A Project Report

On

**CRIME TYPE AND OCCURRENCE PREDICTION USING MACHINE LEARNING ALGORITHM**

### Submitted to

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY**

### Kukatpally, Hyderabad-500085, Telangana, India

In partial fulfilment of the requirement for the award of degree of

## BACHELOR OF TECHNOLOGY

In

## INFORMATION TECHNOLOGY

By

## TANGELLAMUDI SAI TEJA [19E31A1226]

Under the guidance of

## Mr. B. MALLIAH

**Assistant Professor, MIST**



## Department of Information Technology

## MAHAVEER INSTITUTE OF SCIENCE AND TECHNOLOGY

**(**Affiliated to JNTU Hyderabad ,Approved by AICTE)

Vyasapuri, bandlaguda post:keshavgiri, Hyderabad-500005

2020-2021

## MAHAVEER INSTITUTE OF SCIENCE AND TECHNOLOGY

**(**Affiliated to JNTU Hyderabad ,Approved by AICTE)

Vyasapuri,bandlaguda post:keshavgiri,Hyderabad-500005

## Department of Information Technology



**CERTIFICATE**

## This is to certify that this Project work Report titled “CRIME TYPE AND OCCURRENCE PREDICTION USING MACHINE LEARNING ALGORITHM” which is being submitted by TANGELLAMUDI SAI TEJA [19E31A1226] in partial fulfilment for the award of the Degree of Bachelor of Technology in Information Technology, affiliated of Jawaharlal Nehru Technological University, Hyderabad and record of the bonafide work carried out by them under our guidance during 2022- 2023.

Signature of Project Guide Signature of Head of the department

**Mr. B. MALLIAH Dr.A.NANDA GOPAL REDDY**

**(Assistant Professor) (Head of the Department)**

**External Examiner DR. B.V SANKER RAM**

**(**principal**)**

**ACKNOWLEDGMENT**

We would like to express our deep-felt appreciation and gratitude to **MR. B. MALLIAH** our project guide, for her skilful guidance, constant supervision, timely suggestion, keen interest and encouragement in completing the project within stipulated time.

We wish to express my special thanks to **MRS .K.PRIYANKA ,** Project co-ordinator who has shown keen interest and even rendered his valuable guidance in terms of suggestions and encouragement extended to us with an immense care and zeal.

We express our profound sense of gratitude to **DR. A.NANDA GOPAL REDDY**, HOD of Dept of IT, who has served as a host of valuable corrections and for providing us time and amenities to complete this project.

We extend my thanks to  **DR. B. V. SANKER RAM,** Principal of our college and the management of **MAHAVEER INSTITUTE OF SCIENCE & TECHNOLOGY** for providing excellent academic environment in the college.

We wish to express our gratitude to the Members of staff and all others who helped us in more than one way. We would also like to thank Lab Assistants and Programmers for helping us through our project.

## TANGELLAMUDI SAI TEJA

## [19E31A1226]

# **DECLARATION**

We hereby declare that the project entitled “**CRIME TYPE AND OCCURRENCE PREDICTION USING MACHINE LEARNING ALGORITHM ”** submitted to partial fulfilment of the requirements for award of degree to  **BACHELOR OF TECHNOLOGY** **at MAHAVEER INSTITUTE OF SCIENCE AND TECHNOLOGY**, affiliated to **JAWAHARLAL NEHRU TECHNOLOGY UNIVERSITY, HYDERABAD**  in academic work and has not been submitted to any other university institute for award of any degree.

## TANGELLAMUDI SAI TEJA

## [19E31A1226]

**ABSTRACT**

In this era of recent times, crime has become an evident way of making people and society under trouble. An increasing crime factor leads to an imbalance in the constituency of a country. In order to analyse and have a response ahead this type of criminal activities, it is necessary to understand the crime patterns. This study imposes one such crime pattern analysis by using crime data obtained from Kaggle open source which in turn used for the prediction of most recently occurring crimes. The major aspect of this project is to estimate which type of crime contributes the most along with time period and location where it has happened. Some machine learning algorithms such as Naïve Bayes is implied in this work in order to classify among various crime patterns and the accuracy achieved was comparatively high when compared to precomposed works.

Keywords: Crime, Analyse, Crime patterns, Kaggle, Estimate, Naïve Bayes, Accuracy

# **TABLE OF CONTENTS**

**TITLE PAGE NO**

**CHAPTER-1 01**

**INTRODUCTION**

**CHAPTER-2 02**

**LITERATURE SURVEY**

**CHAPTER-3**

**SYSTEM ANALYSIS**

* 1. EXISTING SYSTEM
  2. PROPOSED SYSTEM
  3. SYSTEM ARCHITECTURE
  4. HARDWARE AND SOFTWARE SPECIFICATION
     1. HARDWARE
     2. SOFTWARE

**CHAPTER-4**

**SYSTEM DESIGN**

* 1. **DATA FLOW DIAGRAM**

4.1.0 LEVEL 0

4.1.1 LEVEL 1

4.1.2 LEVEL 2

* 1. **UML DIAGRAMS**

4.2.1 USECASE DIAGRAM

4.2.2 CLASS DIAGRAM

4.2.3 OBJECT DIAGRAM

4.2.4 SEQUENCE DIAGRAM

4.2.5 ACTIVITY DIAGRAM

4.2.6 COLLABRATION DIAGRAM

4.2.7 STATECHART DIAGRAM

4.2.8 COMPONENT DIAGRAM

4.2.9 DEPLOYMENT DIAGRAM

CHAPTER-5

SOFTWARE ENVIRONMENTS

* 1. [MACHINE LEARNING](#_TOC_250004)
  2. WORKING
  3. MEACHINE LEARNING ALGORITHM
     1. [SUPERVISED LEARNING](#_TOC_250003)
     2. [UNSUPERVISED LEARNING](#_TOC_250002)
  4. APPLICAIONS OF MACHINE LEARNING
  5. [DEEP LEARNING](#_TOC_250001) 
     1. [REINFORCEMENT LEARNING](#_TOC_250000)
     2. APPLICATIONS OF DEEP LEARNING
  6. TENSORFLOW ARCHITECTURE
  7. REQUIREMENT ANALYSIS
     1. PYTHON OVERVIEW
     2. ENVIRONMENT
     3. ANACONDA NAVIGATOR

**CHAPTER-6**

**ALGORITHM**

* 1. K-MEANS CLUSTERING ALGORITHM
  2. RANDOM FOREST ALGORITHM
  3. LOGISTIC REGRESSION
  4. SUPPORT VECTOR MACHINE ALGORITHM

**CHAPTER-7**

**IMPLEMENTATION AND CODING**

* 1. MODULE
  2. SAMPLE CODE

**CHAPTER-8**

**OUTPUT SCREENS**

**CHAPTER-9**

**TESTING**

**CHAPTER-10**

**CONCLUSION**

## CHAPTER-11

**FUTURE SCOPE**

**BIBLIOGRAPHY**

**CHAPTER 1**

**INTRODUCTION**

Crime has become a major thread imposed which is considered to grow relatively high in intensity. An action stated is said to be a crime, when it violates the rule, against the government laws and it is highly offensive. The crime pattern analysis requires a study in the different aspects of criminology and also in indicating patterns. The Government has to spend a lot of time and work to imply technology to govern some of these criminal activities. Hence, use of machine learning techniques and its records is required to predict the crime type and patterns. It imposes the uses of existing crime data and predicts the crime type and its occurrence bases on the location and time. Researchers undergone many studies that helps in analysing the crime patterns along with their relations in a specific location. Some of the hotspots analysed has become easier way of classifying the crime patterns. This leads to assist the officials to resolve them faster. This approach uses a dataset obtained from Kaggle open source based on various factors along with the time and space where it occurs over a certain period of time. We implied a classification algorithm that helps in locating the type of crime and hotspots of the criminal actions that takes place on the certain time and day. In this proposed one to impose a machine learning algorithm to find the matching criminal patterns along with the assist of its category with the given temporal and spatial data.

**CHAPTER 2**

**LITERATURE SURVEY**

Crime are of different type that occurs at different locations around the various geographical location. Many research scholars have been suggesting a mechanism to analyse the relationship between crime and social variables that includes unemployed individuals, earning amount, level of education and so on.

1.Suhong Kim and Param Joshi proposed two different machine learning models which is used for prediction, K nearest neighbour algorithm (KNN) and decision tree approach. The accuracy obtained ranges between 39 to 44 percent when predicting crime patterns and finding the crime type.

2. Benjamin Fredrick David he imposed a data mining technique that involves evaluating and inspect large pre-existing datasets in accordance to deliver more information. The extraction of new patterns is cross checked with predefined datasets available.

3. Shraddha S. Kavathekar used association rule mining in predicting crimes. Some Machine learning algorithms including Deep Neural Network (DNN) and Artificial Neural Network (ANN) have been implied. A deep neural network works more accurately using the feature level dataset. Using DNN, entirely connected convolution layers has been used in building the prediction model, mainly for multilabelled data classification. It was implemented using Tenser flow that is an API mainly designed for Deep learning technique with the dropout layers. These findings suggest that when there is more count of missing values, there is a need for pre-processing because crimes do not occur in the same manner but focuses on some particular areas. Artificial Neural Network [ANN] is based on the prognosis by trend analysis in solving problems. It comprises of enormous amount of processing constituent that works altogether in building a model.

4.Chandy and Abraham proposed a random forest classifier in extracting the features for data processing using cloud computing. The extracted features are request number, user identification, expiry time, time of arrival nd memory requirement. After feature extraction, the prediction of work load is done by using the trained data that has been perceived from the learning stage that allows to learn the details of the extracted features from user’s request.

5. Rohit Patil, Muzamil Kacchi, Pranali Gavali and Komal Pimparia suggests an Apriori algorithm for frequent patterns and the result obtained from K-means is used. Due to increase in crime rate over these recent years, system has to handle an enormous amount of data which requires more time to analyse them manually. Hence, advance machine learning approaches like K means clustering has been used.

6. A literature survey on Spatial and Temporal Hotspot prediction of crime proposed a study to categorize and evaluate the location and time of the crime hotspot detection techniques by performing (SLR) Systematic Literature Review.

7. Fuzhan Nasiri, Zakikhani, Kimiya and Tarek Zayed [7] suggested a failure prediction model that helps in detecting the corrosion in the pipelines of gas transmission. Most of the prediction model depend absolutely on the experimental tests data or involving some of the limited historical data records. This helps in ignoring the corrosion from various geographical circumstances.

8.Nikhli Dubey and Setu K. Chaturvedi imposed pertinent analysis of data mining approaches for the detection of the impeding future crime.

9.A Computational mechanism to classify the crime using machine learning techniques proposed a malleable computational implementation tool to analyse the crime rate in a country helps in classifying cybercrimes.

10.Hyeon-Woo Kang and Hang-Bong Kang suggested a fusion method based on Deep Neural Network in predicting the criminal activities from the feature level data with sufficient parameters.

