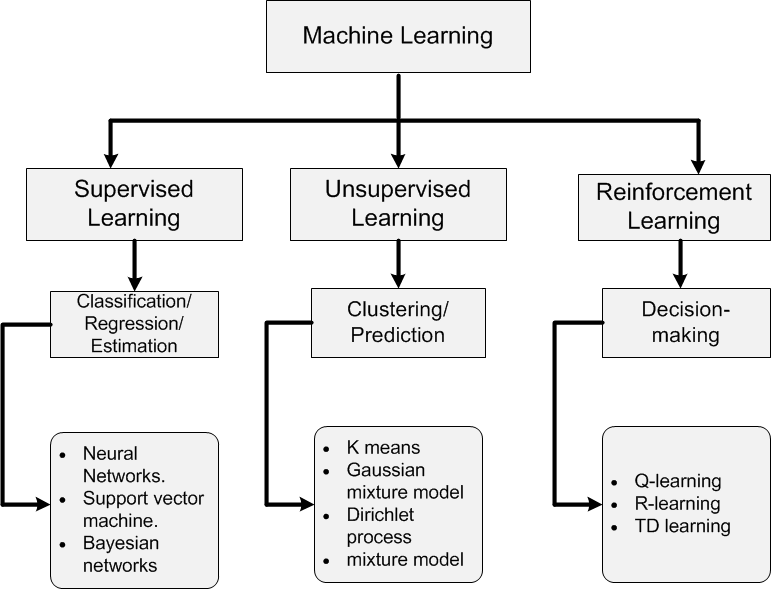
Machine Learning?

Arthur Samuel coined the term Machine Learning in the year 1959. He was a pioneer in Artificial Intelligence and computer gaming, and defined Machine Learning as “Field of study that gives computers the capability to learn without being explicitly programmed”.

In this article, firstly, we will discuss Machine Learning in detail covering different aspects, processes, and applications. Secondly, we will start with understanding the importance of Machine Learning. We will also explain the standard terms used in Machine Learning and the steps to approach an ML problem. Further, we will understand the building blocks of Machine Learning and how does it work. Moreover, we will establish why Python is the best programming language for Machine Learning. We will also list the different types of Machine Learning approaches and industrial applications. Finally, the article ends with the job prospects and career opportunities in the field of Machine Learning with salary trends across top metropolitan cities in India.

Machine Learning is a subset of Artificial Intelligence. Machine Learning is the study of making machines more human-like in their behaviour and decisions by giving them the ability to learn and develop their own programs. This is done with minimum human intervention, i.e., no explicit programming. The learning process is automated and improved based on the experiences of the machines throughout the process. Good quality data is fed to the machines, and different algorithms are used to build ML models to train the machines on this data. The choice of algorithm depends on the type of data at hand, and the type of activity that needs to be automated.



A learning example E in the usual form accepted by CN2 is a pair (A, C), where A is an

attribute-value vector,

Essentials of Machine Learning Algorithms .

Machine Learning has been classfied into three main Algorithms.

1.Supervised Learning.

Supervised Learning

This algoritham contains a target (or) outcome variable (or dependent variable) which is to be predicted from any given set of predictors (independent variables).Using these set of variables, we can generate a function that maps inputs to a desired outputs.This process atands until a model achieves a desired level of accuracy on a training data.

Ex : Regression, Decision Tree, Random Forest,KNN,Logistic Regression etc.

2.Unsupervised Learning. This type of Algorithm we do not have any outcome variable (or) target to predict.It is mainly used for clustering population in different groups, such as widely used for a segmenting customers in different groups for a specfic intervention.: Apriori

3.Reinforcement Learning

This Algorithm the machine is trained to make a specfic decisions.It works in such a way as a Machine is exposed to an envirnoment such that it trains y itself and continually using trail and error.This machine learns from a past ecperience and tries to capture all the best possible knowledge for accurate decisions.

Ex: Markov Decision Proces

such examples, ABCN2 also accepts argumented examples. An argumented example AE is a triple of the form:

AE = (A, C,Arguments).

As usual, A is an attribute-value vector and C is a class value. Arguments is a set of arguments Arg1,...,Argn, where

an argument Argi has one of the following forms:

C because Reasons

or

C despite Reasons

The former specifies a positive argument (speaks for the given class value), while the latter specifies a negative

argument (speaks against the class value). Reasons is a conjunction of reasons r1,...,rn,

Reasons = r1 ∧ r2 ∧···∧ rn

where each of the reasons ri can be in one of five possible forms. In the explanation of these forms below we assume

that ri is a part of a positive argument; for negative arguments, the explanations are exactly the opposite. The five

forms of reasons are:

• X = xi means that value xi of attribute X is the reason why example is in the class as given. This is the only

allowed form for discrete attributes.

