UNIT-2

1. **AI in Health Care Industry**Machine learning is one of the most common forms of artificial intelligence in healthcare. It is a broad technique at the core of many approaches to AI and healthcare technology and there are many versions of it.

Using artificial intelligence in healthcare, the most widespread utilization of traditional machine learning is precision medicine. Being able to predict what treatment procedures are likely to be successful with patients based on their make-up and the treatment framework is a huge leap forward for many healthcare organizations. The majority of AI technology in healthcare that uses machine learning and precision medicine applications require data for training, for which the end result is known. This is known as supervised learning.

Artificial intelligence in healthcare that uses deep learning is also used for speech recognition in the form of natural language processing (NLP). Features in deep learning models typically have little meaning to human observers and therefore the model's results may be challenging to delineate without proper interpretation.

Natural Language Processing

Making sense of human language has been a goal of artificial intelligence and healthcare technology for over 50 years. Most NLP systems include forms of speech recognition or text analysis and then translation. A common use of artificial intelligence in healthcare involves NLP applications that can understand and classify clinical documentation. NLP systems can analyze unstructured clinical notes on patients, giving incredible insight into understanding quality, improving methods, and better results for patients.

Rule-based Expert Systems

Expert systems based on variations of ‘if-then’ rules were the prevalent technology for AI in healthcare in the 80s and later periods. The use of artificial intelligence in healthcare is

widely used for clinical decision support to this day. Many electronic health record systems (EHRs) currently make available a set of rules with their software offerings.

Expert systems usually entail human experts and engineers to build an extensive series of rules in a certain knowledge area. They function well up to a point and are easy to follow and process. But as the number of rules grows too large, usually exceeding several thousand, the rules can begin to conflict with each other and fall apart. Also, if the knowledge area changes in a significant way, changing the rules can be burdensome and laborious. Machine learning in healthcare is slowly replacing rule-based systems with approaches based on interpreting data using proprietary medical algorithms.

Diagnosis and Treatment Applications

Diagnosis and treatment of disease has been at the core of artificial intelligence AI in healthcare for the last 50 years. Early rule-based systems had potential to accurately diagnose and treat disease, but were not totally accepted for clinical practice. They were not significantly better at diagnosing than humans, and the integration was less than ideal with clinician workflows and health record systems.

But whether rules-based or algorithmic, using artificial intelligence in healthcare for diagnosis and treatment plans can often be difficult to marry with clinical workflows and EHR systems. Integration issues have been a greater barrier to widespread adoption of AI in healthcare when compared to the accuracy of suggestions. Much of the AI and healthcare capabilities for diagnosis and treatment from medical software vendors are standalone and address only a certain area of care. Some EHR software vendors are beginning to build limited healthcare analytics functions with AI into their product offerings, but are in the elementary stages. To take full advantage of the use of artificial intelligence in healthcare using a stand alone EHR system providers will either have to undertake substantial integration projects themselves, or leverage the capabilities of third party vendors that have AI capabilities and can integrate with their EHR.

Administrative Applications

There are a number of administrative applications for artificial intelligence in healthcare. The use of artificial intelligence in hospital settings is somewhat less game changing in this area as compared to patient care. But artificial intelligence in hospital administrative areas can provide substantial efficiencies. AI in healthcare can be used for a variety of applications, including claims processing, clinical documentation, revenue cycle management and medical records management.

Another use of artificial intelligence in healthcare applicable to claims and payment administration is machine learning, which can be used for pairing data across different databases. Insurers and providers must verify whether the millions of claims submitted daily are correct. Identifying and correcting coding issues and incorrect claims saves all parties time, money and resources.

Challenges for Artificial Intelligence in Healthcare

One challenge of using artificial intelligence in healthcare is that it requires access to large amounts of data in order to be effective. Another challenge is that there is a risk of bias if the data used to train the algorithms is not representative of the population as a whole. Finally, there is a lack of standardization across different artificial intelligence systems, which can make it difficult to compare results or combine data from multiple sources.

