1.1 Types of Machine Learning Systems

At a broad level, machine learning can be classified into four types:

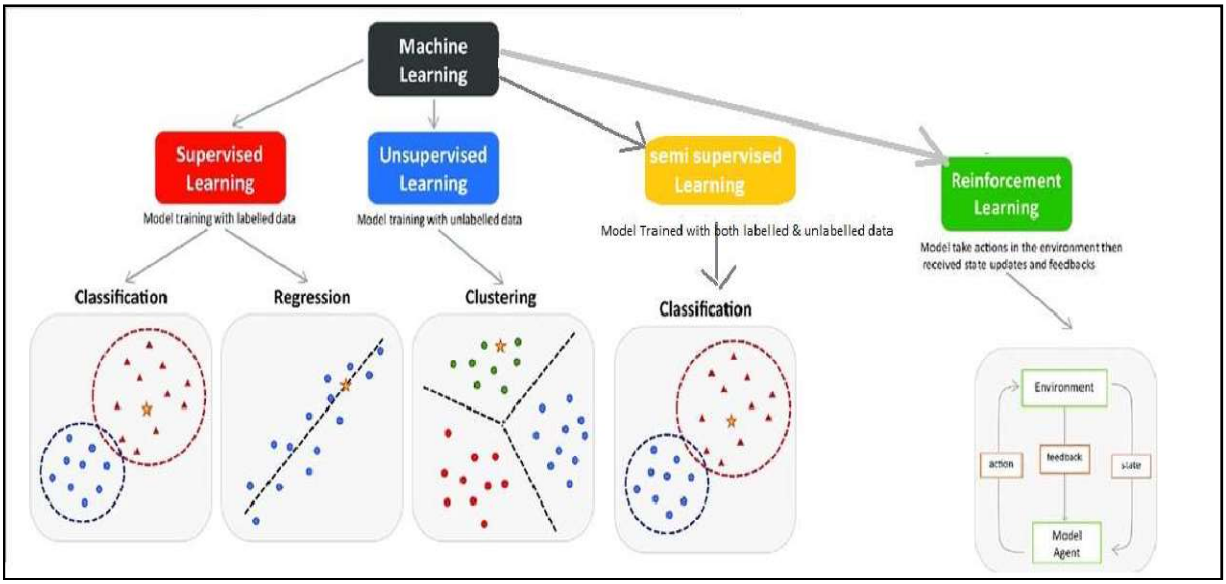

Fig1: Types of Machine Learning Techniques

1. **Supervised learning**
2. **Unsupervised learning**
3. **Semi supervised learning**
4. **Reinforcement learning**

1) Supervised Learning

Supervised learning is a type of machine learning method in which we provide sample labeled data to the machine learning system in order to train it, and on that basis, it predicts the output. The system creates a model using labeled data to understand the datasets and learn about each data, once the training and processing are done then we test the model by providing a sample data to check whether it is predicting the exact output or not.

For Example:

- Let us consider images that are labeled a spoon or a knife. This known data is fed to the machine, which analyzes and learns the association of these images based on its features such as shape, size, sharpness, etc.
- Now when a new image is fed to the machine without any label, the machine is able to predict accurately that it is a spoon with the help of the past data.