**CHAPTER 3**

**SYSTEM ANALYSIS**

**3.1 Existing system**

Based on the previous year crime details in Indian states, It present statistical models through Weighted Moving Average, Functional Coefficient Regression and Arithmetic-Geometric Progression based prediction of the crime in coming years. Difference between actual records and our predicted values for both years gives the accuracy of the proposed approaches between the range 85% and 90%. In future, this work can be modified by using Machine Learning (ML) models for forecasting crime, as the data points will sufficiently increase to apply ML models. This can also increase the accuracy of the predictions. Further, statistical modeling’s methods can also be clubbed with ML models and then calculate weighted accuracy for a district, this can make the solution more robust.

**Disadvantages:**

Although some approaches and some detection techniques are present like women safety security system in iot and embedded, there may be some accuracy problems.

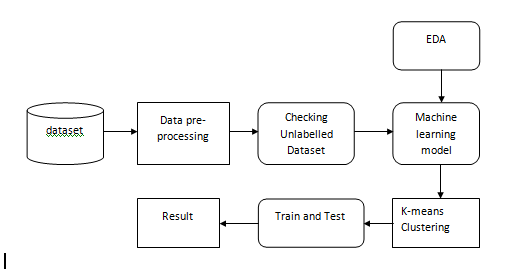
**3.2 Proposed System**

There are many machine learning algorithms available to users that can be implemented on datasets. However, there are two major types of learning algorithms: supervised learning and unsupervised learning algorithms. The clustering algorithms are given a particular attribute or set of attributes to predict. Data pre-processing process includes methods to remove any null values or infinite values which may affect the accuracy of the system. The main steps include Formatting, cleaning and sampling.

**Crimes Prediction ways:**

* To utilize the resources identify the hotspots of crimes and allocate vigilante resources such as policeman, police cars, weapons etc. reschedule patrols according to the vulnerability of a place.
* So, that avoids crimes Ensure better civilization through avoiding happening crimes such as murder, rapes, thefts, drug, smugglings etc.

**3.3 System Architecture**



**Advantages**

* Accuracy of prediction of crimes rates increases.
* Safety and security of people around us. Thus, machine learning algorithms also helps in safety and security system.

**3.4 System requirements**

Software and Hardware Requirements:

**3.4.1 Hardware:**

* OS – Windows 7, 8 and 10 (32 and 64 bit)
* RAM – 4GB

**3.4.2 Software:**

* In Python Language
* Anaconda Navigator
* Jupyter Notebook

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 DATA FLOW DIGARAM**

**4.1.0 Level 0**

|  |
| --- |
|  |
|  |  |

**4.1.1 Level 1**

|  |
| --- |
|  |
|  |  |

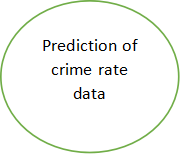
**4.1.2 Level 2**

|  |
| --- |
| ­­­ |
|  |  |



|  |
| --- |
|  |
|  |  |







## 4.2 UML DIAGRAMS

The Unified Modeling Language (UML) is used to specify, visualize, modify, construct and document the artifacts of an object-oriented software intensive system under development. UML offers a standard way to visualize a system's architectural blueprints, including elements such as:

* actors

* business processes

* (logical) components

* activities

* programming language statements

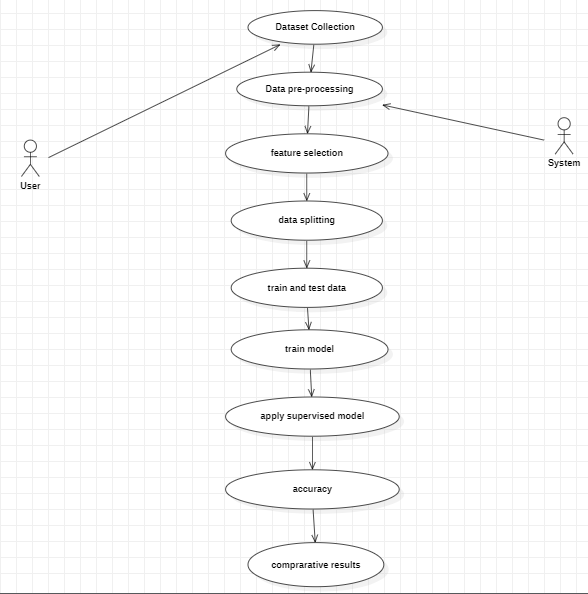
* database schemas, and

* Reusable software components.

UML combines best techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies. UML has synthesized the notations of the Booch method, the Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them into a single, common and widely usable modeling language. UML aims to be a standard modeling language which can model concurrent and distributed systems.

**4.2.1 USE CASE DIAGRAM:**

* UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.
* UML was created by Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.
* OMG is continuously putting effort to make a truly industry standard.
* UML stands for Unified Modeling Language.
* UML is a pictorial language used to make software blueprints

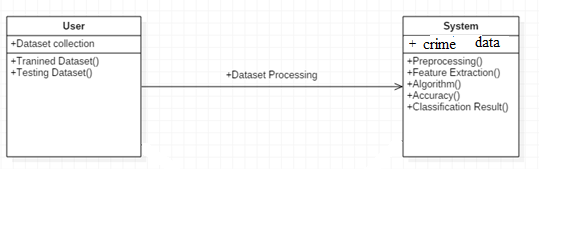


## 4.2.2 Class diagram

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the system of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling.[1] The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

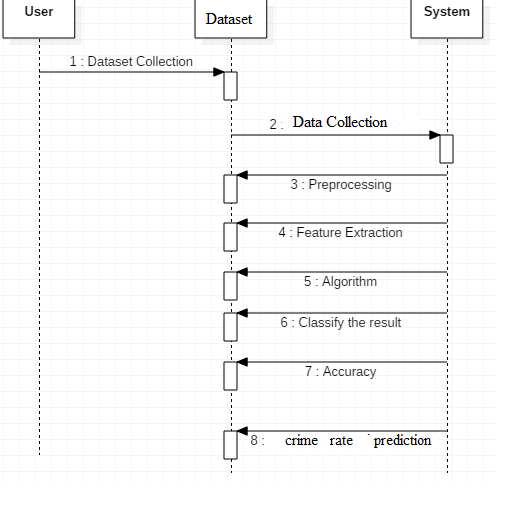
In the diagram, classes are represented with boxes that contain three compartments:

* The top compartment contains the name of the class. It is printed in bold and centered, and the first letter is capitalized.
* The middle compartment contains the attributes of the class. They are left-aligned and the first letter is lowercase.
* The bottom compartment contains the operations the class can execute. They are also left-aligned and the first letter is lowercase.



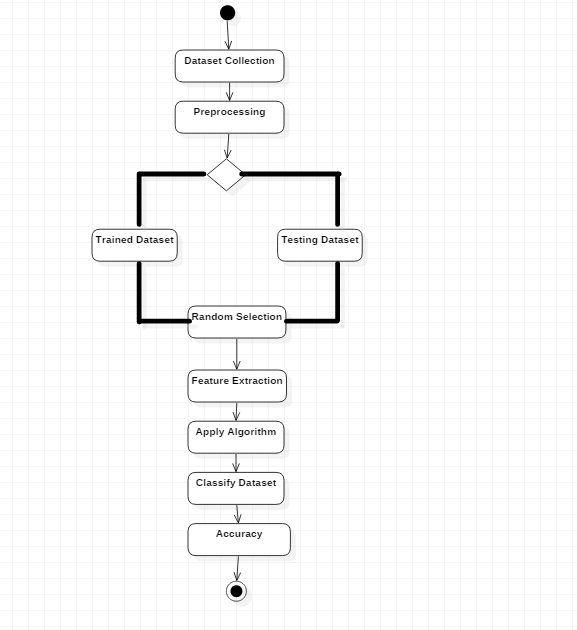
**4.2.3 Sequence Diagram:**

Sequence Diagrams Represent the objects participating the interaction horizontally and time vertically. A Use Case is a kind of behavioral classifier that represents a declaration of an offered behavior. Each use case specifies some behavior, possibly including variants that the subject can perform in collaboration with one or more actors. Use cases define the offered behavior of the subject without reference to its internal structure. These behaviors, involving interactions between the actor and the subject, may result in changes to the state of the subject and communications with its environment. A use case can include possible variations of its basic behavior, including exceptional behavior and error handling.



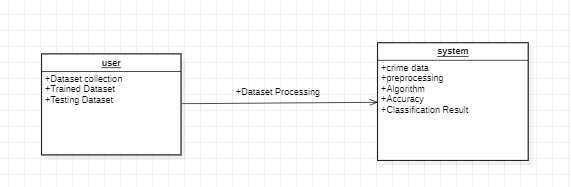
## 4.2.4 Activity Diagrams:

Activity diagrams are graphical representation of workflows of stepwise activities and actions with support for choice, iteration and con currency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflow of components in a system. An activity diagrams shows the overall flow of control.



**4.2.5 Object Diagram**

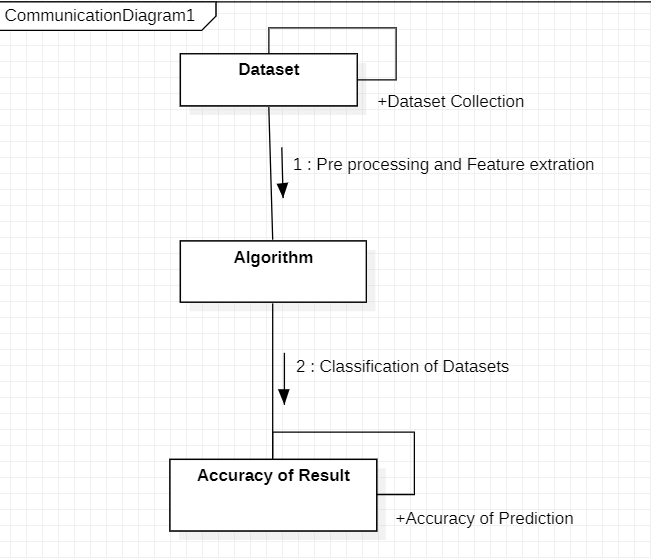
An object diagram is a UML structure diagram that shows the instance of the classifier in model. Object diagram use notation that is similar to that used in class diagram. Class diagram shows the actual classifier and their relationship in a system



**4.2.6 Collaboration Diagram**

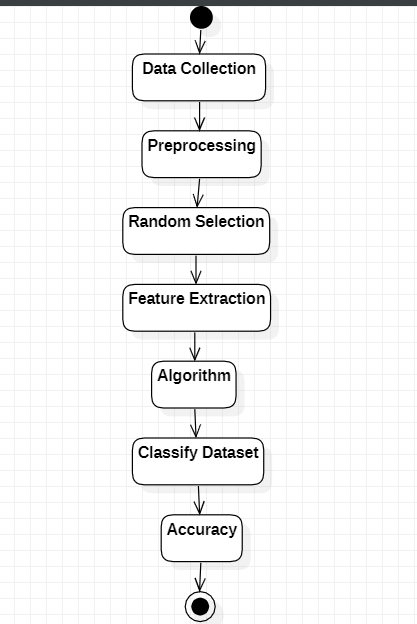
A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software object in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behaviour of a particular use case and define the role of each object.