2. AI in Customer support

AI-enabled CRM software will be able to:

- Analyze customer records & business data the system collects from sales, e-commerce activity, emails, IoT generated data & social media, etc., and provide automated insights. For example, you will be able to:
    - Identify target businesses as well as provide companies' previous purchase patterns, informing you whether your solution is complementary or redundant. The predictive analytics will provide better sentiment/intent analysis, product recommendations, and upselling. This intelligence will also increase customer retention.
    - Assist sales reps to focus on the most promising leads based on engagement data analysis & advanced lead scoring tools. Also, recommend when to trigger email campaigns using machine learning algorithms and customer response history.
- Help determine the best person to contact to win the opportunity. Over time, the system will be able to teach itself to do an even better job of targeting potential customers and decision-makers by identifying which personal or professional attributes hold the most weight. As the software evolves, it will effectively create a system of continuous improvement for sales and marketing teams.
- By enhancing and automating the routine-based parts of the business process, sales teams are allowed to focus on their efforts on better serving the more complicated and human-demanding needs of customers.
- Use Chatbots/assistants that:
    - Can automatically reach out to anyone who has shown interest in the company, such as by downloading a whitepaper or requesting information from the website. The assistant processes replies from customers determine feedback and potential questions and issues a response. The assistant passes off the lead to a human salesperson when the time is right.
    - In conjunction with AI algorithms, companies can efficiently identify and resolve customer issues/problems through natural conversation. These assistants free up customer support rep time for more critical and complicated tasks.

3. AI IN THE ENERGY SECTOR

While incorporating AI in the energy sector isn't seamless, the prospective benefits outweigh the implementation costs. Some of the possible applications of Artificial Intelligence in energy include but are not limited to smart grids, data digitalization, forecasting, and more advanced resource management. Let's take a look at some significant benefits of AI in energy.

1. **Data digitalization**. As the energy sector has been rapidly digitalizing in recent years, AI has played a vital role in this process. Deloitte specifies that 66% of oil and gas companies highlight that the benefits of digitalization outweigh any cybersecurity risks [8]. AI can help transform energy companies by automating grid data collection and implementing analysis frameworks. With the vast amount of data existing in the energy sector, converting it into reusable information for AI and Machine Learning algorithms is a go-to option.
2. **Smart forecasting**. Even when discussing renewables, forecasting is widely used to determine the energy output in particular geographical areas accurately. Deep learning AI algorithms have a larger predictive capacity than all industry specialists combined.

Forecasting, in this sense, can take various forms, ranging from predicting demand and price trends to identifying potential growth areas.

3. **Resource management**. Artificial Intelligence in energy and utilities heavily relies upon controlling, sustaining, and supplying uninterrupted power output. With AI-powered resource management, suppliers can balance traditional and renewable energy proportions. Proper resource management can also fine-tune the grid for optimal use or request maintenance in critical situations.
4. **Failure prevention**. Over the last years, dozens of ill-reputed energy-related cases have become more public, including oil spills or hazardous coal extraction facilities. In this context, AI-powered failure prediction is a top priority in the industry. By monitoring data for patterns and trends, AI can identify potential problems before they happen. It ultimately allows taking corrective action to avoid disruptions. Modern AI solutions in the energy sector utilize SCADA, maintenance, and budget data to prevent shortages or grid failures.
5. **Predictive analytics for renewables**. Predictive analytics for renewables includes identifying areas with the highest potential for Artificial Intelligence in renewable energy development, such as wind and solar panels. With well-rounded analytics on the subject matter, suppliers can utilize Artificial Intelligence in energy output efficiently.

4. COMMON ENERGY-RELATED AI USE CASES

Currently, the most ambitious projects are concerned with a smart grid, energy-efficiency programs, digital twins, and renewable energy integration.

- Smart grid

A smart grid is a new approach to energy efficiency networks, capitalizing on the two-way flow of electricity and data. The main difference from the usual networks is the implementation of AI, Cloud, and digital technologies that support control and self-regulation. One of the prominent examples of the smart grid is the cooperation between London's National Grid and IBM's cloud-based analytics. The smart grid provides preventive and predictive maintenance, which are crucial parameters of the grid's functionality. Overall, the AI-powered smart grid helps provide more precise forecasting, displaying higher resilience and improved security for the grid.

- Energy-efficiency programs

Energy efficiency, one of the Sustainable Development Goals, should be treated seriously. AI-powered energy efficiency programs oversee energy usage, provide a framework for smart forecasting, and regulate usage during peak hours. When using model-based predictive control, it's possible to yield an energy-efficiency improvement of 10.2% to 40% [9]. Predictive analytics and Machine Learning can present up-to-the-point predictions. Consequently, these estimates are used for designing and implementing energy efficiency plans at the company, municipality, or state levels.