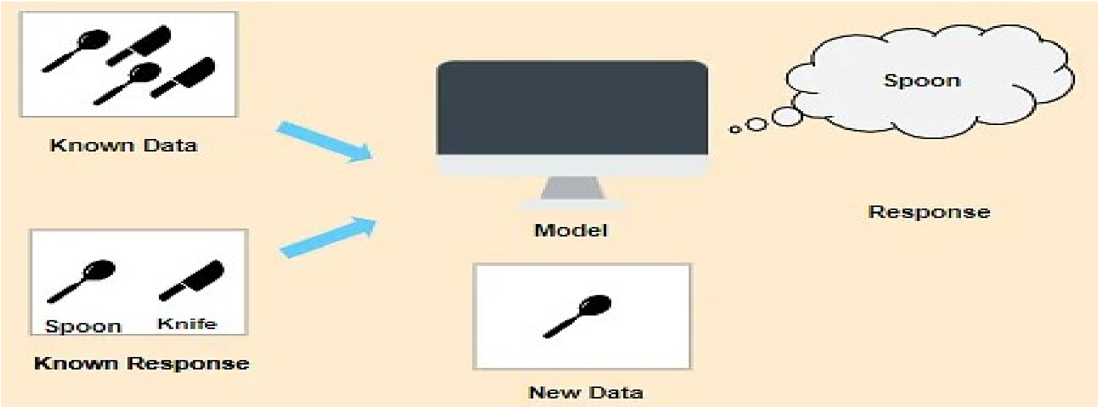

Fig2: Example for Supervised learning algorithm

The goal of supervised learning is to map input data with the output data. The supervised learning is based on supervision, and it is the same as when a student learns things in the supervision of the teacher. The example of supervised learning is **spam filtering**.

Supervised learning can be grouped further in two categories of algorithms:

- **Classification**
- **Regression**

Classification - Supervised Learning

- Classification is used when the output variable is **categorical** i.e. with 2 or more classes.
- For example, yes or no, male or female, true or false, etc.

Example: Spam Filtering

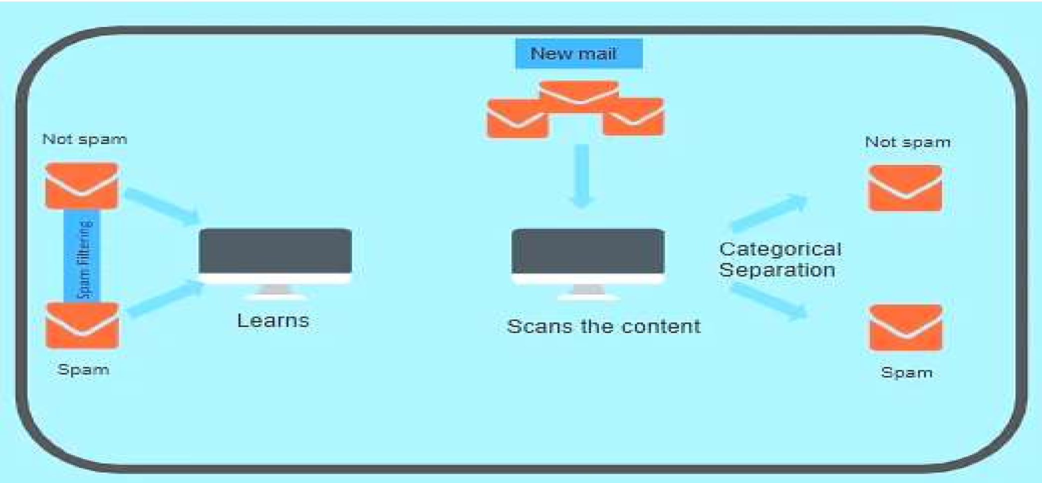

Fig3: Example for classification

- In order to predict whether a mail is spam or not, we need to first teach the machine what a spam mail is.
- This is done based on a lot of spam filters - reviewing the content of the mail, reviewing the mail header, and then searching if it contains any false information.
- Certain keywords and blacklist filters that blackmails are used from already blacklisted spammers.
- All of these features are used to score the mail and give it a spam score. The lower the total spam score of the email, the more likely that it is not a scam.
- Based on the content, label, and the spam score of the new incoming mail, the algorithm decides whether it should land in the inbox or spam folder.

Regression - Supervised Learning

- Regression is used when the output variable is a **real or continuous value**. In this case, there is a relationship between two or more variables i.e., a change in one variable is associated with a change in the other variable.
- For example, salary based on work experience or weight based on height, etc.