In the above figure it depicts the illustration of the relationships and interaction among objects between the user and the application. The model will be generated after that URL will be uploaded and the URL will be monitored based legitimate or not and the output will be displayed.



**4.2.7 Stat Chart Diagram**

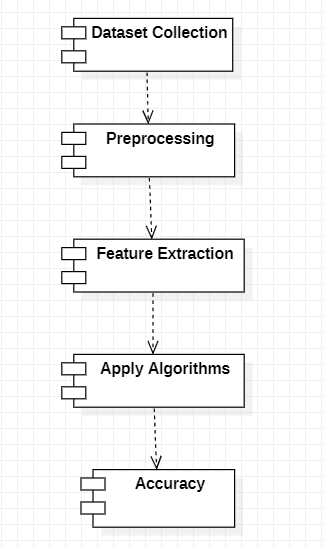
A State chart diagram describes a state machine. State machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events.



4.2.8 Component Diagram

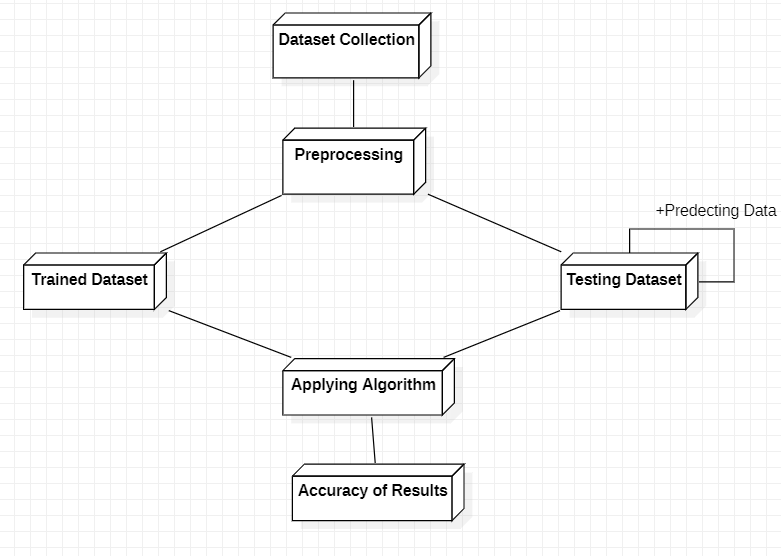
A component diagram, also known as a UML component diagram describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation detail and double-check that every aspect of the system's required functions is covered by planned development

In the above figure the project is divided into several components and these components describe organization and wiring of a system. First the user will load the model after the model is loaded, we can start behavior monitoring where the results or output will be displayed whether it is normal or abnormal based on that will be displayed.



**4.2.9 Deployment Diagram**

A deployment diagram is a UML diagram type that shows the execution architecture of a system, include nodes such as hardware or software execution environment, and the middleware connecting/algorithm. Deployment diagram are typically used to visualize the physical hardware and software of a system.



**CHAPTER 5**

**SOFTWARE ENVIRONMENTS**

**5.1 MACHINE LEARNING**

Machine Learning is a system that can learn from example through self-improvement and without being explicitly coded by programmer. The breakthrough comes with the idea that a machine can singularly learn from the data (i.e., example) to produce accurate results.

Machine learning combines data with statistical tools to predict an output. This output is then used by corporate to makes actionable insights. Machine learning is closely related to data mining and Bayesian predictive modeling. The machine receives data as input, use an algorithm to formulate answers.

A typical machine learning tasks are to provide a recommendation. For those who have a Netflix account, all recommendations of movies or series are based on the user's historical data. Tech companies are using unsupervised learning to improve the user experience with personalizing recommendation.

Machine learning is also used for a variety of task like fraud detection, predictive maintenance, portfolio optimization, automatize task and so on.

**Machine Learning vs. Traditional Programming**

Traditional programming differs significantly from machine learning. In traditional programming, a programmer code all the rules in consultation with an expert in the industry for which software is being developed. Each rule is based on a logical foundation; the machine will execute an output following the logical statement. When the system grows complex, more rules need to be written. It can quickly become unsustainable to maintain.

Data Rules

COMPUTER

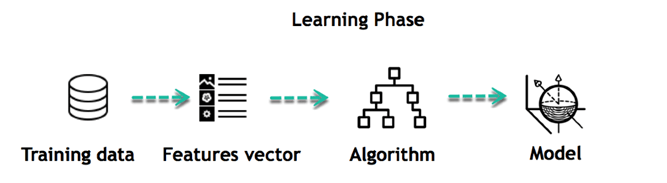
Output

## 5.2 How does Machine learning work?

Machine learning is the brain where all the learning takes place. The way the machine learns is similar to the human being. Humans learn from experience. The more we know, the more easily we can predict. By analogy, when we face an unknown situation, the likelihood of success is lower than the known situation. Machines are trained the same. To make an accurate prediction, the machine sees an example. When we give the machine a similar example, it can figure out the outcome. However, like a human, if it’s feed a previously unseen example, the machine has difficulties to predict.

The core objective of machine learning is the learningand inference. First of all, the machine learns through the discovery of patterns. This discovery is made thanks to the data. One crucial part of the data scientist is to choose carefully which data to provide to the machine. The list of attributes used to solve a problem is called a feature vector**.** You can think of a feature vector as a subset of data that is used to tackle a problem.

The machine uses some fancy algorithms to simplify the reality and transform this discovery into a model. Therefore, the learning stage is used to describe the data and summarize it into a model.



For instance, the machine is trying to understand the relationship between the wage of an individual and the likelihood to go to a fancy restaurant. It turns out the machine finds a positive relationship between wage and going to a high-end restaurant: This is the model

#### **Inferring**

When the model is built, it is possible to test how powerful it is on never-seen-before data. The new data are transformed into a features vector, go through the model and give a prediction. This is all the beautiful part of machine learning. There is no need to update the rules or train again the model. You can use the model previously trained to make inference on new data.

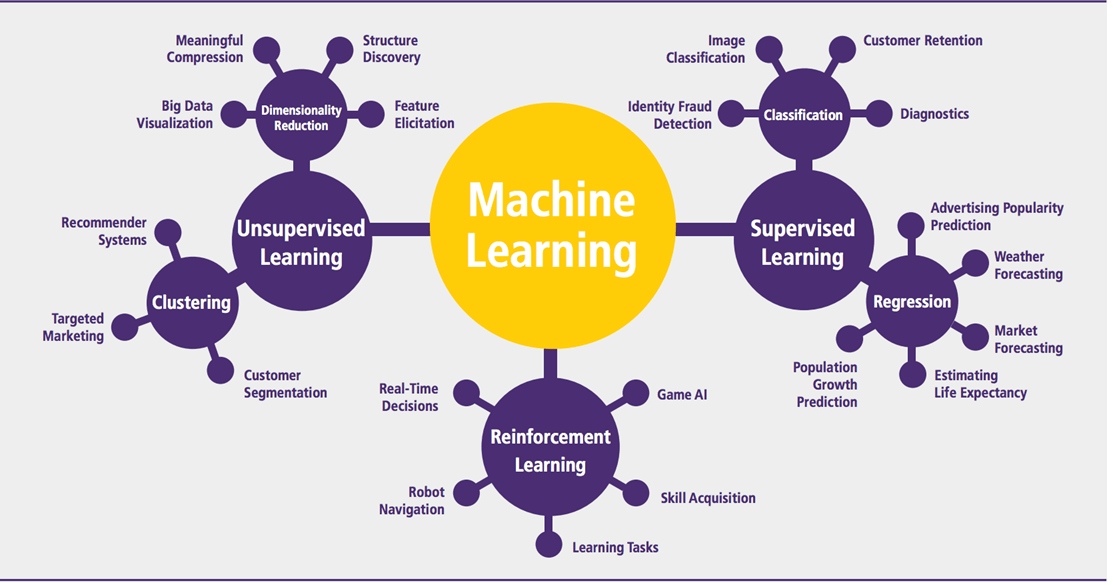


The life of Machine Learning programs is straightforward and can be summarized in the following points:

1. Define a question
2. Collect data
3. Visualize data
4. Train algorithm
5. Test the Algorithm
6. Collect feedback
7. Refine the algorithm
8. Loop 4-7 until the results are satisfying
9. Use the model to make a prediction

Once the algorithm gets good at drawing the right conclusions, it applies that knowledge to new sets of data.

**5.3 Machine learning Algorithms and where they are used?**



Machine learning can be grouped into two broad learning tasks: Supervised and Unsupervised. There are many other algorithms

#### **5.3.1 Supervised learning**

#### An algorithm uses training data and feedback from humans to learn the relationship of given inputs to a given output. For instance, a practitioner can use marketing expense and weather forecast as input data to predict the sales of cans.

You can use supervised learning when the output data is known. The algorithm will predict new data.

|  |  |  |
| --- | --- | --- |
| Algorithm Name | Description | Type |
| Linear regression | Finds a way to correlate each feature to the output to help predict future values. | Regression |
| Logistic regression | Extension of linear regression that's used for classification tasks. The output variable 3is binary (e.g., only black or white) rather than continuous (e.g., an infinite list of potential colors) | Classification |
| Decision tree | Highly interpretable classification or regression model that splits data-feature values into branches at decision nodes (e.g., if a feature is a color, each possible color becomes a new branch) until a final decision output is made | Regression Classification |
| Naive Bayes | The Bayesian method is a classification method that makes use of the Bayesian theorem. The theorem updates the prior knowledge of an event with the independent probability of each feature that can affect the event. | Regression Classification |
| Support vector machine | Support Vector Machine, or SVM, is typically used for the classification task. SVM algorithm finds a hyperplane that optimally divided the classes. It is best used with a non-linear solver. | Regression (not very common) Classification |
| Random forest | The algorithm is built upon a decision tree to improve the accuracy drastically. Random forest generates many times simple decision trees and uses the 'majority vote' method to decide on which label to return. For the classification task, the final prediction will be the one with the most vote; while for the regression task, the average prediction of all the trees is the final prediction. | Regression Classification |
| AdaBoost | Classification or regression technique that uses a multitude of models to come up with a decision but weighs them based on their accuracy in predicting the outcome | Regression Classification |
| Gradient-boosting trees | Gradient-boosting trees is a state-of-the-art classification/regression technique. It is focusing on the error committed by the previous trees and tries to correct it. | Regression Classification |

* Classification task
* Regression task

#### **Classification**

Imagine you want to predict the gender of a customer for a commercial. You will start gathering data on the height, weight, job, salary, purchasing basket, etc. from your customer database. You know the gender of each of your customer, it can only be male or female. The objective of the classifier will be to assign a probability of being a male or a female (i.e., the label) based on the information (i.e., features you have collected). When the model learned how to recognize male or female, you can use new data to make a prediction. For instance, you just got new information from an unknown customer, and you want to know if it is a male or female. If the classifier predicts male = 70%, it means the algorithm is sure at 70% that this customer is a male, and 30% it is a female.