- Smart heaters

Smart heaters can be part of modern renewable strategies thanks to their control of the entire heat system. The fundamental approach here is to allocate the power reasonably, allowing it to direct the unused energy to particular areas. The N-iX team created the end-to-end signup flow for smart heaters. We received a legacy flow that lacked payment functionality and was unfinished.

The task was to redesign, develop the front end, refactor the back end, and build a production environment. A new flow created by our team allowed the user to choose the smart heater and create a request to analyze the house by an expert. To develop the solution, we used Scala, React.js, and AWS. The N-iX team released the MVP to production and developed additional features.

- Digital twins

Digital twins were seen as the life-saving framework for the industrial energy complex. A digital twin refers to the multi-dimensional visual representation of a process, facility, or physical object. These digital twins act like real-time virtual models that present more research possibilities than simulations. In the energy and Artificial Intelligence sector, digital twins help study wind turbines and power-generation facilities. Using AI, a digital twin can be a step forward in better servicing, experimenting, maintaining, and optimizing the energy network, either traditional or renewable.

- Renewable energy integration

Finding a balance between traditional and renewable energy sources is crucial for major energy providers. Thanks to Artificial Intelligence in energy and Machine Learning, it's now possible to forecast and predict the best circumstances for accurate integration of renewables. In other words, it helps manage integrating renewable energy sources into the traditional power grid. It can involve forecasting wind and solar farms and dispatching the energy outputs to balance the existing system.

6. AI in Oil and Gas

With the dynamic landscape for energy production, AI provides powerful benefits across the entire value chain. AI helps oil and gas companies assess the value of specific reservoirs, customize drilling and completion plans according to the geology of the area, and assess risks of each individual well. In addition, downstream operations can be optimized to minimize costs and maximize spreads.

AI Applications in the Oil and Gas Industry
AI is proving to be a real enabler for oil & gas projects offering multiple applications. These include production optimization with computer vision to analyze seismic and subsurface data faster, reducing downtime for predictive maintenance of equipment, reservoir understanding,

and modeling for predicting oil corrosion risks to reduce maintenance costs. Few more notable use cases and application areas are enlisted here:

1. Surface Analysis/Geological Assessment
2. Reducing Well/Equipment Downtime
3. Optimizing Production and Scheduling
4. Asset Tracking and Maintenance Using Digital Twins
5. Defect Detection
6. AI-led Cybersecurity
7. Workplace Safety
8. Analytics-Driven Decision Making
9. Emission Tracking
10. Logistics Network Optimizations and Logistics
11. AI Led Inventory Management
12. Backoffice Process Optimization
13. Optimized Procurement

7. Understanding the Machine Learning Workflow

Machine learning workflows define the steps initiated during a particular machine learning implementation. Machine learning workflows vary by project, but four basic phases are typically included.

Gathering machine learning data

Gathering data is one of the most important stages of machine learning workflows. During data collection, you are defining the potential usefulness and accuracy of your project with the quality of the data you collect.

To collect data, you need to identify your sources and aggregate data from those sources into a single dataset. This could mean streaming data from Internet of Things sensors, downloading open source data sets, or constructing a data lake from assorted files, logs, or media.

Data pre-processing

Once your data is collected, you need to pre-process it. Pre-processing involves cleaning, verifying, and formatting data into a usable dataset. If you are collecting data from a single source, this may be a relatively straightforward process. However, if you are aggregating several sources you need to make sure that data formats match, that data is equally reliable, and remove any potential duplicates.

Building datasets

This phase involves breaking processed data into three datasets—training, validating, and testing:

- **Training set**—used to initially train the algorithm and teach it how to process information. This set defines model classifications through parameters.
- **Validation set**—used to estimate the accuracy of the model. This dataset is used to finetune model parameters.

- **Test set**—used to assess the accuracy and performance of the models. This set is meant to expose any issues or mistrainings in the model.

Training and refinement

Once you have datasets, you are ready to train your model. This involves feeding your training set to your algorithm so that it can learn appropriate parameters and features used in classification.

Once training is complete, you can then refine the model using your validation dataset. This may involve modifying or discarding variables and includes a process of tweaking model-specific settings (hyperparameters) until an acceptable accuracy level is reached.

Machine learning evaluation

Finally, after an acceptable set of hyperparameters is found and your model accuracy is optimized you can test your model. Testing uses your test dataset and is meant to verify that your models are using accurate features. Based on the feedback you receive you may return to training the model to improve accuracy, adjust output settings, or deploy the model as needed.