• X>xi (or X xi) means that the value of attribute X of example being greater than (greater or equal to) xi is

the reason for class value.

• X<xi (or X xi) the opposite to X>xi (X xi).

• X>(or X ) “X is high”; similar to X>xi (X xi), just that we do not know the threshold value and it has to

be found by ABCN2 automatically. Such an argument says that the value of X of the example is high enough for

the example to be in the class as given.

• X<(or X ); “X is low”, the opposite of X > (X ).

For example, the expert’s argument for approving credit to Mrs. Brown (see Table 1) can be: Mrs. Brown received

credit because she is rich. A negative argument can be: Mrs. Brown received credit despite her not paying regularly.

The Brown example would in our syntax be written as:

((PaysRegularly = no,Rich = yes,HairColor = blond),

CreditApproved = yes,

{CreditApproved = yes because Rich = yes,

CreditApproved = yes despite PaysRegularly = no}).

Arguments given to examples additionally constrain rules covering this example. Remember that in CN2, rules have

the form:

IF Complex THEN Class

926 M. Možina et al. / Artificial Intelligence 171 (2007) 922–937

where Complex is the conjunction of simple conditions, called selectors. For the purpose of this paper, a selector

simply specifies the value of an attribute, for example HairColor = blond or a threshold on an attribute value, for

example Salary > 5000. A rule for our credit approval domain can be:

IF PaysRegularly = no AND HairColor = blond THEN CreditApproved = yes

The condition part of the rule is satisfied by the attribute values of Mrs. Brown example, so we say that this rule

covers this example.

A rule R is consistent with an argument “C because Reasons” (or “C despite Reasons”), if for all reasons ri of

Reasons it is true that:

(1) If the reason ri is in one of the forms: “X = xi” or “X>xi” or “X<xi” (“X xi” or “X xi”), then exactly

the same selector needs to be present in the complex of the rule R.

(2) If the reason ri has the form “X >” (or “X <”, “X ”, “X ”), then the complex of the rule R needs to contain

a selector “X>xi” (or “X<xi”, “X xi”, “X xi”). The threshold value xi does not matter for consistency.

With the use of arguments in examples, the definition of a rule covering an example needs to be refined. In the

standard definition, a rule covers an example if the condition part of the rule is true for this example. In argument

based rule learning, this definition is modified to: A rule R AB-covers an argumented example E if:

(1) All the conditions in R are true for E (same as in CN2),

(2) R is consistent with at least one positive argument of E, and

(3) R is not consistent with any of the negative arguments of E.

As an illustration of the differences between AB-covering and the usual definition of covering, consider again the

Brown example with the argument that she received credit because she is rich and despite her not paying regularly.

Now consider four rules:

R1: IF HairColor = blond THEN CreditApproved = yes

R2: IF PaysRegularly = no AND HairColor = blond THEN CreditApproved = yes

R3: IF PaysRegularly = no AND Rich = yes THEN CreditApproved = yes

R4: IF HairColor = blond AND Rich = yes THEN CreditApproved = yes

Python and R Codes

There have been dozens of articles for writing and comparing a python and R Data Codes for a subjective point of view.

Common Machine Learning Algorithms

1.Linear Regression.

Linear Regression is used to estimate the real values based on continuous variables such as (cost of houses,number of calls,total sales etc.)

2.Logistic Regression.

Logistic Regression is a classification not a regression algorithm.It is used to estimate discrete values as Biary 0/1,yes/no,true/false.

3.Decision Tree

It is a type of supervised learning algoritham that is mostly used for classfication of problems.

It works for both categorical and continuous dependent variables.Here in this algoritham we can split the population into 2 or more homogeneous sets.

4.SVM (Support Vector Machine)

This algoritham is called as classfication method.

Here we plot each data item as apoint in n-dimensional space with a value of each feature being value of a particular coordinate.

5.Naive Bayes

This algorithm is based on Bayes' theorem with an assumption of and ndependence predictors.

Naive Bayesian model is an easy to build and particularly useful for very large data sets.

6.kNN (k- Nearest Neighbors)

It can be used for classfication and Regression problems.

K nearest neighbours is a simple algorithm that stores all available cases and classifies.

7.k-Means

It is a type of unsupervised which sloves the clustering problem

R-Code library(cluster) fit <- kmeans(X, 3) # 5 cluster solution

8.Random Forest.

Random Forest is a trademark term for an ensemble of decision trees.