The label can be of two or more classes. The above example has only two classes, but if a classifier needs to predict object, it has dozens of classes (e.g., glass, table, shoes, etc. each object represents a class)

#### **Regression**

When the output is a continuous value, the task is a regression. For instance, a financial analyst may need to forecast the value of a stock based on a range of feature like equity, previous stock performances, macroeconomics index. The system will be trained to estimate the price of the stocks with the lowest possible error.

#### **5.3.2 Unsupervised learning**

In unsupervised learning, an algorithm explores input data without being given an explicit output variable (e.g., explores customer demographic data to identify patterns)

You can use it when you do not know how to classify the data, and you want the algorithm to find patterns and classify the data for you

|  |  |  |
| --- | --- | --- |
| Algorithm | Description | Type |
| K-means clustering | Puts data into some groups (k) that each contains data with similar characteristics (as determined by the model, not in advance by humans) | Clustering |
| Gaussian mixture model | A generalization of k-means clustering that provides more flexibility in the size and shape of groups (clusters | Clustering |
| Hierarchical clustering | Splits clusters along a hierarchical tree to form a classification system.  Can be used for Cluster loyalty-card customer | Clustering |
| Recommender system | Help to define the relevant data for making a recommendation. | Clustering |
| PCA/T-SNE | Mostly used to decrease the dimensionality of the data. The algorithms reduce the number of features to 3 or 4 vectors with the highest variances. | Dimension Reduction |

**5.4 Application of Machine learning**

**Augmentation**:

* Machine learning, which assists humans with their day-to-day tasks, personally or commercially without having complete control of the output. Such machine learning is used in different ways such as Virtual Assistant, Data analysis, software solutions. The primary user is to reduce errors due to human bias.

**Automation**:

* Machine learning, which works entirely autonomously in any field without the need for any human intervention. For example, robots performing the essential process steps in manufacturing plants.

**Finance Industry**

* Machine learning is growing in popularity in the finance industry. Banks are mainly using ML to find patterns inside the data but also to prevent fraud.

**Government organization**

* The government makes use of ML to manage public safety and utilities. Take the example of China with the massive face recognition. The government uses Artificial intelligence to prevent jaywalker.

**Healthcare industry**

* Healthcare was one of the first industry to use machine learning with image detection.

**Marketing**

* Broad use of AI is done in marketing thanks to abundant access to data. Before the age of mass data, researchers develop advanced mathematical tools like Bayesian analysis to estimate the value of a customer. With the boom of data, marketing department relies on AI to optimize the customer relationship and marketing campaign.

**Example of application of Machine Learning in Supply Chain**

Machine learning gives terrific results for visual pattern recognition, opening up many potential applications in physical inspection and maintenance across the entire supply chain network.

Unsupervised learning can quickly search for comparable patterns in the diverse dataset. In turn, the machine can perform quality inspection throughout the logistics hub, shipment with damage and wear.

For instance, IBM's Watson platform can determine shipping container damage. Watson combines visual and systems-based data to track, report and make recommendations in real-time.

In past year stock manager relies extensively on the primary method to evaluate and forecast the inventory. When combining big data and machine learning, better forecasting techniques have been implemented (an improvement of 20 to 30 % over traditional forecasting tools). In term of sales, it means an increase of 2 to 3 % due to the potential reduction in inventory costs.

**Example of Machine Learning Google Car**

For example, everybody knows the Google car. The car is full of lasers on the roof which are telling it where it is regarding the surrounding area. It has radar in the front, which is informing the car of the speed and motion of all the cars around it. It uses all of that data to figure out not only how to drive the car but also to figure out and predict what potential drivers around the car are going to do. What's impressive is that the car is processing almost a gigabyte a second of data.

**5.5 Deep Learning**

Deep learning is a computer software that mimics the network of neurons in a brain. It is a subset of machine learning and is called deep learning because it makes use of deep neural networks. The machine uses different layers to learn from the data. The depth of the model is represented by the number of layers in the model. Deep learning is the new state of the art in term of AI. In deep learning, the learning phase is done through a neural network.

**5.5.1 Reinforcement Learning**

Reinforcement learningis a subfield of machine learning in which systems are trained by receiving virtual "rewards" or "punishments," essentially learning by trial and error. Google's DeepMind has used reinforcement learning to beat a human champion in the Go games. Reinforcement learning is also used in video games to improve the gaming experience by providing smarter bot.

One of the most famous algorithms are:

* Q-learning
* Deep Q network
* State-Action-Reward-State-Action (SARSA)
* Deep Deterministic Policy Gradient (DDPG)

**5.5.2 Applications/ Examples of deep learning applications**

* **AI in Finance:** The financial technology sector has already started using AI to save time, reduce costs, and add value. Deep learning is changing the lending industry by using more robust credit scoring. Credit decision-makers can use AI for robust credit lending applications to achieve faster, more accurate risk assessment, using machine intelligence to factor in the character and capacity of applicants.
* Underwrite is a Fintech company providing an AI solution for credit makers company. underwrite.ai uses AI to detect which applicant is more likely to pay back a loan. Their approach radically outperforms traditional methods.
* **AI in HR:** Under Armour, a sportswear company revolutionizes hiring and modernizes the candidate experience with the help of AI. In fact, Under Armour Reduces hiring time for its retail stores by 35%. Under Armour faced a growing popularity interest back in 2012. They had, on average, 30000 resumes a month. Reading all of those applications and begin to start the screening and interview process was taking too long. The lengthy process to get people hired and on-boarded impacted Under Armour's ability to have their retail stores fully staffed, ramped and ready to operate.
* At that time, Under Armour had all of the 'must have' HR technology in place such as transactional solutions for sourcing, applying, tracking and onboarding but those tools weren't useful enough. Under armour choose **HireVue**, an AI provider for HR solution, for both on-demand and live interviews. The results were bluffing; they managed to decrease by 35% the time to fill. In return, the hired higher quality staffs.

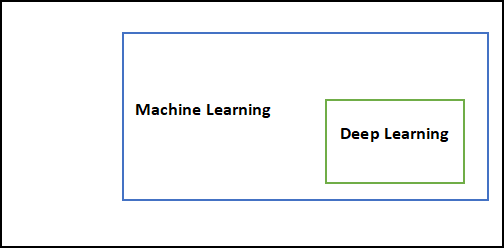
**AI in Marketing:**

AI is a valuable tool for customer service management and personalization challenges. Improved speech recognition in call-center management and call routing as a result of the application of AI techniques allows a more seamless experience for customers.

For example, deep-learning analysis of audio allows systems to assess a customer's emotional tone. If the customer is responding poorly to the AI chatbot, the system can be rerouted the conversation to real, human operators that take over the issue.

Apart from the three examples above, AI is widely used in other sectors/industries.

**Artificial Intelligence**



## Difference between Machine Learning and Deep Learning

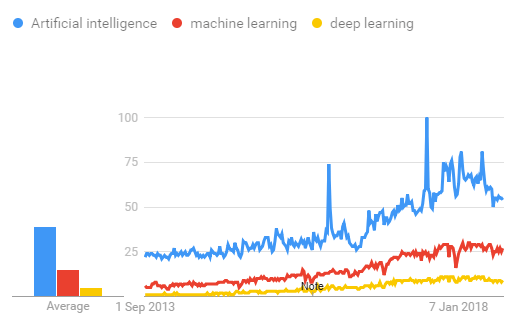
|  |  |  |
| --- | --- | --- |
|  | Machine Learning | Deep Learning |
| Data Dependencies | Excellent performances on a small/medium dataset | Excellent performance on a big dataset |
| Hardware dependencies | Work on a low-end machine. | Requires powerful machine, preferably with GPU: DL performs a significant amount of matrix multiplication |
| Feature engineering | Need to understand the features that represent the data | No need to understand the best feature that represents the data |
| Execution time | From few minutes to hours | Up to weeks. Neural Network needs to compute a significant number of weights |
| Interpretability | Some algorithms are easy to interpret (logistic, decision tree), some are almost impossible (SVM, XGBoost) | Difficult to impossible |

## When to use ML or DL?

In the table below, we summarize the difference between machine learning and deep learning.

|  |  |  |
| --- | --- | --- |
|  | Machine learning | Deep learning |
| Training dataset | Small | Large |
| Choose features | Yes | No |
| Number of algorithms | Many | Few |
| Training time | Short | Long |

With machine learning, you need fewer data to train the algorithm than deep learning. Deep learning requires an extensive and diverse set of data to identify the underlying structure. Besides, machine learning provides a faster-trained model. Most advanced deep learning architecture can take days to a week to train. The advantage of deep learning over machine learning is it is highly accurate. You do not need to understand what features are the best representation of the data; the neural network learned how to select critical features. In machine learning, you need to choose for yourself what features to include in the model.



## 5.6 TensorFlow

The most famous deep learning library in the world is Google's TensorFlow. Google product uses machine learning in all of its products to improve the search engine, translation, image captioning or recommendations.

To give a concrete example, Google users can experience a faster and more refined the search with AI. If the user types a keyword a the search bar, Google provides a recommendation about what could be the next word.

Google wants to use machine learning to take advantage of their massive datasets to give users the best experience. Three different groups use machine learning:

* Researchers
* Data scientists
* Programmers.

They can all use the same toolset to collaborate with each other and improve their efficiency.

Google does not just have any data; they have the world's most massive computer, so TensorFlow was built to scale. TensorFlow is a library developed by the Google Brain Team to accelerate machine learning and deep neural network research.

It was built to run on multiple CPUs or GPUs and even mobile operating systems, and it has several wrappers in several languages like Python, C++ or Java.

In this tutorial, you will learn

**TensorFlow Architecture**

Tensor flow architecture works in three parts:

* Pre-processing the data
* Build the model
* Train and estimate the model

It is called Tensor flow because it takes input as a multi-dimensional array, also known as tensors. You can construct a sort of flowchart of operations (called a Graph) that you want to perform on that input. The input goes in at one end, and then it flows through this system of multiple operations and comes out the other end as output.

This is why it is called TensorFlow because the tensor goes in it flows through a list of operations, and then it comes out the other side.

**Where can Tensor flow run?**

TensorFlow can hardware, and software requirements can be classified into

1. Development Phase: This is when you train the mode. Training is usually done on your Desktop or laptop.
2. Run Phase or Inference Phase: Once training is done Tensorflow can be run on many different platforms. You can run it on

* Desktop running Windows, macOS or Linux
* Cloud as a web service
* Mobile devices like iOS and Android

You can train it on multiple machines then you can run it on a different machine, once you have the trained model.

