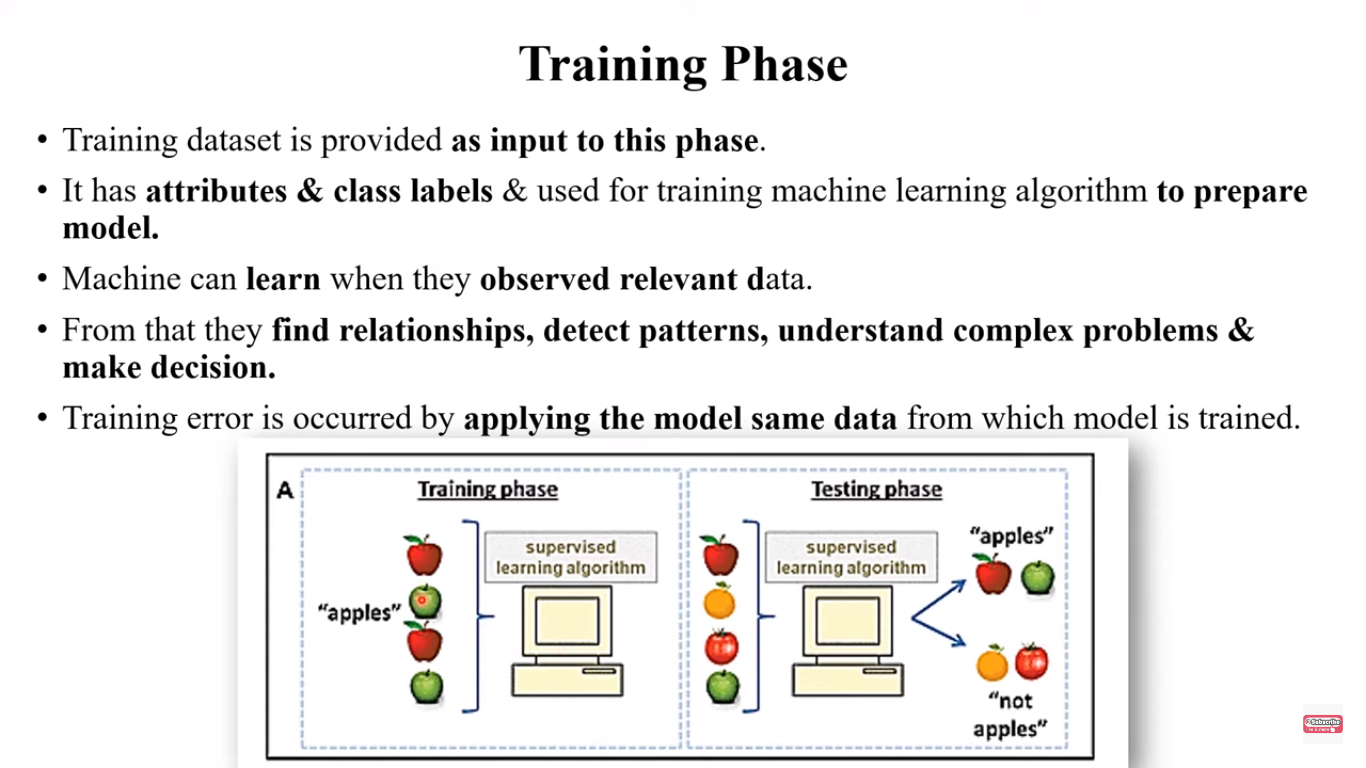
The type of training data that we provide to the model is highly responsible for the model's accuracy and prediction ability. It means that the better the quality of the training data, the better will be the performance of the model. Training data is approximately more than or equal to 60% of the total data for an ML project.