Example: humidity and temperature

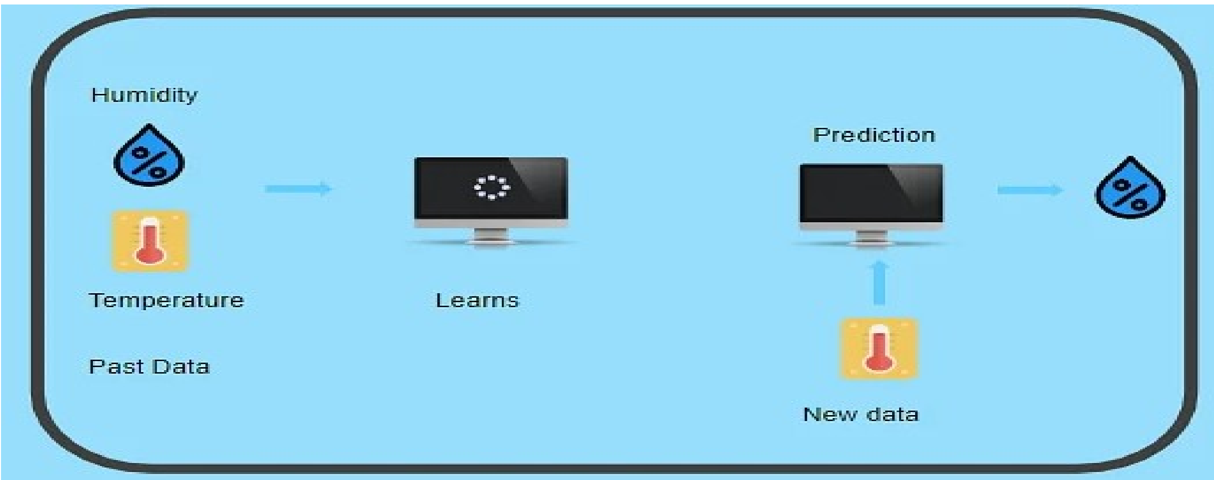

Fig4: Example for Regression

Let's consider two variables -humidity and temperature. Here, 'temperature' is the independent variable and 'humidity' is the dependent variable. If the temperature increases, then the humidity decreases.

These two variables are fed to the model and the machine learns the relationship between them. After the machine is trained, it can easily predict the humidity based on the given temperature.

Some of the supervised learning applications are:

- Sentiment analysis (Twitter, Facebook, Netflix, YouTube, etc)
- Natural Language Processing
- Image classification
- Predictive analysis
- Pattern recognition
- Spam detection
- Speech/Sequence processing

Supervised Learning: Uses

- Prediction of future cases: Use the rule to predict the output for future inputs
- Knowledge extraction: The rule is easy to understand
- Compression: The rule is simpler than the data it explains
- Outlier detection: Exceptions that are not covered by the rule, e.g., fraud

Supervised Learning: Limitations

- Slow -it requires human experts to manually label training examples one by one
- Costly -a model should be trained on the large volumes of hand-labeled data to provide accurate predictions.

2) Unsupervised Learning

Unsupervised learning is a learning method in which a machine learns without any supervision. The training is provided to the machine with the set of data that has not been labeled, classified, or categorized, and the algorithm needs to act on that data without any supervision. The goal of unsupervised learning is to restructure the input data into **new features or a group of objects with similar patterns.**

Example:

- Let's take a similar example as before, but this time we do not tell the machine whether it's a spoon or a knife.
- The machine identifies patterns from the given set and groups them based on their patterns, similarities, etc.

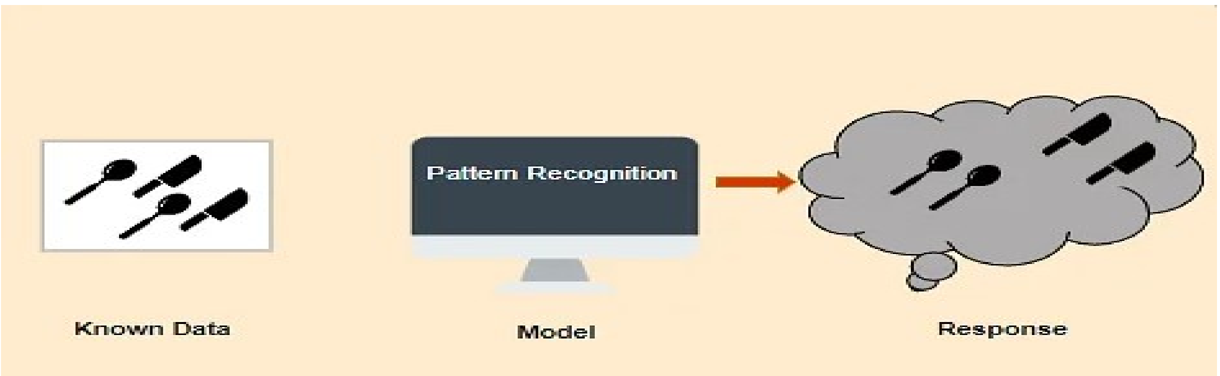

Fig5: Example for Unsupervised Learning

In unsupervised learning, we don't have a predetermined result. The machine tries to find useful insights from the huge amount of data. It can be further classifieds into two categories of algorithms:

- **Clustering**
- **Association**

Clustering - Unsupervised Learning

- Clustering is the method of dividing the objects into clusters that are similar between them and are dissimilar to the objects belonging to another cluster.
- For example, finding out which customers made similar product purchases.