The model can be trained and used on GPUs as well as CPUs. GPUs were initially designed for video games. In late 2010, Stanford researchers found that GPU was also very good at matrix operations and algebra so that it makes them very fast for doing these kinds of calculations. Deep learning relies on a lot of matrix multiplication. TensorFlow is very fast at computing the matrix multiplication because it is written in C++. Although it is implemented in C++, TensorFlow can be accessed and controlled by other languages mainly, Python.

Finally, a significant feature of Tensor Flow is the Tensor Board. The Tensor Board enables to monitor graphically and visually what TensorFlow is doing.

**List of Prominent Algorithms supported by TensorFlow**

* Linear regression: tf. estimator. Linear Regressor
* Classification: tf. Estimator. Linear Classifier
* Deep learning classification: tf. estimator. DNN Classifier
* Booster tree regression: tf. estimator. Boosted Trees Regressor
* Boosted tree classification: tf. estimator. Boosted Trees Classifier­­

**5.7 REQUIRMENT ANALYSIS**

**5.7.1 Python Overview**

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

* **Python is Interpreted:** Python is processed at runtime by the interpreter.You do not need to compile your program before executing it. This is similar to PERL and PHP.
* **Python is Interactive:** You can actually sit at a Python prompt and interactwith the interpreter directly to write your programs.
* **Python is Object-Oriented:** Python supports Object-Oriented style ortechnique of programming that encapsulates code within objects.
* **Python is a Beginner's Language:** Python is a great language for thebeginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

**History of Python**

* Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.
* Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, Unix shell, and other scripting languages.
* Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).
* Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

**Python Features**

Python's features include:

* **Easy-to-learn:** Python has few keywords, simple structure, and a clearlydefined syntax. This allows the student to pick up the language quickly.
* **Easy-to-read:** Python code is more clearly defined and visible to the eyes.
* **Easy-to-maintain:** Python's source code is fairly easy-to-maintain.
* **A broad standard library:** Python's bulk of the library is very portable andcross-platform compatible on UNIX, Windows, and Macintosh.
* **Interactive Mode:** Python has support for an interactive mode which allowsinteractive testing and debugging of snippets of code.
* **Portable:** Python can run on a wide variety of hardware platforms and has thesame interface on all platforms.
* **Extendable:** You can add low-level modules to the Python interpreter. Thesemodules enable programmers to add to or customize their tools to be more efficient.
* **Databases:** Python provides interfaces to all major commercial databases.

**GUI Programming:** Python supports GUI applications that can be created andported to many system calls, libraries, and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.

**Scalable:** Python provides a better structure and support for large programsthan shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below:

* IT supports functional and structured programming methods as well as OOP.
* It can be used as a scripting language or can be compiled to byte-code for building large applications.
* It provides very high-level dynamic data types and supports dynamic type checking.
* IT supports automatic garbage collection.
* It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

## 5.7.2 PYTHON ENVIRONMENT

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

## Python’s standard library

* Pandas
* Numpy
* Sklearn
* seaborn
* matplotlib
* Importing Datasets

## PANDAS

Pandas is quite a game changer when it comes to analysing data with Python and it is one of the most preferred and widely used tools in [data munging/wrangling](https://en.wikipedia.org/wiki/Data_wrangling) if not THE most used one.

Pandas is an open source What’s cool about Pandas is that it takes data (like a CSV or TSV file, or a SQL database) and creates a Python object with rows and columns called data frame that looks very similar to table in a statistical software (think Excel or SPSS for example. People who are familiar with R would see similarities to R too). This is so much easier to work with in comparison to working with lists and/or dictionaries through for loops or list comprehension.

## Installation and Getting Started

In order to “get” Pandas you would need to install it. You would also need to have Python 2.7 and above as a pre-requirement for installation. It is also dependent on other libraries (like NumPy) and has optional dependencies (like Matplotlib for plotting). Therefore, I think that the easiest way to get Pandas set up is to install it through a package like the [Anaconda distribution](https://www.continuum.io/downloads) , “a cross platform distribution for data analysis and scientific computing.”

In order to use Pandas in your Python IDE ([Integrated Development Environment](https://en.wikipedia.org/wiki/Integrated_development_environment)) like Jupiter Notebook or Spyder (both of them come with Anaconda by default), you need to import the Pandas library first. Importing a library means loading it into the memory and then it’s there for you to work with. In order to import Pandas all, you have to do is run the following code:

* **import pandas as pd**
* **import numpy as np**

Usually you would add the second part (‘as pd’) so you can access Pandas with ‘pd.command’ instead of needing to write ‘pandas.command’ every time you need to use it. Also, you would import numpy as well, because it is very useful library for scientific computing with Python. Now Pandas is ready for use! Remember, you would need to do it every time you start a new Jupyter Notebook, Spyder file etc.

**Working with Pandas**

**Loading and Saving Data with Pandas**

When you want to use Pandas for data analysis, you’ll usually use it in one of three different ways:

* Convert a Python’s list, dictionary or Numpy array to a Pandas data frame
* Open a local file using Pandas, usually a CSV file, but could also be a delimited text file (like TSV), Excel, etc
* Open a remote file or database like a CSV or a JSONon a website through a URL or read from a SQL table/database

There are different commands to each of these options, but when you open a file, they would look like this:

* **pd.read\_filetype()**

As I mentioned before, there are different filetypes Pandas can work with, so you would replace “filetype” with the actual, well, filetype (like CSV). You would give the path, filename etc inside the parenthesis. Inside the parenthesis you can also pass different arguments that relate to how to open the file. There are numerous arguments and in order to know all you them, you would have to read the documentation (for example, the [documentation for pd.read\_csv()](http://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html) would contain all the arguments you can pass in this Pandas command.

In order to convert a certain Python object (dictionary, lists etc) the basic command is:

* **pd.DataFrame()**

Inside the parenthesis you would specify the object(s) you’re creating the data frame from. This command also has [different arguments](http://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.html) .

You can also save a data frame you’re working with/on to different kinds of files (like CSV, Excel, JSON and SQL tables). The general code for that is:

* **df.to\_filetype(filename)**

## Viewing and Inspecting Data

Now that you’ve loaded your data, it’s time to take a look. How does the data frame look? Running the name of the data frame would give you the entire table, but you can also get the first n rows with df.head(n) or the last n rows with df.tail(n). df.shape would give you the number of rows and columns. df.info() would give you the index, datatype and memory information. The command s.value\_counts(dropna=False) would allow you to view unique values and counts for a series (like a column or a few columns). A very useful command is df.describe() which inputs summary statistics for numerical columns. It is also possible to get statistics on the entire data frame or a series (a column etc):

* df.mean() Returns the mean of all columns
* df.corr() Returns the correlation between columns in a data frame
* df.count() Returns the number of non-null values in each data frame column
* df.max() Returns the highest value in each column
* df.min() Returns the lowest value in each column
* df.median() Returns the median of each column
* df.std() Returns the standard deviation of each column.

## Selection of Data

One of the things that is so much easier in Pandas is selecting the data you want in comparison to selecting a value from a list or a dictionary. You can select a column (df[col]) and return column with label col as Series or a few columns (df[[col1, col2]]) and returns columns as a new DataFrame. You can select by position (s.iloc[0]), or by index (s.loc['index\_one']) . In order to select the first row you can use df.iloc[0,:] and in order to select the first element of the first column you would run df.iloc[0,0] . These can also be used in different combinations, so I hope it gives you an idea of the different selection and indexing you can perform in Pandas.

## Filter, Sort and Groupby

You can use different conditions to filter columns. For example, df[df[year] > 1984] would give you only the column year is greater than 1984. You can use & (and) or | (or) to add different conditions to your filtering. This is also called boolean filtering.

It is possible to sort values in a certain column in an ascending order using df.sort\_values(col1) ; and also in a descending order using df.sort\_values(col2,ascending=False). Furthermore, it’s possible to sort values by col1 in ascending order then col2 in descending order by using df.sort\_values([col1,col2],ascending=[True,False]).

The last command in this section is group by. It involves splitting the data into groups based on some criteria, applying a function to each group independently and combining the results into a data structure. df.groupby(col) returns a group by object for values from one column while df.groupby([col1,col2]) returns a group by object for values from multiple columns.

## Data Cleaning

Data cleaning is a very important step in data analysis. For example, we always check for missing values in the data by running pd.isnull() which checks for null Values, and returns a boolean array (an array of true for missing values and false for non-missing values). In order to get a sum of null/missing values, run pd.isnull().sum(). pd.notnull() is the opposite of pd.isnull(). After you get a list of missing values you can get rid of them, or drop them by using df.dropna() to drop the rows or df.dropna(axis=1) to drop the columns. A different approach would be to fill the missing values with other values by using df.fillna(x) which fills the missing values with x (you can put there whatever you want) or s.fillna(s.mean()) to replace all null values with the mean (mean can be replaced with almost any function from the statistics section).

It is sometimes necessary to replace values with different values. For example, s.replace(1,'one') would replace all values equal to 1 with 'one'. It’s possible to do it for multiple values: s.replace([1,3],['one','three'])would replace all 1 with 'one' and 3 with 'three'. You can also rename specific columns by running: df.rename(columns={'old\_name': 'new\_ name'})or use df.set\_index('column\_one') to change the index of the data frame.

## Join/Combine

The last set of basic Pandas commands are for joining or combining data frames or rows/columns. The three commands are:

* df1.append(df2)— add the rows in df1 to the end of df2 (columns should be identical)
* df.concat([df1, df2],axis=1) — add the columns in df1 to the end of df2 (rows should be identical)
* df1.join(df2,on=col1,how='inner') — SQL-style join the columns in df1with the columns on df2 where the rows for colhave identical values. how can be equal to one of: 'left', 'right', 'outer', 'inner'.

## NUMPY

Numpy is one such powerful library for array processing along with a large collection of high- level mathematical functions to operate on these arrays. These functions fall into categories like Linear Algebra, Trigonometry, Statistics, Matrix manipulation, etc.

## Getting NumPy

NumPy’s main object is a homogeneous multidimensional array. Unlike python’s array class which only handles one-dimensional array, NumPy’s ndarray class can handle multidimensional array and provides more functionality. NumPy’s dimensions are known as axes. For example, the array below has 2 dimensions or 2 axes namely rows and columns. Sometimes dimension is also known as a rank of that particular array or matrix.

## Importing NumPy

NumPy is imported using the following command. Note here np is the convention followed for the alias so that we don't need to write numpy every time.

* + import numpy as np

NumPy is the basic library for scientific computations in Python and this article illustrates some of its most frequently used functions. Understanding NumPy is the first major step in the journey of machine learning and deep learning.

## Sklearn

In python, scikit-learn library has a pre-built functionality under sklearn. Pre processing.

Next thing is to do feature extraction Feature extraction is an attribute reduction process. Unlike feature selection, which ranks the existing attributes according to their predictive significance, feature extraction actually transforms the attributes. The transformed attributes, or features, are linear combinations of the original attributes. Finally our models are trained using Classifier algorithm.. We use nltk . classify module on Natural Language Toolkit library on Python. We use the labelled dataset gathered . The rest of our labelled data will be used to evaluate the models. Some machine learning algorithms were used to classify pre processed data. The chosen classifiers were Decision tree , Support Vector Machines and Random forest. These algorithms are very popular in text classification tasks.