## What is Test Dataset

Once we train the model with the training dataset, it's time to test the model with the test dataset. This dataset evaluates the performance of the model and ensures that the model can generalize well with the new or unseen dataset. ***The test dataset is another subset of original data, which is independent of the training dataset***. However, it has some similar types of features and class probability distribution and uses it as a benchmark for model evaluation once the model training is completed. Test data is a well-organized dataset that contains data for each type of scenario for a given problem that the model would be facing when used in the real world. Usually, the test dataset is approximately 20-25% of the total original data for an ML project.



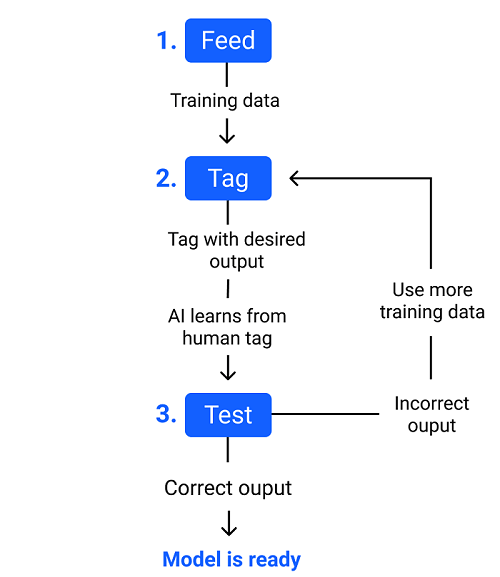
Machine Learning algorithms enable the machines to make predictions and solve problems on the basis of past observations or experiences. These experiences or observations an algorithm can take from the training data, which is fed to it. Further, one of the great things about ML algorithms is that they can learn and improve over time on their own, as they are trained with the relevant training data.



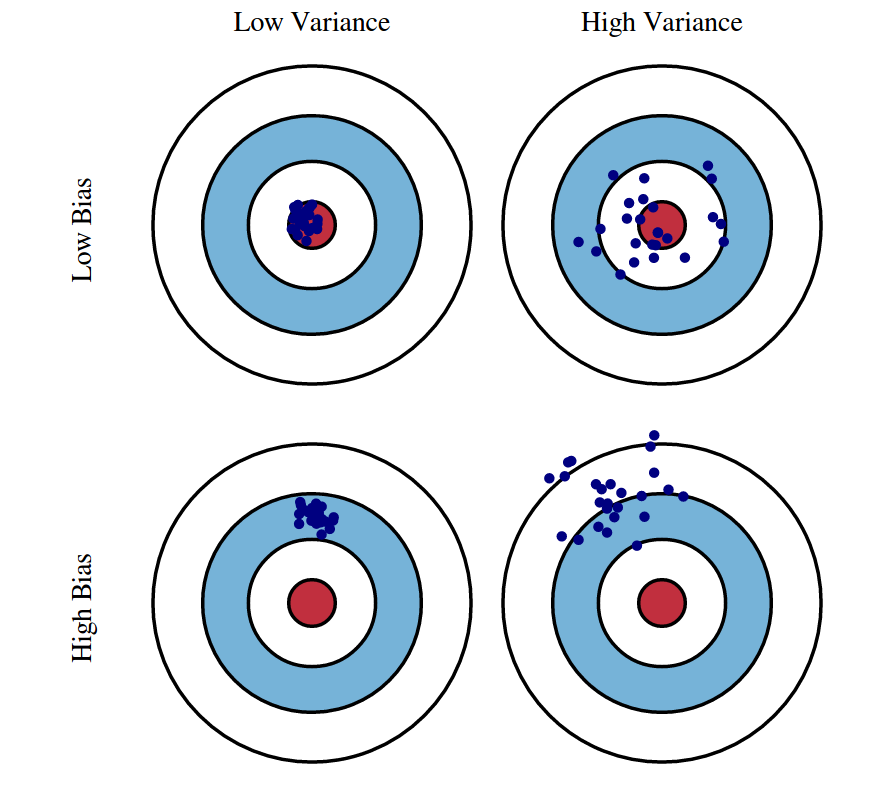
Once the model is trained enough with the relevant training data, it is tested with the test data. We can understand the whole process of training and testing in three steps, which are as follows:

1. **Feed:** Firstly, we need to train the model by feeding it with training input data.
2. **Define:** Now, training data is tagged with the corresponding outputs (in Supervised Learning), and the model transforms the training data into text vectors or a number of data features.
3. **Test:** In the last step, we test the model by feeding it with the test data/unseen dataset. This step ensures that the model is trained efficiently and can generalize well.