Example:

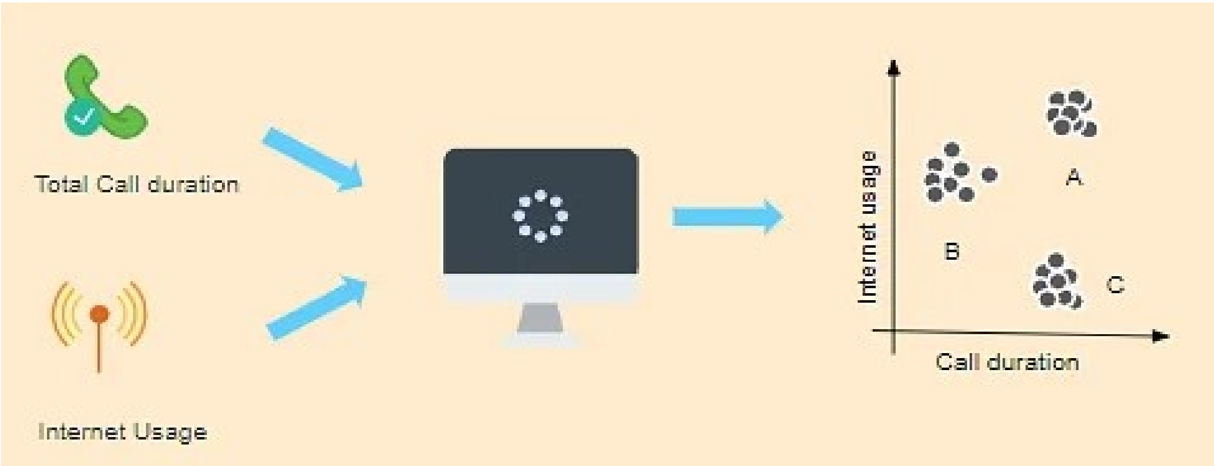

Fig6: Example for Clustering

- Suppose a telecom company wants to reduce its customer churn rate by providing personalized call and data plans.
- The behavior of the customers is studied and the model segments the customers with similar traits.
- Group A customers use more data and also have high call durations. Group B customers are heavy Internet users, while Group C customers have high call duration.
- So, Group B will be given more data benefit plants, while Group C will be given cheaper called call rate plans and group A will be given the benefit of both.

Association - Unsupervised Learning

- Association is a **rule-based machine learning** to discover the probability of the co-occurrence of items in a collection.
- For example, finding out which products were purchased together.

Example:

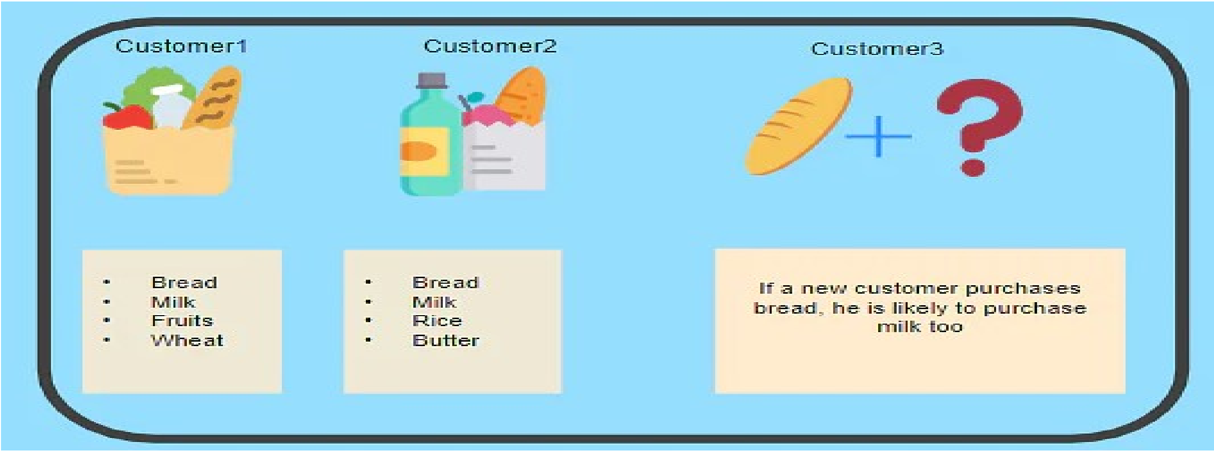

Fig7: Example for Association

- Let's say that a customer goes to a supermarket and buys bread, milk, fruits, and wheat. Another customer comes and buys bread, milk, rice, and butter.
- Now, when another customer comes, it is highly likely that if he buys bread, he will buy milk too.
- Hence, a relationship is established based on customer behavior and recommendations are made.

Uses of Unsupervised Learning

- Unsupervised learning is used for more complex tasks as compared to supervised learning because, in unsupervised learning, we don't have labeled input data.
- Unsupervised learning is preferable as it is easy to get unlabeled data in comparison to labeled data.

Unsupervised Learning- Limitations

- It has a limited area of applications, mostly for clustering purposes.
- It provides less accurate results.

An example of a clustering algorithm is k-Means where *k* refers to the number of clusters to discover in the data.

Unsupervised Learning applications are:

1. Similarity detection
2. Automatic labeling
3. Object segmentation (such as Person, Animal, Films)

The goal in such unsupervised learning problems may be to discover groups of similar examples within the data, where it is called clustering, or to determine the distribution of data within the input space, known as density estimation, or to project the data from a high-dimensional space down to two or three dimensions for the purpose of visualization.

3) Semi-Supervised Learning

Semi-supervised learning is supervised learning where the training data contains very few labeled examples and a large number of unlabeled examples.
The goal of a semi-supervised learning model is to make effective use of all of the available data, not just the labeled data like in supervised learning.

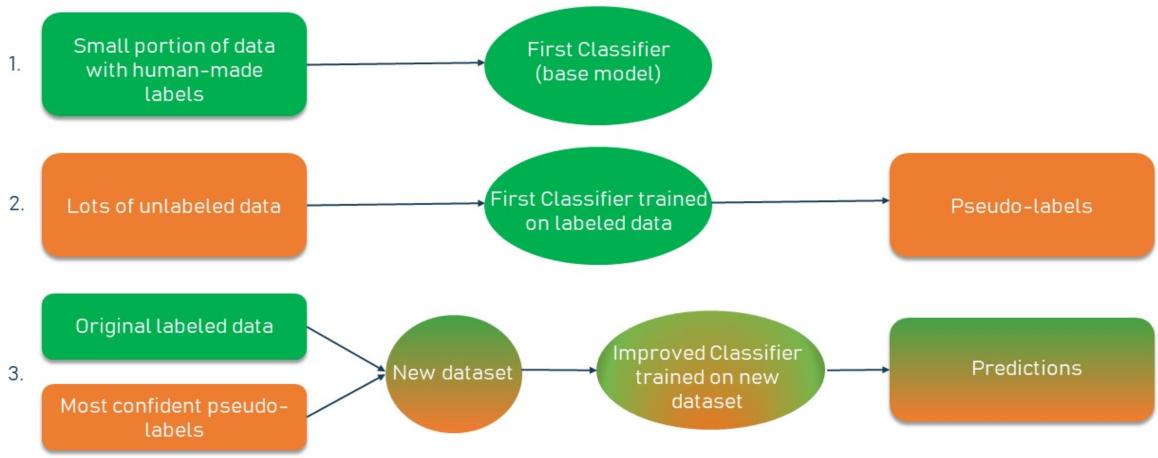

Fig8: Example for Semi supervised Learning

- Semi-supervised learning is an important category that lies between the Supervised and Unsupervised machine learning.
- **To overcome the drawbacks of supervised learning and unsupervised learning algorithms, the concept of Semi-supervised learning is introduced**.
- Labeled data exists with a very small amount while it consists of a huge amount of unlabeled data.
- Initially, similar data is clustered along with an unsupervised learning algorithm, and further, it helps to label the unlabeled data into labeled data.
- It is why label data is a comparatively, more expensive acquisition than unlabeled data.