## SEABORN

**Data Visualization in Python**

Data visualization is the discipline of trying to understand data by placing it in a visual context, so that patterns, trends and correlations that might not otherwise be detected can be exposed.

Python offers multiple great graphing libraries that come packed with lots of different features. No matter if you want to create interactive, live or highly customized plots python has a excellent library for you.

## To get a little overview here are a few popular plotting libraries:

* [Matplotlib:](https://matplotlib.org/) low level, provides lots of freedom
* [Pandas Visualization:](https://pandas.pydata.org/pandas-docs/stable/visualization.html) easy to use interface, built on Matplotlib
* [Seaborn:](https://seaborn.pydata.org/) high-level interface, great default styles
* [ggplot:](http://ggplot.yhathq.com/) based on R’s ggplot2, uses [Grammar of Graphics](https://www.amazon.com/Grammar-Graphics-Statistics-Computing/dp/0387245448)
* [Plotly:](https://plot.ly/python/) can create interactive plots

In this article, we will learn how to create basic plots using Matplotlib, Pandas visualization and Seaborn as well as how to use some specific features of each library. This article will focus on the syntax and not on interpreting the graphs.

## Matplotlib

Matplotlib is the most popular python plotting library. It is a low level library with a Matlab like interface which offers lots of freedom at the cost of having to write more code.

1. To install Matplotlib pip and conda can be used.
2. pip install matplotlib
3. conda install matplotlib

Matplotlib is specifically good for creating basic graphs like line charts, bar charts, histograms and many more. It can be imported by typing:

* + **import matplotlib.pyplot as plt**

## Line Chart

In Matplotlib we can create a line chart by calling the plot method. We can also plot multiple columns in one graph, by looping through the columns we want, and plotting each column on the same axis.

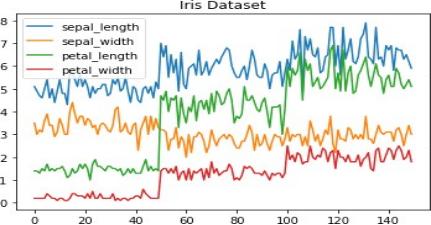


Fig : Line chart

## Histogram

In Matplotlib we can create a Histogram using the hist method. If we pass it categorical data like the points column from the wine-review dataset it will automatically calculate how often each class occurs.

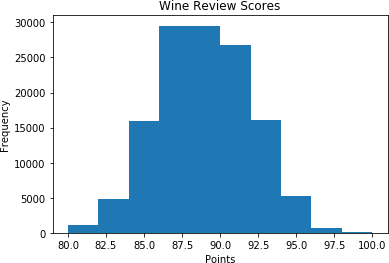


Fig: Histogram

## Bar Chart

A bar-chart can be created using the bar method. The bar-chart isn’t automatically calculating the frequency of a category so we are going to use pandas value\_counts function to do this. The bar- chart is useful for categorical data that doesn’t have a lot of different categories (less than 30) because else it can get quite messy.

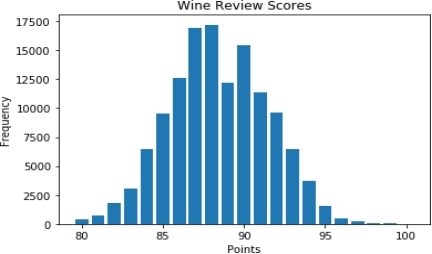


Fig: Bar Chat

## Pandas Visualization

Pandas is a open source high-performance, easy-to-use library providing data structures, such as dataframes, and data analysis tools like the visualization tools we will use in this article.

Pandas Visualization makes it really easy to create plots out of a pandas dataframe and series. It also has a higher level API than Matplotlib and therefore we need less code for the same results.

* **Pandas can be installed using either pip or conda**
* **pip install pandas**
* **conda install pandas**

## Heatmap

A Heatmap is a graphical representation of data where the individual values contained in a matrix are represented as colors. Heatmaps are perfect for exploring the correlation of features in a dataset.

To get the correlation of the features inside a dataset we can call <dataset>.corr() , which is a Pandas dataframe method. This will give use the correlation matrix.

We can now use either Matplotlib or Seaborn to create the heatmap.

## Matplotlib:

Fig: Heatmap without annotations



Data visualization is the discipline of trying to understand data by placing it in a visual context, so that patterns, trends, and correlations that might not otherwise be detected can be exposed.

Python offers multiple great graphing libraries that come packed with lots of different features. In this article we looked at Matplotlib, Panda’s visualization and Seaborn.

**5.7.3 ANACONDA NAVIGATOR**

Anaconda Navigator is a desktop graphical user interface (GUI) included in Anaconda distribution that allows you to launch applications and easily manage conda packages, environments and channels without using command-line commands. Navigator can search for packages on Anaconda Cloud or in a local Anaconda Repository. It is available for Windows, mac OS and Linux.

## Why use Navigator?

In order to run, many scientific packages depend on specific versions of other packages. Data scientists often use multiple versions of many packages, and use multiple environments to separate these different versions.

The command line program conda is both a package manager and an environment manager, to help data scientists ensure that each version of each package has all the dependencies it requires and works correctly.

Navigator is an easy, point-and-click way to work with packages and environments without needing to type conda commands in a terminal window. You can use it to find the packages you want, install them in an environment, run the packages and update them, all inside Navigator.

## What applications can i access using navigator?

The following applications are available by default in Navigator:

* Jupyter Lab
* Jupyter Notebook
* QT Console
* Spyder
* VS Code
* Glue viz
* Orange 3 App
* Rodeo
* RStudio

Advanced conda users can also build your own Navigator applications.

## How can I run code with Navigator?

The simplest way is with Spyder. From the Navigator Home tab, click Spyder, and write and execute your code.

You can also use Jupyter Notebooks the same way. Jupyter Notebooks are an increasingly popular system that combine your code, descriptive text, output, images and interactive interfaces into a single notebook file that is edited, viewed and used in a web browser.

## What’s new in 1.9?

* Add support for Offline Mode for all environment related actions.
* Add support for custom configuration of main windows links.
* Numerous bug fixes and performance enhancements.

# **CHAPTER 6**

# **ALGORITHMS**

# **6.1 K-MEANS CLUSTERING ALGORITHM**

K-Means Clustering is an Unsupervised learning algorithm which groups the unlabelled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.

It is an iterative algorithm that divides the unlabeled dataset into k different clusters in such a way that each dataset belongs only one group that has similar properties.

It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training.

It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means clustering algorithm mainly performs two tasks:

* Determines the best value for K centre points or centroids by an iterative process.
* Assigns each data point to its closest k-centre. Those data points which are near to the particular k-centre, create a cluster.

## How does the K-Means Algorithm Work?

The working of the K-Means algorithm is explained in the below steps:

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

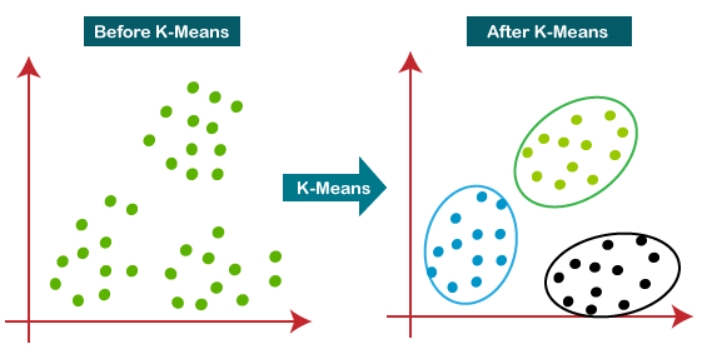
Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.



**6.2 RANDOM FOREST ALGORITHM**

Random forest is a type of supervised machine learning algorithm based on ensemble learning. Ensemble learning is a type of learning where you join different types of algorithms or same algorithm multiple times to form a more powerful prediction model. The random forest algorithm combines multiple algorithms of the same type i.e. multiple decision *trees*, resulting in a forest of trees, hence the name "Random Forest". The random forest algorithm can be used for both regression and classification tasks.

## HOW RANDOM FOREST WORKS

The following are the basic steps involved in performing the random forest algorithm:

* Pick N random records from the dataset.
* Build a decision tree based on these N records.
* Choose the number of trees you want in your algorithm and repeat steps 1 and 2.
* For classification problem, each tree in the forest predicts the category to which the new record belongs. Finally, the new record is assigned to the category that wins the majority vote.

## ADVANTAGES OF USING RANDOM FOREST

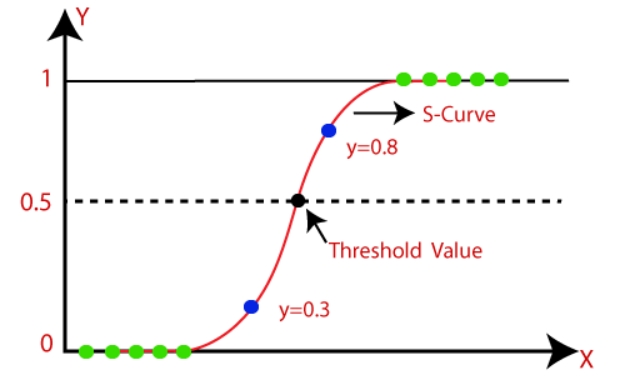
Pros of using random forest for classification and regression.

* The random forest algorithm is not biased, since, there are multiple trees and each tree is trained on a subset of data. Basically, the random forest algorithm relies on the power of "the crowd"; therefore, the overall biasedness of the algorithm is reduced.
* This algorithm is very stable. Even if a new data point is introduced in the dataset the overall algorithm is not affected much since new data may impact one tree, but it is very hard for it to impact all the trees.
* The random forest algorithm works well when you have both categorical and numerical features.
* The random forest algorithm also works well when data has missing values or it has not been scaled.



# **6.3 LOGISTIC REGRESSION**

* Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.
* Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.
* Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.
* In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).
* The curve from the logistic function indicates the likelihood of something such as whether the cells are cancerous or not, a mouse is obese or not based on its weight, etc.
* Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.
* Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:



## 6.4 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification and regression challenges. It is generally utilized in characterization issues. In the SVM, we plot every given thing as a point in n-dimensional space (where n is number of highlights you have) with the calculation of each element being the estimation of a specific arrange. At that point, we perform order by finding the hyper-plane that separates the two classes quite well. Bolster Vectors are essentially the co-ordinates of individual perception. The SVM classifier is a wilderness which best isolates the two classes (hyper-plane/line).

SVM is binary classification algorithm. Given a set of points of 2 types in N dimensional place, SVM generates a (N — 1) dimensional hyper-plane to separate those points into 2 groups. Say you have some points of 2 types in a paper which are linearly separable. SVM will find a straight line which separates those points into 2 types and situated as far as possible from all those points.