The above process is explained using a flowchart given below:

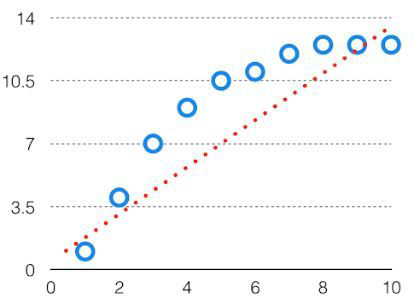


**TRADEOFFS IN STATISTICAL LEARNING**:

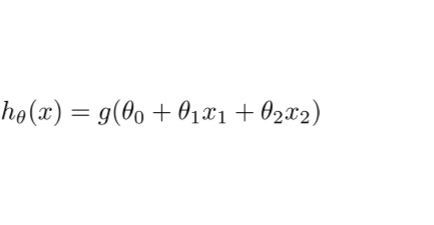


It is important to understand prediction errors (bias and variance) when it comes to accuracy in any machine learning algorithm. There is a tradeoff between a model’s ability to minimize bias and variance which is referred to as the best solution for selecting a value of **Regularization** constant. Proper understanding of these errors would help to avoid the overfitting and underfitting of a data set while training the algorithm.

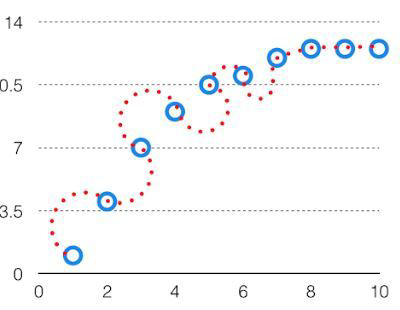
**Bias**  
The bias is known as the difference between the prediction of the values by the ML model and the correct value. Being high in biasing gives a large error in training as well as testing data. Its recommended that an algorithm should always be low biased to avoid the problem of underfitting.  
By high bias, the data predicted is in a straight line format, thus not fitting accurately in the data in the data set. Such fitting is known as **Underfitting of Data**. This happens when the hypothesis is too simple or linear in nature. Refer to the graph given below for an example of such a situation.



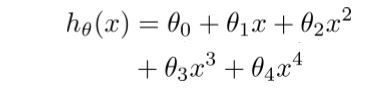
*HighBias*

In such a problem, a hypothesis looks like follows.  
**Variance**  
The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model. The model with high variance has a very complex fit to the training data and thus is not able to fit accurately on the data which it hasn’t seen before. As a result, such models perform very well on training data but has high error rates on test data.  
When a model is high on variance, it is then said to as **Overfitting of Data**. Overfitting is fitting the training set accurately via complex curve and high order hypothesis but is not the solution as the error with unseen data is high.  
While training a data model variance should be kept low.

The high variance data looks like follows.