R-Code

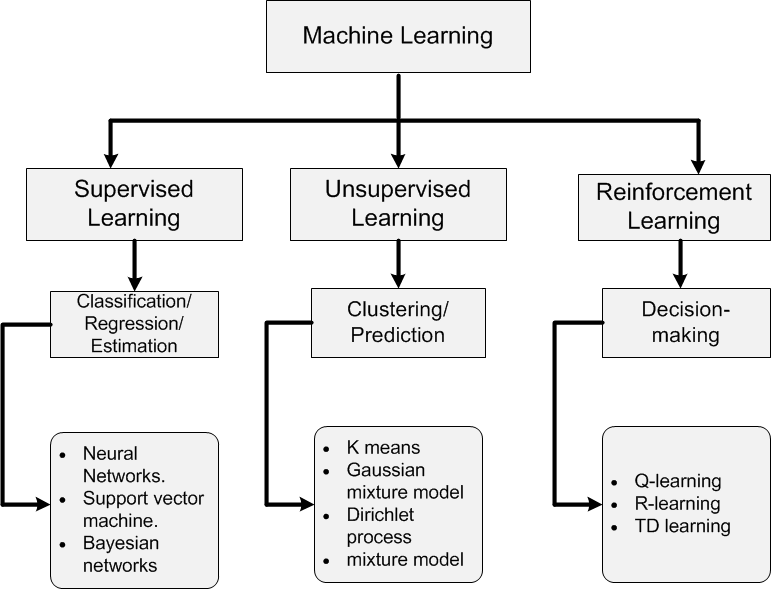
library(randomForest) x <- cbind(x\_train,y\_train) # Fitting model fit <- randomForest(Species ~ ., x,ntree=500) summary(fit) #Predict Output predicted= predict(fit,x\_test)

What is Machine Learning?

Arthur Samuel coined the term Machine Learning in the year 1959. He was a pioneer in Artificial Intelligence and computer gaming, and defined Machine Learning as “Field of study that gives computers the capability to learn without being explicitly programmed”.

In this article, firstly, we will discuss Machine Learning in detail covering different aspects, processes, and applications. Secondly, we will start with understanding the importance of Machine Learning. We will also explain the standard terms used in Machine Learning and the steps to approach an ML problem. Further, we will understand the building blocks of Machine Learning and how does it work. Moreover, we will establish why Python is the best programming language for Machine Learning. We will also list the different types of Machine Learning approaches and industrial applications. Finally, the article ends with the job prospects and career opportunities in the field of Machine Learning with salary trends across top metropolitan cities in India.

Machine Learning is a subset of Artificial Intelligence. Machine Learning is the study of making machines more human-like in their behaviour and decisions by giving them the ability to learn and develop their own programs. This is done with minimum human intervention, i.e., no explicit programming. The learning process is automated and improved based on the experiences of the machines throughout the process. Good quality data is fed to the machines, and different algorithms are used to build ML models to train the machines on this data. The choice of algorithm depends on the type of data at hand, and the type of activity that needs to be automated.



A learning example E in the usual form accepted by CN2 is a pair (A, C), where A is an

attribute-value vector,

Essentials of Machine Learning Algorithms .

Machine Learning has been classfied into three main Algorithms.

1.Supervised Learning.

Supervised Learning

This algoritham contains a target (or) outcome variable (or dependent variable) which is to be predicted from any given set of predictors (independent variables).Using these set of variables, we can generate a function that maps inputs to a desired outputs.This process atands until a model achieves a desired level of accuracy on a training data.

Ex : Regression, Decision Tree, Random Forest,KNN,Logistic Regression etc.

2.Unsupervised Learning. This type of Algorithm we do not have any outcome variable (or) target to predict.It is mainly used for clustering population in different groups, such as widely used for a segmenting customers in different groups for a specfic intervention.: Apriori

3.Reinforcement Learning

This Algorithm the machine is trained to make a specfic decisions.It works in such a way as a Machine is exposed to an envirnoment such that it trains y itself and continually using trail and error.This machine learns from a past ecperience and tries to capture all the best possible knowledge for accurate decisions.

Ex: Markov Decision Proces

such examples, ABCN2 also accepts argumented examples. An argumented example AE is a triple of the form:

AE = (A, C,Arguments).

As usual, A is an attribute-value vector and C is a class value. Arguments is a set of arguments Arg1,...,Argn, where

an argument Argi has one of the following forms:

C because Reasons

or

C despite Reasons

The former specifies a positive argument (speaks for the given class value), while the latter specifies a negative

argument (speaks against the class value). Reasons is a conjunction of reasons r1,...,rn,

Reasons = r1 ∧ r2 ∧···∧ rn

where each of the reasons ri can be in one of five possible forms. In the explanation of these forms below we assume

that ri is a part of a positive argument; for negative arguments, the explanations are exactly the opposite. The five

forms of reasons are:

• X = xi means that value xi of attribute X is the reason why example is in the class as given. This is the only

allowed form for discrete attributes.