Data Engineering

The initial step in any data science workflow is to acquire and prepare the data to be analyzed. Typically, data is being integrated from various resources and has different formats. The data preparation follows the data acquisition step, which is according to Gartner *"an iterative and agile process for exploring, combining, cleaning and transforming raw data into curated datasets for data integration, data science, data discovery and analytics/business intelligence (BI) use cases"*. Notably, even though the preparation phase is an intermediate phase aimed to prepare data for analysis, this phase is reported to be the most expensive with respect to resources and time. Data preparation is a critical activity in the data science workflow because it is important to avoid the propagation of data errors to the next phase, data analysis, as this would result in the derivation of wrong insights from the data.

The Data Engineering pipeline includes a sequence of operations on the available data that leads to supplying training and testing datasets for the machine learning algorithms:

1. *Data Ingestion* - Collecting data by using various frameworks and formats, such as Spark, HDFS, CSV, etc. This step might also include synthetic data generation or data enrichment.
2. *Exploration and Validation* - Includes data profiling to obtain information about the content and structure of the data. The output of this step is a set of metadata, such as max, min, avg of values. Data validation operations are user-defined error detection functions, which scan the dataset in order to spot some errors.
3. *Data Wrangling (Cleaning)* - The process of re-formatting particular attributes and correcting errors in data, such as missing values imputation.
4. *Data Labeling* - The operation of the Data Engineering pipeline, where each data point is assigned to a specific category.
5. *Data Splitting* - Splitting the data into training, validation, and test datasets to be used during the core machine learning stages to produce the ML model.

Model Engineering

The core of the ML workflow is the phase of writing and executing machine learning algorithms to obtain an ML model. The Model Engineering pipeline includes a number of operations that lead to a final model:

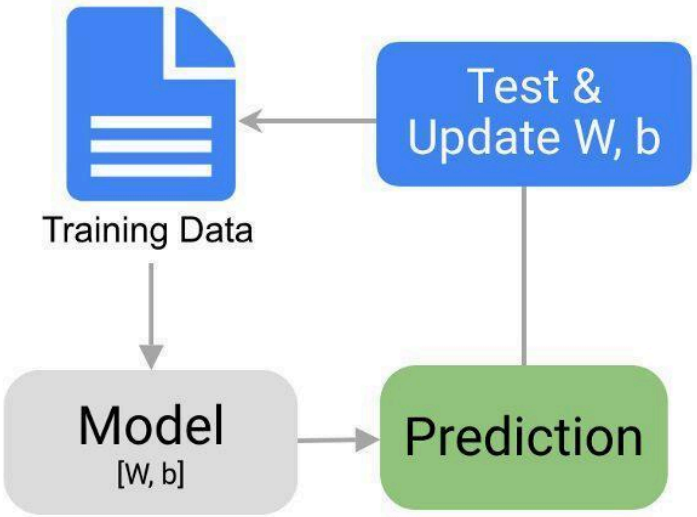
1. *Model Training* - The process of applying the machine learning algorithm on training data to train an ML model. It also includes feature engineering and the hyperparameter tuning for the model training activity.
2. *Model Evaluation* - Validating the trained model to ensure it meets original codified objectives before serving the ML model in production to the end-user.
3. *Model Testing* - Performing the final "Model Acceptance Test" by using the hold backtest dataset.
4. *Model Packaging* - The process of exporting the final ML model into a specific format (e.g. PMML, PFA, or ONNX), which describes the model, in order to be consumed by the business application.

Model Deployment

Once we trained a machine learning model, we need to deploy it as part of a business application such as a mobile or desktop application. The ML models require various data points (feature vector) to produce predictions. The final stage of the ML workflow is the integration of the previously engineered ML model into existing software. This stage includes the following operations:

1. *Model Serving* - The process of addressing the ML model artifact in a production environment.
2. *Model Performance Monitoring* - The process of observing the ML model performance based on live and previously unseen data, such as prediction or recommendation. In particular, we are interested in ML-specific signals, such as prediction deviation from previous model performance. These signals might be used as triggers for model re-training.
3. *Model Performance Logging* - Every inference request results in the log-record.