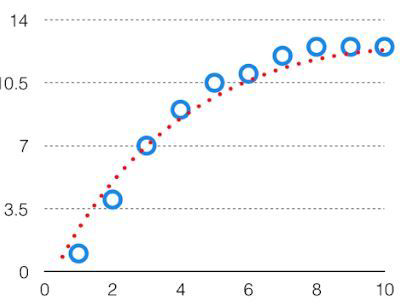


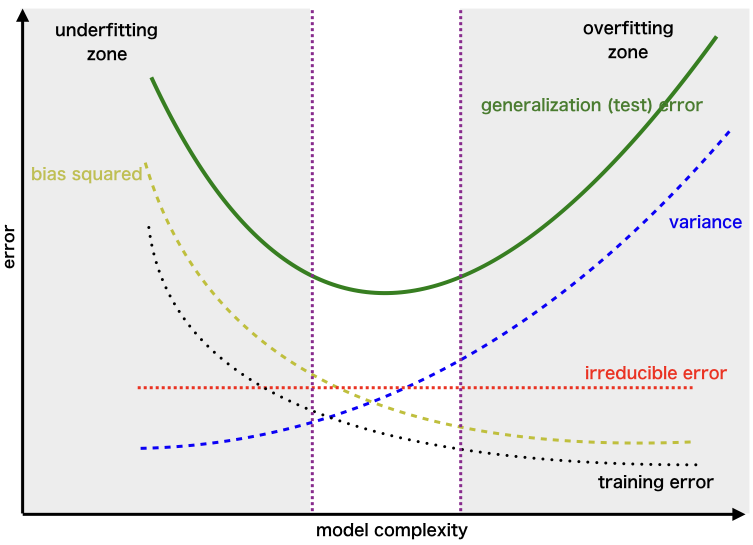
*High Variance*

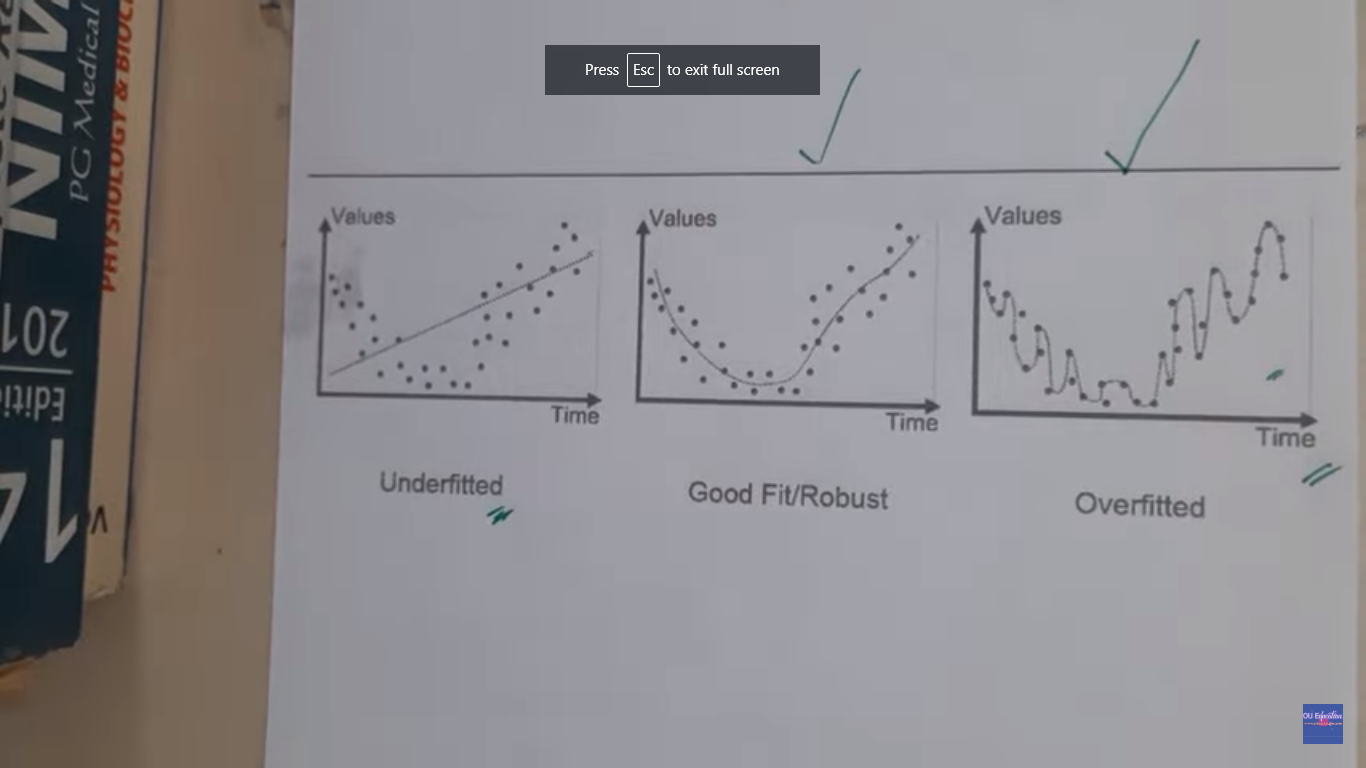
In such a problem, a hypothesis looks like follows.  
  