• X>xi (or X xi) means that the value of attribute X of example being greater than (greater or equal to) xi is

the reason for class value.

• X<xi (or X xi) the opposite to X>xi (X xi).

• X>(or X ) “X is high”; similar to X>xi (X xi), just that we do not know the threshold value and it has to

be found by ABCN2 automatically. Such an argument says that the value of X of the example is high enough for

the example to be in the class as given.

• X<(or X ); “X is low”, the opposite of X > (X ).

For example, the expert’s argument for approving credit to Mrs. Brown (see Table 1) can be: Mrs. Brown received

credit because she is rich. A negative argument can be: Mrs. Brown received credit despite her not paying regularly.

The Brown example would in our syntax be written as:

((PaysRegularly = no,Rich = yes,HairColor = blond),

CreditApproved = yes,

{CreditApproved = yes because Rich = yes,

CreditApproved = yes despite PaysRegularly = no}).

Arguments given to examples additionally constrain rules covering this example. Remember that in CN2, rules have

the form:

IF Complex THEN Class

926 M. Možina et al. / Artificial Intelligence 171 (2007) 922–937

where Complex is the conjunction of simple conditions, called selectors. For the purpose of this paper, a selector

simply specifies the value of an attribute, for example HairColor = blond or a threshold on an attribute value, for

example Salary > 5000. A rule for our credit approval domain can be:

IF PaysRegularly = no AND HairColor = blond THEN CreditApproved = yes

The condition part of the rule is satisfied by the attribute values of Mrs. Brown example, so we say that this rule

covers this example.

A rule R is consistent with an argument “C because Reasons” (or “C despite Reasons”), if for all reasons ri of

Reasons it is true that:

(1) If the reason ri is in one of the forms: “X = xi” or “X>xi” or “X<xi” (“X xi” or “X xi”), then exactly

the same selector needs to be present in the complex of the rule R.

(2) If the reason ri has the form “X >” (or “X <”, “X ”, “X ”), then the complex of the rule R needs to contain

a selector “X>xi” (or “X<xi”, “X xi”, “X xi”). The threshold value xi does not matter for consistency.

With the use of arguments in examples, the definition of a rule covering an example needs to be refined. In the

standard definition, a rule covers an example if the condition part of the rule is true for this example. In argument

based rule learning, this definition is modified to: A rule R AB-covers an argumented example E if:

(1) All the conditions in R are true for E (same as in CN2),

(2) R is consistent with at least one positive argument of E, and

(3) R is not consistent with any of the negative arguments of E.

As an illustration of the differences between AB-covering and the usual definition of covering, consider again the

Brown example with the argument that she received credit because she is rich and despite her not paying regularly.

Now consider four rules:

R1: IF HairColor = blond THEN CreditApproved = yes

R2: IF PaysRegularly = no AND HairColor = blond THEN CreditApproved = yes

R3: IF PaysRegularly = no AND Rich = yes THEN CreditApproved = yes

R4: IF HairColor = blond AND Rich = yes THEN CreditApproved = yes

Python and R Codes

There have been dozens of articles for writing and comparing a python and R Data Codes for a subjective point of view.

Common Machine Learning Algorithms

1.Linear Regression.

Linear Regression is used to estimate the real values based on continuous variables such as (cost of houses,number of calls,total sales etc.)

2.Logistic Regression.

Logistic Regression is a classification not a regression algorithm.It is used to estimate discrete values as Biary 0/1,yes/no,true/false.

3.Decision Tree

It is a type of supervised learning algoritham that is mostly used for classfication of problems.

It works for both categorical and continuous dependent variables.Here in this algoritham we can split the population into 2 or more homogeneous sets.

4.SVM (Support Vector Machine)

This algoritham is called as classfication method.

Here we plot each data item as apoint in n-dimensional space with a value of each feature being value of a particular coordinate.

5.Naive Bayes

This algorithm is based on Bayes' theorem with an assumption of and ndependence predictors.

Naive Bayesian model is an easy to build and particularly useful for very large data sets.

6.kNN (k- Nearest Neighbors)

It can be used for classfication and Regression problems.

K nearest neighbours is a simple algorithm that stores all available cases and classifies.

7.k-Means

It is a type of unsupervised which sloves the clustering problem

R-Code library(cluster) fit <- kmeans(X, 3) # 5 cluster solution

8.Random Forest.

Random Forest is a trademark term for an ensemble of decision trees.

R-Code

library(randomForest) x <- cbind(x\_train,y\_train) # Fitting model fit <- randomForest(Species ~ ., x,ntree=500) summary(fit) #Predict Output predicted= predict(fit,x\_test)

Jkm m,,,l