## How does it work?

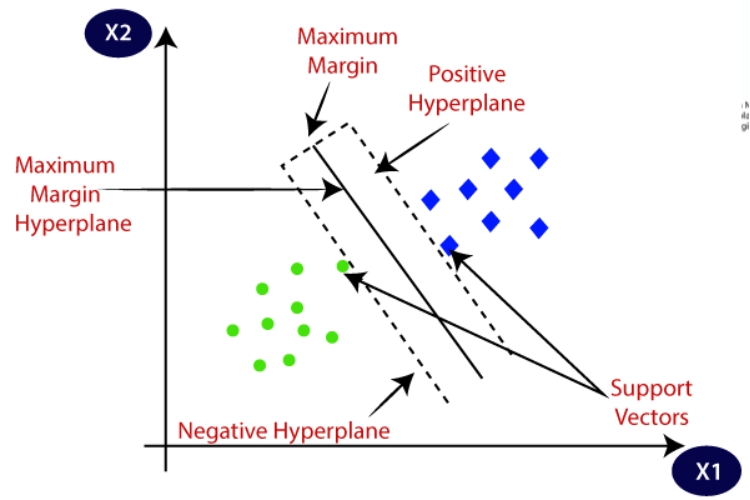
Above, we got accustomed to the process of segregating the two classes with a hyper-plane. Now the burning question is “How can we identify the right hyper-plane?”. Don’t worry, it’s not as hard as you think!

Let’s understand:

* Identify the right hyper-plane
* Classify two classes
* Find the hyper-plane to segregate to classes

## Advantages

* It works really well with a clear margin of separation
* It is effective in high dimensional spaces.
* It is effective in cases where the number of dimensions is greater than the number of samples.
* It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient



**CHAPTER 7**

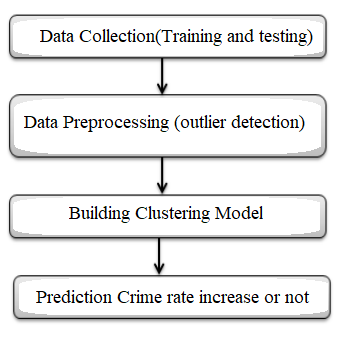
**IMPLEMENTATION AND CODING**

**7.1 MODULES**

1. DATA COLLECTION
2. DATA PRE-PROCESSING
3. FEATURE EXTRATION
4. EVALUATION MODEL

**Process of machine learning algorithms**

Flow diagram of machine learning algorithms: -



**DATA COLLECTION**

Data collection is a process in which information is gathered from many sources which is later used to develop the machine learning models. The data should be stored in a way that makes sense for problem. In this step the data set is converted into the understandable format which can be fed into machine learning models.

Data used in this paper is a set of cervical cancer data with 15 features. This step is concerned with selecting the subset of all available data that you will be working with. ML problems start with data preferably, lots of data (examples or observations) for which you already know the target answer. Data for which you already know the target answer is called labelled data.

**DATA PRE-PROCESSING**

Organize your selected data by formatting, cleaning and sampling from it.

Three common data pre-processing steps are:

* **Formatting:** The data you have selected may not be in a format that is suitable for you to work with. The data may be in a relational database and you would like it in a flat file, or the data may be in a proprietary file format and you would like it in a relational database or a text file.
* **Cleaning:** Cleaning data is the removal or fixing of missing data. There may be data instances that are incomplete and do not carry the data you believe you need to address the problem. These instances may need to be removed. Additionally, there may be sensitive information in some of the attributes and these attributes may need to be anonymized or removed from the data entirely.
* **Sampling:** There may be far more selected data available than you need to work with. More data can result in much longer running times for algorithms and larger computational and memory requirements. You can take a smaller representative sample of the selected data that may be much faster for exploring and prototyping solutions before considering the whole dataset.

**FEATURE EXTRATION**

Next thing is to do Feature extraction is an attribute reduction process. Unlike feature selection, which ranks the existing attributes according to their predictive significance, feature extraction actually transforms the attributes. The transformed attributes, or features, are linear combinations of the original attributes. Finally, our models are trained using Classifier algorithm. We use classify module on Natural Language Toolkit library on Python. We use the labelled dataset gathered. The rest of our labelled data will be used to evaluate the models. Some machine learning algorithms were used to classify pre-processed data. The chosen classifiers were Random Forest. These algorithms are very popular in text classification tasks.

**EVALUATION MODEL**

Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can easily generate overoptimistic and over fitted models. There are two methods of evaluating models in data science, Hold-Out and Cross-Validation. To avoid over fitting, both methods use a test set (not seen by the model) to evaluate model performance.

Performance of each classification model is estimated base on its averaged. The result will be in the visualized form. Representation of classified data in the form of graphs.

Accuracy is defined as the percentage of correct predictions for the test data. It can be calculated easily by dividing the number of correct predictions by the number of total predictions.

**Proposed Approach Steps:**

1. First, we take cervical dataset.

2. Filter dataset according to requirements and create a new dataset which has attribute according to analysis to be done.

3. Perform Pre-Processing on the dataset.

4. Split the data into training and testing.

5. Train the model with training data then analyses testing dataset over classification algorithm.

6. Finally you will get results as accuracy metrics.

**7.2 SAMPLE CODE**

**// Import Modules //**

import matplotlib.pyplot as plt

%matplotlib inline

import numpy as np

import pandas as pd

// **Loading the Dataset //**

df = pd.read\_csv('2001-2012.csv')

df.head()

years\_title = [str(i) for i in range(2001,2013)]

STATES\_IN\_INDIA = df['STATE/UT'].unique()

STATES\_IN\_INDIA = STATES\_IN\_INDIA[:-4]

STATES\_IN\_INDIA

TYPES\_OF\_CASES = df['CRIME HEAD'].unique()

TYPES\_OF\_CASES = TYPES\_OF\_CASES[:-1]

TYPES\_OF\_CASES

for state in STATES\_IN\_INDIA:

fig=plt.figure(figsize=(12, 8), dpi= 80, facecolor='w', edgecolor='k')

plt.title(state)

plt.xlabel('Years')

plt.ylabel('No of Cases')

for case in TYPES\_OF\_CASES:

temp\_df = df[(df['STATE/UT'] == state ) & (df['CRIME HEAD'] == case)]

N\_cases = [temp\_df[c].values[0] for c in years\_title]

plt.plot(years\_title,N\_cases)

plt.legend(TYPES\_OF\_CASES)

fig=plt.figure(figsize=(20, 10), dpi= 80, facecolor='w', edgecolor='k')

plt.title('TOTAL CRIME YEAR WISE')

plt.xlabel('Years')

plt.ylabel('No of Cases')

for state in STATES\_IN\_INDIA:

temp\_df = df[(df['STATE/UT'] == state ) & (df['CRIME HEAD'] == 'TOTAL CRIMES AGAINST WOMEN')]

N\_cases = [temp\_df[c].values[0] for c in years\_title]

plt.plot(years\_title,N\_cases)

plt.legend(STATES\_IN\_INDIA)

print('Data set:')

for col\_name in df.columns:

if df[col\_name].dtypes == 'object' :

unique\_cat = len(df[col\_name].unique())

print("Feature '{col\_name}' has {unique\_cat} categories".format(col\_name=col\_name, unique\_cat=unique\_cat))

print()

f, ax = plt.subplots(figsize=(24, 15))

stats = df.sort\_values([ "cluster", "STATE/UT"], ascending=True)

sns.set\_color\_codes("pastel")

sns.barplot(y="STATE/UT", x="2012", data=stats)

sns.despine(left=True, bottom=True)

**// Pre-processing the Dataset //**

from sklearn import preprocessing

lab=preprocessing.LabelEncoder()

#df['STATE/UT']=lab.fit\_transform(df['STATE/UT'])

df['CRIME HEAD']=lab.fit\_transform(df['CRIME HEAD'])

df.head()

df

**// K-Means Clustering //**

from sklearn.cluster import KMeans

kmeans = KMeans(n\_clusters=9)

kmeans.fit(df.iloc[:,1:])

kmeans.cluster\_centers\_

labels=kmeans.labels\_

labels

kmeans.inertia\_

kmeans.score

cust = [[7,871,1002,946,1016,935,1049,1070,1257,1188,1362,1442,1341]]

kmeans.predict(cust)[0]

**// Import Packages //**

import numpy as np

unique, counts = np.unique(kmeans.labels\_, return\_counts=True)

dict\_data = dict(zip(unique, counts))

dict\_data

import seaborn as sns

sns.lmplot('2011', '2012', data=df, hue='cluster', palette='coolwarm', size=5, aspect=1, fit\_reg=False)

**// Data Splitting //**

X = df.iloc[:,1:14]

y = df.iloc[:,df.columns=='cluster']

print(X.head())

y.head()

**// Model Training //**

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X,y, test\_size = 0.2, random\_state=0)

**// Random Forest //**

from sklearn.ensemble import RandomForestClassifier

random\_forest = RandomForestClassifier(n\_estimators=100)

random\_forest.fit(X\_train,y\_train)

y\_pred = random\_forest.predict(X\_test)

print(y\_pred)

from sklearn.metrics import accuracy\_score,classification\_report

acc1=accuracy\_score(y\_pred,y\_test)

print(acc1)

clf=classification\_report(y\_pred,y\_test)

print(clf)

**// Logistic Regression //**

from sklearn.linear\_model import LogisticRegression

lg=LogisticRegression()

lg.fit(X\_train,y\_train)

predic2=lg.predict(X\_test)

acc2=accuracy\_score(predic2,y\_test)

print(acc2)

clf2=classification\_report(predic2,y\_test)

print(clf2)

**// Support Vector Machine //**

from sklearn import svm

sv=svm.LinearSVC()

sv.fit(X\_train,y\_train)

predic3=sv.predict(X\_test)

acc3=accuracy\_score(predic3,y\_test)

print(acc3)

clf3=classification\_report(predic3,y\_test)

print(clf3)

**// Visualization //**

import matplotlib.pyplot as plt; plt.rcdefaults()

objects = ('Random Forest','LogisticRegression','Support Vector')

y\_pos = np.arange(len(objects))

performance = [acc1,acc2,acc3]

plt.bar(y\_pos, performance, align='center', alpha=0.5)

plt.xticks(y\_pos, objects)

plt.ylabel('Accuracy')

plt.title('RF vs LR vs SVM')

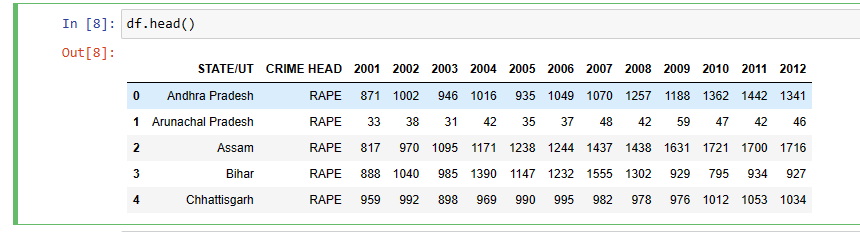
plt.show()