"Semisupervised" learning attempts to improve the accuracy of supervised learning by exploiting information in unlabeled data. This sounds like magic, but it can work!

Real-world applications of Semi-supervised Learning

- Speech Analysis
- Web content classification
- Protein sequence classification
- Text document classifier

4) Reinforcement Learning

Reinforcement learning is learning what to do — how to map situations to actions—so as to maximize a numerical reward signal. The learner is not told which actions to take, but instead must discover which actions yield the most reward by trying them.

Terms used in Reinforcement Learning

- Environment — Physical world in which the agent operates
- State — Current situation of the agent
- Reward — Feedback from the environment
- Policy — Method to map agent's state to actions
- Value — Future reward that an agent would receive by taking an action in a particular state

Reinforcement learning is a **feedback-based learning method**, in which a learning agent gets a reward for each right action and gets a penalty for each wrong action. The agent learns automatically with these feedbacks and improves its performance. In reinforcement learning, the agent interacts with the environment and explores it. The goal of an agent is to get the most reward points, and hence, it improves its performance.

The robotic dog, which automatically learns the movement of his arms, is an example of Reinforcement learning.

An example of a reinforcement problem is playing a game where the agent has the goal of getting a high score and can make moves in the game and received feedback in terms of punishments or rewards.

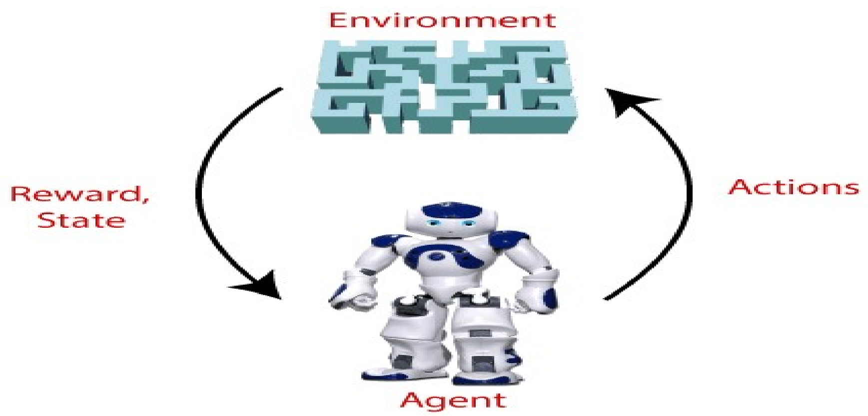

Fig9: Example for Reinforcement learning

In many complex domains, reinforcement learning is the only feasible way to train a program to perform at high levels. For example, in game playing, it is very hard for a human to provide accurate and consistent evaluations of large numbers of positions, which would be needed to train an evaluation function directly from examples. Instead, the program can be told when it has won or lost, and it can use this information to learn an evaluation function that gives reasonably accurate estimates of the probability of winning from any given position.

- Suppose there is an AI agent present within a maze environment, and his goal is to find the diamond.

- The agent interacts with the environment by performing some actions, and based on those actions, the state of the agent gets changed, and it also receives a reward or penalty as feedback.
- The agent continues doing these three things (**take action, change state/remain in the same state, and get feedback**), and by doing these actions, he learns and explores the environment.
- The agent learns that what actions lead to positive feedback or rewards and what actions lead to negative feedback penalty.
- As a positive reward, the agent gets a positive point, and as a penalty, it gets a negative point.