**Bias Variance Tradeoff**

If the algorithm is too simple (hypothesis with linear eq.) then it may be on high bias and low variance condition and thus is error-prone. If algorithms fit too complex ( hypothesis with high degree eq.) then it may be on high variance and low bias. In the latter condition, the new entries will not perform well. Well, there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

This tradeoff in complexity is why there is a tradeoff between bias and variance. An algorithm can’t be more complex and less complex at the same time. For the graph, the perfect tradeoff will be like.

  
The best fit will be given by hypothesis on the tradeoff point.

The error to complexity graph to show trade-off is given as –  
  
This is referred to as the best point chosen for the training of the algorithm which gives low error in training as well as testing data.



**ESTIMATING RISK STATISTICS:**

Unraveling the genetic background of human diseases serves a number of goals. One aim is to identify genes that modify the susceptibility to disease. In this context, we ask questions like: “Is this genetic variant more frequent in patients with the disease of interest than in unaffected controls?” or “Is the mean phenotype higher in carriers of this genetic variant than in non-carriers?” From the answers, we possibly learn about the pathogenesis of the disease, and we can identify possible targets for therapeutic interventions. Looking back at the past decade, it can be summarized that genome-wide association (GWA) studies have been useful in this endeavor (Hindorff et al. [2012](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3432206/#CR37)).

When we consider classical measures for strength of association on the one hand, such as the odds ratio (OR), and for classification on the other hand, such as sensitivity (sens) and specificity (spec), there is a simple relationship between them with

An external file that holds a picture, illustration, etc.
Object name is 439_2012_1194_Article_IEq1.gif

The overall process of rule construction and evaluation is shown in Fig. [1](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3432206/figure/Fig1/).

[An external file that holds a picture, illustration, etc.
Object name is 439_2012_1194_Fig1_HTML.jpg](https://www.ncbi.nlm.nih.gov/core/lw/2.0/html/tileshop_pmc/tileshop_pmc_inline.html?title=Click%20on%20image%20to%20zoom&p=PMC3&id=3432206_439_2012_1194_Fig1_HTML.jpg)

[Fig. 1](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3432206/figure/Fig1/)

Path to construct, evaluate and validate a rule of classification or probability estimation

**SAMPLING DISTRIBUTION OF AN ESTIMATOR:**

One of the most important concepts discussed in the context of inferential data analysis is the idea of sampling distributions. Understanding sampling distributions helps us better comprehend and interpret results from our descriptive as well as predictive data analysis investigations. Sampling distributions are also frequently used in decision making under uncertainty and hypothesis testing.

What are sampling distributions?

You may already be familiar with the idea of probability distributions. A probability distribution gives us an understanding of the probability and likelihood associated with values (or range of values) that a random variable may assume. A random variable is a quantity whose value (outcome) is determined randomly. Some examples of a random variable include, the monthly revenue of a retail store, the number of customers arriving at a car wash location on any given day, the number of accidents on a certain highway on any given day, weekly sales volume at a retail store, etc.

Sampling distribution of the sample mean

Assuming that *X*represents the data (population), if X has a distribution with average μ and standard deviation σ, and if X is approximately normally distributed or if the sample size *n* is large,

https://miro.medium.com/max/700/1*VaA_ROP5M89rZQmddLZ6-Q.png

The above distribution is only valid if,

X is approximately normal **or**sample size *n* is large, and,

* the data (population) standard deviation σ is known.