8. The 7 Steps of Machine Learning



1. Data Collection
   → The quantity & quality of your data dictate how accurate our model is
   → The outcome of this step is generally a representation of data (Guo simplifies to specifying a table) which we will use for training
   → Using pre-collected data, by way of datasets from Kaggle, UCI, etc., still fits into this step
2. Data Preparation
   → Wrangle data and prepare it for training
   → Clean that which may require it (remove duplicates, correct errors, deal with missing values, normalization, data type conversions, etc.)
   → Randomize data, which erases the effects of the particular order in which we collected and/or otherwise prepared our data
   → Visualize data to help detect relevant relationships between variables or class imbalances (bias alert!), or perform other exploratory analysis
   → Split into training and evaluation sets
3. Choose a Model
   → Different algorithms are for different tasks; choose the right one
4. Train the Model
   → The goal of training is to answer a question or make a prediction correctly as often as possible
   → Linear regression example: algorithm would need to learn values for m (or W) and b (x is input, y is output)
   → Each iteration of process is a training step
5. Evaluate the Model
   → Uses some metric or combination of metrics to "measure" objective performance of model
   → Test the model against previously unseen data
   → This unseen data is meant to be somewhat representative of model performance in the real world, but still helps tune the model (as opposed to test data, which does not)
   → Good train/eval split? 80/20, 70/30, or similar, depending on domain, data availability, dataset particulars, etc.
6. Parameter Tuning
   → This step refers to *hyperparameter* tuning, which is an "artform" as opposed to a science
   → Tune model parameters for improved performance
   → Simple model hyperparameters may include: number of training steps, learning rate, initialization values and distribution, etc.
7. MakePredictions
   → Using further (test set) data which have, until this point, been withheld from the model (and for which class labels are known), are used to test the model; a better approximation of how the model will perform in the real world

A Simplified Machine Learning Process

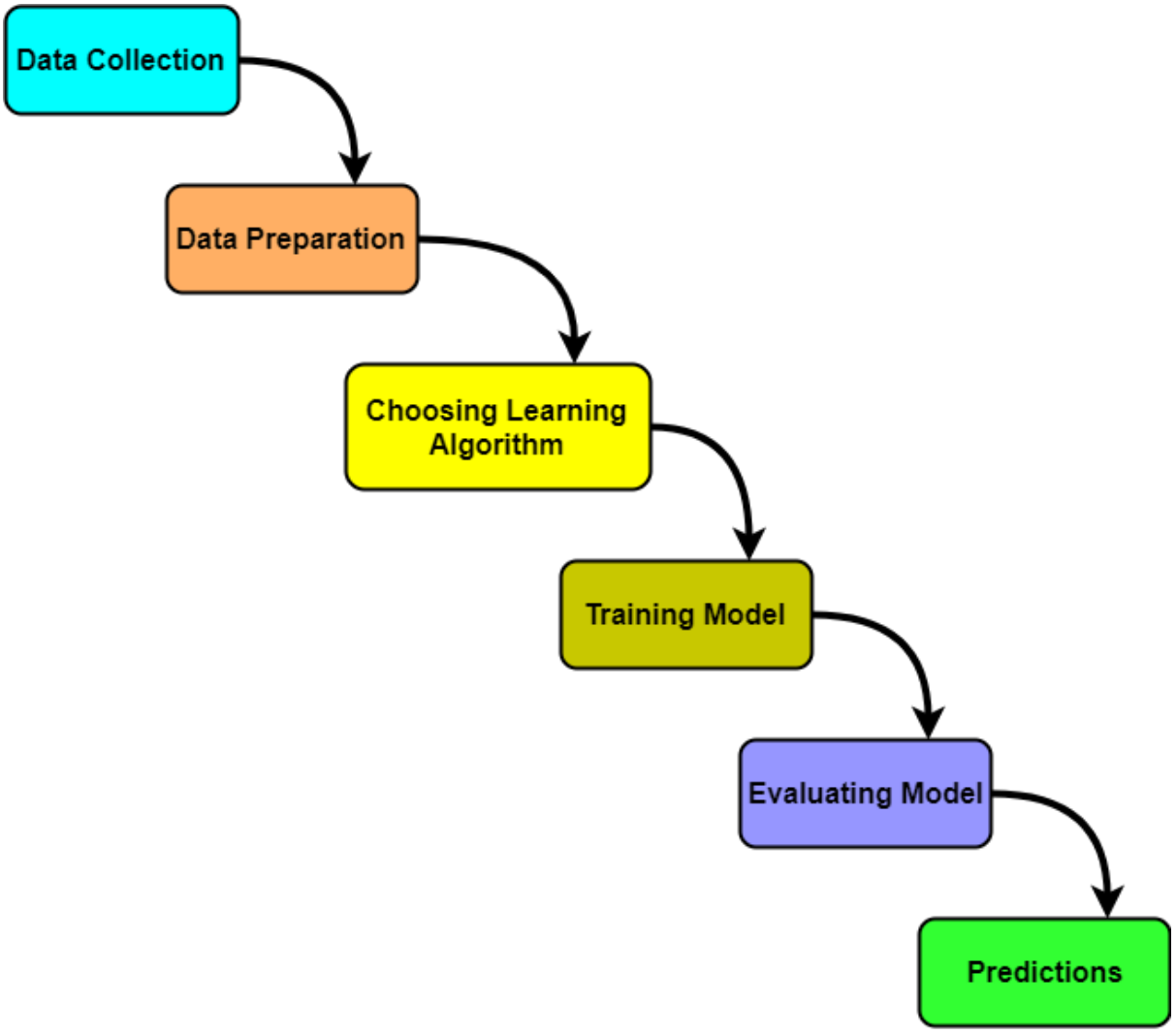
Let's use the above to put together a simplified framework to machine learning, the **5 main areas of the machine learning process**:

1. Data collection and preparation
   → everything from choosing where to get the data, up to the point it is clean and ready for feature selection/engineering
2. Feature selection and feature engineering
   → this includes all changes to the data from once it has been cleaned up to when it is ingested into the machine learning model
3. Choosing the machine learning algorithm and training our first model
   → getting a "better than baseline" result upon which we can (hopefully) improve
4. Evaluating our model
   → this includes the selection of the measure as well as the actual evaluation; seemingly a smaller step than others, but important to our end result
5. Model tweaking, regularization, and hyperparameter tuning
   → this is where we iteratively go from a "good enough" model to our best effort

Machine Learning Workflow-

Machine learning workflow refers to the series of stages or steps involved in the process of building a successful machine learning system.

The various stages involved in the machine learning workflow are-



Machine Learning Workflow

1. Data Collection
2. Data Preparation
3. Choosing Learning Algorithm
4. Training Model
5. Evaluating Model
6. Predictions

Let us discuss each stage one by one.

1. Data Collection-

In this stage,

- Data is collected from different sources.
- The type of data collected depends upon the type of desired project.
- Data may be collected from various sources such as files, databases etc.
- The quality and quantity of gathered data directly affects the accuracy of the desired system.

2. Data Preparation-

In this stage,

- Data preparation is done to clean the raw data.
- Data collected from the real world is transformed to a clean dataset.
- Raw data may contain missing values, inconsistent values, duplicate instances etc.
- So, raw data cannot be directly used for building a model.

Different methods of cleaning the dataset are-

- Ignoring the missing values
- Removing instances having missing values from the dataset.
- Estimating the missing values of instances using mean, median or mode.
- Removing duplicate instances from the dataset.
- Normalizing the data in the dataset.