Parameters	**Supervised Learning**	**Unsupervised Learning**	**Semi-Supervised Learning**	**Reinforcement Learning**
Definition	Learns by using labeled data	Trained using unlabeled data without any guidance	Trained using both labeled& unlabeled data	Works on interacting with the environment
Type of data	Labeled data	Unlabeled data	Both	No –predefined data
Type of problems	Regression and classification	Association and Clustering	Classification and Regression	Exploitation or Exploration
Supervision	Yes	No		No supervision
Algorithms	Linear Regression, Logistic Regression, SVM, KNN etc.	K –Means	Text document classifier	Q –Learning, SARSA
Aim	Calculate outcomes	Discover underlying patterns	Classify the data and also discovers underlying patterns	Learn a series of action
Application	Risk Evaluation, Forecast Sales	Recommendation System, Anomaly Detection	Text Classification	Self Driving Cars, Gaming, Healthcare

Table1: Comparison of Machine Learning Techniques

1.4 Batch and Online Learning

Batch learning represents the training of machine learning models in a batch manner. In other words, batch learning represents the training of the models at regular intervals such as weekly, bi-weekly, monthly, quarterly, etc. The data gets accumulated over a period of time. The models then get trained with the accumulated data from time to time at periodic intervals.

Batch learning is also called **offline learning**. The models trained using batch or offline learning are moved into production only at regular intervals based on the performance of models trained with new data.

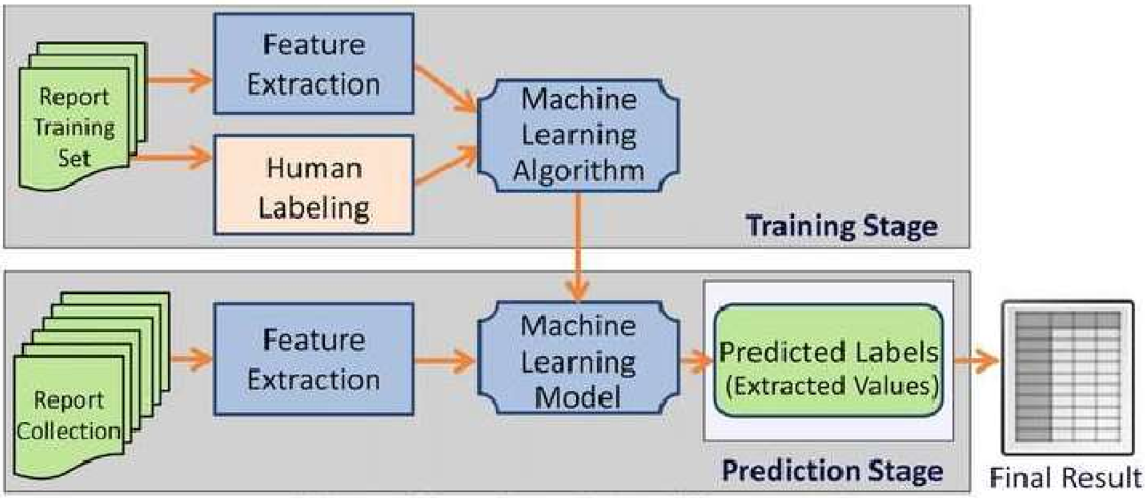

Fig10: Process of Batch Learning

Building offline models or models trained in a batch manner requires training the models with the entire training data set. Improving the model performance would require re-training all over again with the entire training data set. These models are static in nature which means that once they get trained; their performance will not improve until a new model gets re-trained. Offline models or models trained using batch learning are deployed in the production environment by replacing the old model with the newly trained model.

There can be various reasons why we can choose to adopt batch learning for training the models. Some of these reasons are the following:

- The business requirements do not require frequent learning of models.
- The data distribution is not expected to change frequently. Therefore, batch learning is suitable.
- The software systems (big data) required for batch learning is not available due to various reasons including the cost. The fact that the model is trained with a lot of accumulated data takes a lot of time and resources (CPU, memory space, disk space, disk I/O, network I/O, etc.).
- The expertise required for creating the system for incremental learning is not available.

If the models trained using batch learning needs to learn about new data, the models need to be retrained using the new data set and replaced appropriately with the model already in production based on different criteria such as model performance. The whole process of batch learning can be automated as well. The disadvantage of batch learning is it takes a lot of time and resources to re-training the model.

The criteria based on which the machine learning models can be decided to train in a batch manner depends on the model performance. Red-amber-green statuses can be used to determine the health of the model based on the prediction accuracy or error rates. Accordingly, the models can be chosen to be retrained or otherwise.