If X is normal, then X̅ is also normally distributed regardless of the sample size *n*. [Central Limit Theorem](https://www.simplypsychology.org/central-limit-theorem.html) tells us that even if X is not normal, if the sample size is large enough (usually greater than 30), then X̅’s distribution is approximately normal (Sharpe, De Veaux, Velleman and Wright, 2020, pp. 318–320). If X̅ is normal, we can easily standardize and convert it to the standard normal distribution *Z*.

If the population standard deviation σ is *not* known, we cannot assume that the sample mean X̅ is normally distributed. If certain conditions are satisfied (explained below), then we can transform X̅ to another random variable *t* such that,

https://miro.medium.com/max/83/1*XValJZz4KA0YOVKU1P0b1g.png

The random variable *t* is said to follow the *t*-distribution with *n-1* degrees of freedom, where *n* is the sample size. The t-distribution is bell-shaped and symmetric (just like the normal distribution) but has fatter tails compared to the normal distribution. This means values further away from the mean have a higher likelihood of occurring compared to that in the normal distribution.

The conditions to use the t-distribution for the random variable *t* are as follows (Sharpe et al., 2020, pp. 415–420):

If X is normally distributed, even for small sample sizes (*n<*15), the t-distribution can be used.

If the sample size is between 15 and 40, the t-distribution can be used as long as X is unimodal and reasonably symmetric.

For sample sizes greater than 40, the t-distribution can be used unless X’s distribution is heavily skewed

**Empirical risk minimization (ERM):**

Empirical risk minimization (ERM): It is a principle in statistical learning theory which defines a  
family of learning algorithms and is used to give theoretical bounds on their performance.

The idea is that we don’t know exactly how well an algorithm will work in practice (the true "risk") because we don't know the true distribution of data that the algorithm will work on but as an alternative we can measure its performance on a known set of training data.

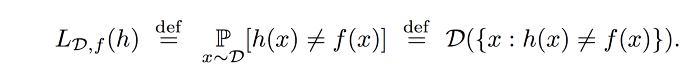
We assumed that our samples come from this distribution and use our dataset as an approximation.  
If we compute the loss using the data points in our dataset, it’s called empirical risk.

It is “empirical”and not “true” because we are using a dataset that’s a subset of the whole population.

When our learning model is built, we have to pick a function that minimizes the empirical risk that is the delta between predicted output and actual output for data points in the dataset.

This process of finding this function is called empirical risk minimization (ERM). We want to minimize the true risk.  
We don’t have information that allows us to achieve that, so we hope that this empirical risk will almost be the same as the true empirical risk.

In the equation below, we can define the**true error**, which is based on the whole domain **X**:



Since we only have access to ***S***, a subset of the input domain, we learn based on that sample of training examples. We don’t have access to the**true error**, but to the **empirical error**:

https://miro.medium.com/max/700/1*3gv-eAjhBxHB4ZGJi5EbjA.png

Let’s get a better understanding by Example.

We would want to build a model that can differentiate between a male and a female based on specific features.

If we select 150 random people where women are really short, and men are really tall, then the model might incorrectly assume that height is the differentiating feature.

For building a truly accurate model, we have to gather all the women and men in the world to extract differentiating features.

Unfortunately, that is not possible! So we select a small number of people and hope that this sample is representative of the whole population.

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

Introduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of th

ntroduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of the

ntroduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of the

ntroduction

Unraveling the genetic background of human diseases

serves a number of goals. One aim is to identify genes that

modify the susceptibility to disease. In this context, we ask

questions like: ‘‘Is this genetic variant more frequent in

patients with the disease of interest than in unaffected

controls?’’ or ‘‘Is the mean phenotype higher in carriers of

this genetic variant than in non-carriers?’’ From the

answers, we possibly learn about the pathogenesis of the