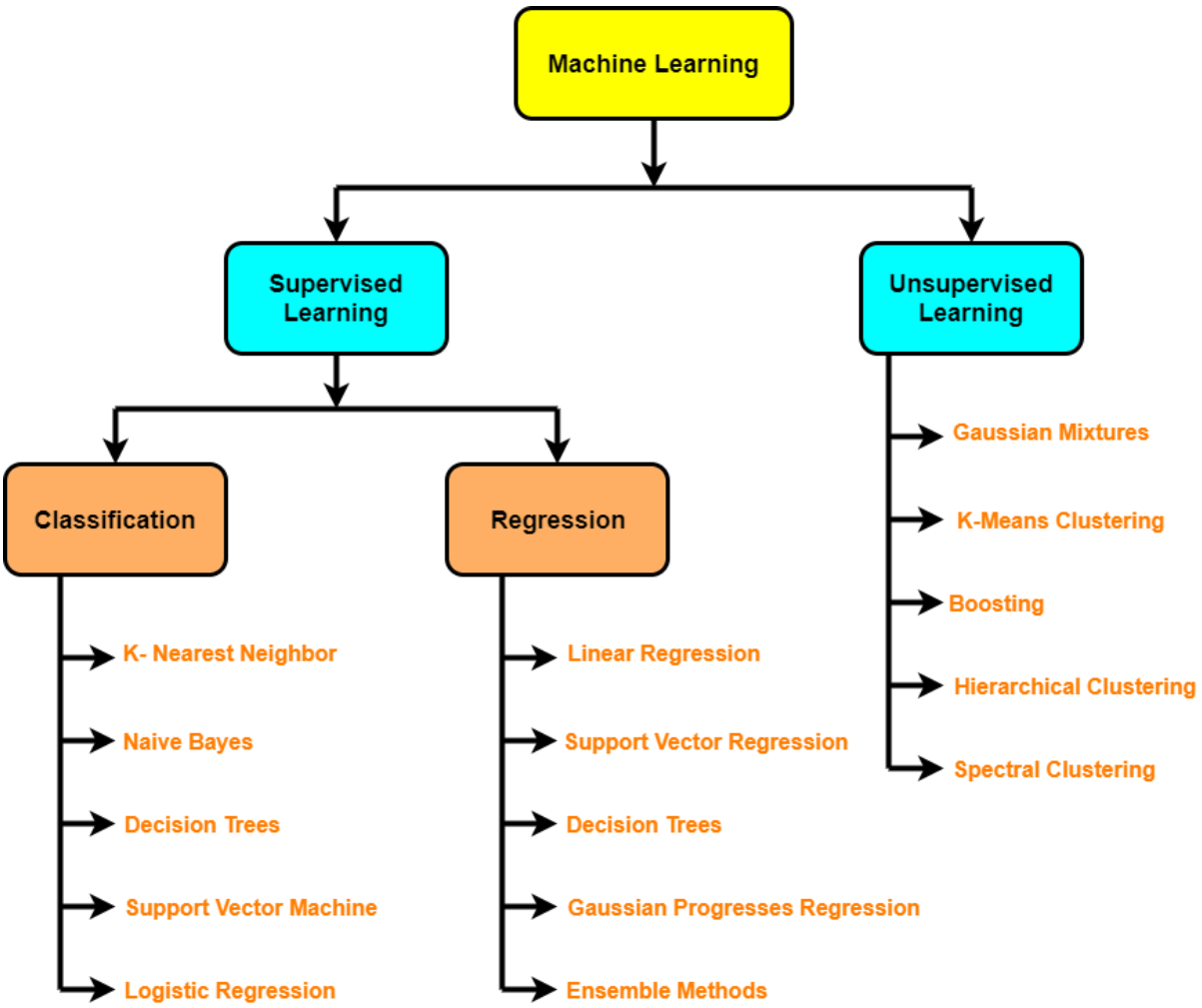
This is the most time consuming stage in machine learning workflow.

3. Choosing Learning Algorithm-

In this stage,

- The best performing learning algorithm is researched.
- It depends upon the type of problem that needs to solved and the type of data we have.
- If the problem is to classify and the data is labeled, classification algorithms are used.
- If the problem is to perform a regression task and the data is labeled, regression algorithms are used.
- If the problem is to create clusters and the data is unlabeled, clustering algorithms are used.

The following chart provides the overview of learning algorithms-



4. Training Model-

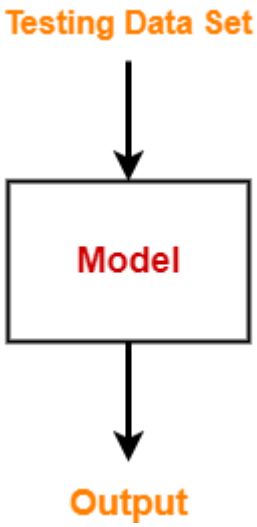
In this stage,

- The model is trained to improve its ability.
- The dataset is divided into training dataset and testing dataset.
- The training and testing split is order of 80/20 or 70/30.
- It also depends upon the size of the dataset.
- Training dataset is used for training purpose.
- Testing dataset is used for the testing purpose.
- Training dataset is fed to the learning algorithm.
- The learning algorithm finds a mapping between the input and the output and generates the model.

5. Evaluating Model-

In this stage,

- The model is evaluated to test if the model is any good.
- The model is evaluated using the kept-aside testing dataset.
- It allows to test the model against data that has never been used before for training.
- Metrics such as accura uh-cy, precision, recall etc are used to test the performance.
- If the model does not perform well, the model is re-built using different hyper parameters.
- The accuracy may be further improved by tuning the hyper parameters.



6. Predictions-

In this stage,

- The built system is finally used to do something useful in the real world.
- Here, the true value of machine learning is realized.