The following stakeholders can be involved in reviewing the model performance and leveraging batch learning:

- Business/product owners
- Product managers
- Data scientists
- ML engineers

In online learning, the training happens in an incremental manner by continuously feeding data as it arrives or in a small group. Each learning step is fast and cheap, so the system can learn about new data on the fly, as it arrives.

Online learning is great for machine learning systems that receive data as a continuous flow (e.g., stock prices) and need to adapt to change rapidly or autonomously. It is also a good option if you have limited computing resources: once an online learning system has learned about new data instances, it does not need them anymore, so you can discard them (unless you want to be able to roll back to a previous state and "replay" the data) or move the data to another form of storage (warm or cold storage) if you are using the data lake. This can save a huge amount of space and cost.

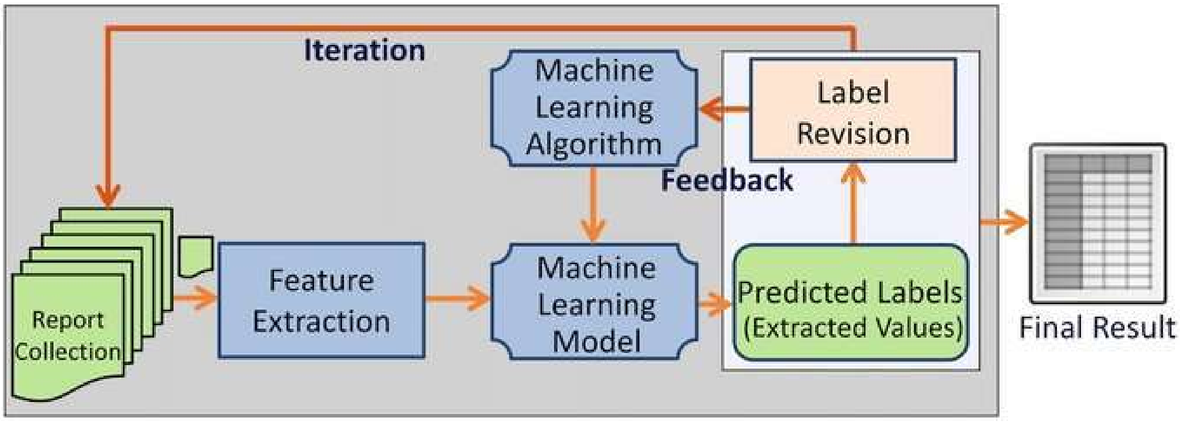

Fig11: Process of Online Learning

Online learning algorithms can also be used to train systems on huge datasets that cannot fit in one machine's main memory (this is also called ***out-of-core learning***). The algorithm loads part of the data runs a training step on that data and repeats the process until it has run on all of the data.

One of the key aspects of online learning is the **learning rate.** The rate at which you want your machine learning to adapt to new data set is called the learning rate. A system with a high learning rate will tend to forget the learning quickly. A system with a low learning rate will be more like batch learning.

One of the big disadvantages of an online learning system is that if it is fed with bad data, the system will have bad performance and the user will see the impact instantly. Thus, it is very important to come up with appropriate data governance strategy to ensure that the data fed is of high quality. In addition, it is very important to monitor the performance of the machine learning system in a very close manner.

The following are some of the challenges for adopting an online learning method:

- Data governance
- Model governance includes appropriate algorithm and model selection on-the-fly

Online models require only a single deployment in the production setting and they evolve over a period of time. The disadvantage that the online models have is that they don't have the entire dataset available for the training. The models are trained in an incremental manner based on the assumptions made using the available data and the assumptions at times can be sub-optimal.

Online learning vs Batch learning		
Complexity	More complex because the model keeps evolving over time as more data becomes available.	Less complex because the model is fed with more consistent data sets periodically.
Computational power	More computational power is required because of the continuous feed of data that leads to continuous refinement.	Less computational power is needed because data is delivered in batches; the model isn't continuously refining itself.
Use in production	Harder to implement and control because the production model changes in real-time according to its data feed.	Easier to implement because offline learning provides engineers with more time to perfect the model before deployment.
Applications	Used in applications where new data patterns are constantly required (e.g., weather prediction tools, Stock market predictions)	Used in applications where data patterns remain constant and don't have sudden concept drifts (e.g., image classification)

Table2: Comparison of Online learning vs Batch learning