**CHAPTER 8**

**OUTPUT SCREENS**

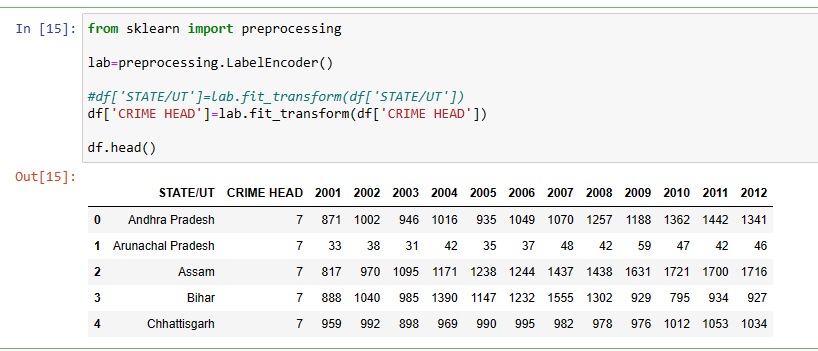
**Screen Number: 1**

## Screen Title: Dataset



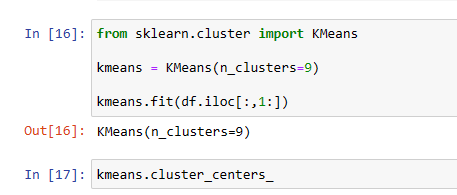
**Screen Number: 2**

## Screen Title: Data Pre-processing



**Screen Number: 3**

**Screen Title: K-means Clustering Algorithm**

****

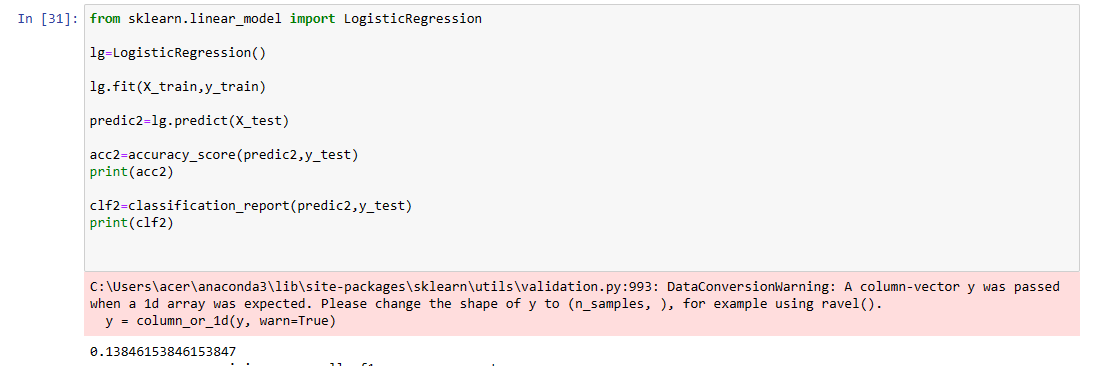
**Screen Number: 4**

## Screen Title: Random Forest Model Result



**Screen Number: 5**

**Screen Title: Logistic Regression**

****

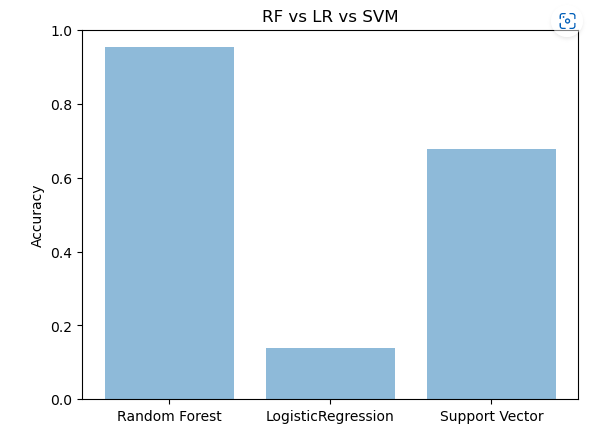
**Screen Number: 6**

**Screen Title: Support Vector Machine**



**Screen Number: 7**

## Screen Title: Data Exploration



**CHAPTER 9**

**SYSTEM TESTING**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

Software Testing can also be stated as the process of validating and verifying that a software program/application/product:

* Meets the business and technical requirements that guided its design and Development.
* Works as expected and can be implemented with the same characteristics.

* 1. **TESTING METHODS**
     1. **Functional Testing**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

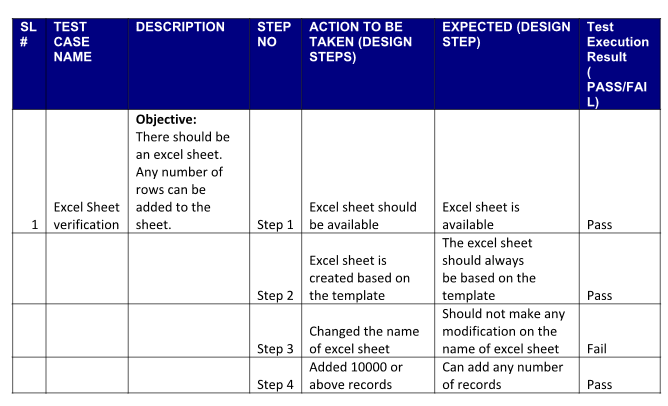
Functional testing is centred on the following items:

* Functions: Identified functions must be exercised.
* Output: Identified classes of software outputs must be exercised.
* Systems/Procedures: system should work properly.
  + 1. **Integration Testing**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

**Test Case for Excel Sheet Verification:**

Here in machine learning we are dealing with dataset which is in excel sheet format so if any test case we need means we need to check excel file. Later on, classification will work on the respective columns of dataset.



**RESULTS**

Data mining is a process to extract knowledge from existing data. It is used as a tool in banking and finance, in general, to discover useful information from the operational and historical data to enable better decision-making. It is an interdisciplinary field, the confluence of Statistics, Database technology, Information science, Machine learning, and Visualization. It involves steps that include data selection, data integration, data transformation, data mining, pattern evaluation, knowledge presentation.

**CHAPTER 10**

**CONCLUSION**

The implementation has been done in Python language. Here we find out which state has more or fewer criminals occur based on each cluster’s values. It is really helpful for the authorities to be more aware. Clustering is the process of creating different groups consisting of similar data points.

The points in one cluster are as similar as possible are there are no similarities between different cluster points. The result of the optimized k-means algorithm is efficient and provides improved accuracy of the final cluster reduced the number of iterations. In the future, the result of crime analysis can be used to make various strategies for crime control and the optimal deployment of resources in crime avoidance.

**CHAPTER 11**

**FUTURE SCOPE**

Though it overcomes the problem of the existing work, it has some limitations. In the situation of absence of class labels, then the probability of the estimation will be zero. As a future extension of the proposed work, the application of more machine learning classification models proves to increase accuracy in crime prediction and will enhance the overall performance. It helps in providing a better study for the future improvement by taking the income information into consideration for neighborhoods places in order to foresee if any relationship between the income levels of a particular in the neighborhood places and their crime rate.

**BIBLIOGRAPHY**

1. Shiju Sathyadevan M.S, Surya Gangadharan: Crime Analysis and Prediction Using Data Mining, in Networks Soft Computing (ICNSC), (2014) First International Conference. https://ieeexplore.ieee.org/document/6906719.

2. H. Benjamin Fredrick David A. Suruliandi: Survey on crime analysis and prediction using data mining techniques. Department of Computer Science and Engineering, Manonmaniam Sundaranar University, India. Ictact journal on soft computing, April (2017). <https://www.researchgate.net/publication/3222541877_SURVEY> ON CRIME ANALAYSIS AND PREDICTION USING DATA MINING TECHNIQUES.

3. Jesia Quader Yuki, Md.Mahfil Quader Sakib, Zaisha Zamal, Khan Mohammad Habibullah, Amit Kumar Das: Predicting Crime Using Time and Location Data(2019). <https://www.researchgate.net/publication/335854157_Predicting_Crime> Using Time and Location Data.

4. Peng Chen, Justin Kurland, Modus Operandi: Time, Place, A Simple Apriori Algorithm Experiment for Crime Pattern Detection (2018).9th International Conference on IISA. <https://www.researchgate.net/publication/330877862_Time_Place_and_Modus_Operandi_A_Simple_Apriori_Algorithm> Experiment for Crime Pattern Detection.

5. Jyoti Agarwal, Renuka Nagpal, Rajni Sehgal: Crime Analysis using K-Means Clustering (2013). <https://www.researchgate.net/publication/269667894> Crime Analysis using K-Means Clustering

6. Malathi. A & Dr. S. Santhosh Baboo: An Enhanced Algorithm to Predict a Future Crime using Data Mining. International Journal of Computer Applications (0975 – 8887) Volume 21– No.1, May 2011

7. Khushabu A. Bokde,Tiksha P. Kakade, Dnyaneshwari S. Tumsare, ChetanG.Wadhai: Crime Analysis Using K-Means Clustering(2018).https://www.ijert.org/crime-analysis-using-k-means-clustering.

8. Mrs.S. Sujatha, Mrs. A. Shanthi Sona: New Fast K-Means Clustering Algorithm using Modified Centroid Selection Method (2013).

9. Keerthi A, Remya M S, Nitha L: Detection of Credit Card Frauds Using Hidden Markov Model with Improved K-Means Clustering Algorithm (2015). https://www.ijarcs/article/download/1239/1227. [7] InformedHealth.org, "Cervical cancer: Overview," 14 Dec 2012. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK279259/. [Accessed Jan 2019].

[8] Cancer Report. February 2017 [cited WHO World Health Organization 25.09.2017]; Available from: http://www.who.int/mediacentre/factsheets/fs297/en/.

[9] Fernandes, K., J.S. Cardoso, and J. Fernandes, Transfer Learning with Partial Observability Applied to Cervical Cancer Screening, in Pattern Recognition and Image Analysis: 8th Iberian Conference, IbPRIA 2017, Faro, Portugal, June 20-23, 2017, Proceedings, L.A. Alexandre, J. Salvador Sánchez, and J.M.F. Rodrigues, Editors. 2017, Springer International Publishing: Cham. p. 243-250.

[10] Sen, T. and S. Das, An Approach to Pancreatic Cancer Detection using Artificial Neural Network, in Proc. of the Second Intl. Conf. on Advances in Computer, Electronics and Electrical Engineering (CEEE2013). 2013. p. 56-60.

[11] Fernandes, K., J.S. Cardoso, and J. Fernandes, Transfer Learning with Partial Observability Applied to Cervical Cancer Screening, in Iberian Conference on Pattern Recognition and Image Analysis. 2017: Faro, Portugal. p. 61-69.