**Enterprise Artificial Intelligence  
Project: Predicting churn**

For Hackveda, VMDD Technologies Using Python

*Submitted in partial fulfillment of the requirements*

*for the award of the degree of*

**Master of Computer Application (MCA)**

To

Guru Gobind Singh Indraprastha University, Delhi

Guide: Submitted by:

Ms. Ruby Dahiya Manoj Kumar

**** Roll No: 00213724417

**Institute of Information Technology & Management,**

**New Delhi – 110058**

**Batch (2016-2019)**

**Certificate**

I Mr. **Manoj Kumar**, Roll No. **00213724417** certify that the Project Report/Dissertation

(Paper Code MCA-302) entitled “**Enterprise Artificial Intelligence Project: Predicting**

**Churn for Hackveda, VMDD Technologies Using Python”** is done by me and it is an

authentic work carried out by me at **Hackveda, VMDD Technologies, Rohini**. The matter

embodied in this project work has not been submitted earlier for the award of any degree or

diploma to the best of my knowledge and belief.

Signature of the Student

Date:

Certified that the Project Report/Dissertation (MCA-302) entitled “**Enterprise Artificial**

**Intelligence Project: Predicting Churn for Hackveda, VMDD Technologies Using**

**Python”** done by Mr. **Manoj Kumar**, Roll No**. 00213724417**, is completed under my

guidance.

Signature of the Guide

Date:

Name of the Guide:

Designation:

Address:

Countersigned Institute of Information Technology

&Management, New Delhi-110058

Director/HOD

**Acknowledgement**

This is a great opportunity to acknowledge the remarkable contribution of all the persons whose

support makes this project “**Enterprise Artificial Intelligence Project: Predicting Churn**” a

reality. I would like to add a few heartfelt words for the people who were part of this project in

numerous ways.

It is my privilege to express my deep sense of gratitude to my guide **Ms.** **Ruby Dahiya**,

**Associate Professor**, Institute of Information Technology & Management, New Delhi**,** for her exemplary guidance, monitoring and encouragement throughout the course of this project. The blessing, benevolent discipline and guidance given by his time to time shall carry me a long way in the journey of life on which I am about to embark.

I am also thankful to **Mr. Devanshu Shukhla**, Project Manager at VMDD Technologies, Hackveda, Rohini for his constant guidance in directing many relevant technical details about the project.

My sincere gratitude to **Dr. Prerna Mahajan, Director**, Institute of Information Technology & Management, New Delhi for her creative and strict guidance throughout the course of study. I am also deeply indebted to all teaching and non-teaching staffs of my Institute for their continuous support and motivation.

Last but not the least, I would like to acknowledge the constant support of my parents and family members, friends whose patience and encouragement during project completion helped in making this project a reality.

**Manoj Kumar**

**00213724417**

**MCA-6th Semester**

**EXECUTIVE SUMMARY**

**Title of the Project:** The title of the project is “Enterprise Artificial Intelligence Project: Predicting churn for Hackveda, VMDD Technologies using Python and Anaconda navigator”.

**Organization:**

The name of the organization where the project is intended is Hackveda, VMDD Technologies. Hackveda, VMDD Technologies is an ISO 9001:2008, approved organization for Software Development. VMDD Technologies was established in 2008 by Mr. V.K.Shukla, RETD. Indian Air Force Officer. During this period VMDD Technologies has successfully delivered product and services to Defense Research Development Organization, Ministry Of Defense (India), Indian Institute of Technology, National Physical Laboratory and various other Govt. and private organization.

VMDD Technologies is providing software product & services since 2011. VMDD started its cyber security & Training Service in 2011. Later Hackveda, VMDD Technologies Introduced android application development & training Services, commercial android applications development etc. Hackveda is also a Creative Learning, Certification, Development, Publishing and Research Center.

**Problem Definition:**

In this era of technologies, the telecom industry continues to face growing pricing pressure worldwide. While regional difference apply, wireless penetration is reaching a saturation point across multiple markets. In addition, the longstanding ability to differentiate products and services based on handset selection and network quality is disappearing, and product lifecycles are shortening. Simultaneously, wireline businesses are facing increasing competition from cable operators and a rising risk of [disruption from OTT players](https://www.mckinsey.com/industries/telecommunications/our-insights/overwhelming-ott-telcos-growth-strategy-in-a-digital-world). All of these powerful trends are forcing telecom companies to respond through more competitive offers, bundles, and price cuts.

Given these challenging industry dynamics, managing the customer base to reduce churn should be among any senior telecom executive’s highest priorities.

**REQUIREMENTS OF NEW SYSTEM**

The problems of the Churn in telecom industry required to develop a comprehensive view of the customer and link that view directly to results. Leading operators are structured and thorough in linking and aggregating disparate data sets to develop a full view of the customer [over the entire decision journey](https://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/the-new-consumer-decision-journey)—from acquisition and onboarding to upgrade cycles and eventual disconnect, if applicable.

Use cutting-edge analytical techniques. Cutting-edge analytics let operators apply advanced algorithms to vast troves of data without needing to program specific transformations.

These algorithms can identify previously hidden variables and combinations of variables that predict customer behaviors such as churn. Companies can then analyze the reasons behind those behaviors to come up with solutions. Break the customer base into scores of microsegments. The full value of data analytics can only be realized when companies can [personalize the treatment of a precisely targeted group of customers](https://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/the-heartbeat-of-modern-marketing) with the highest propensity to leave. Such a tailored approach requires a granular micro-segmentation of the customer base which is then matched to a broad, well-classified library of offers.

**Objective of the project:**

This project will help us to learn how to combine several models to build a churn prevention pipeline: segment customers, create insights to understand churn, and build a model to score new customers.

The data is from a major telecom operator. Just like pretty much any company in the world, they’re concerned with keeping our customers happy, so they won’t leave. In other words, they’re looking at ways to reduce churn. To do this, they set up a task force of data analysis and people from business teams who came up with several business goals to reduce churn.

* Get to know customer better, by accessing the data about their plans and usage, and getting in touch with interesting profiles
* Target clients with more effective advertising based on their usage profiles
* Retrieve customers with very high likeliness of churn so we could get in touch and offer them special deals before they even thought of leaving

**Methodology**:

1. In order to find a possible solution to the problem of
2. churn prediction i.e. successfully apply a machine learning
3. technique to the available data, one needs a deep
4. In order to find a possible solution to the problem of
5. churn prediction i.e. successfully apply a machine learning
6. technique to the available data, one needs a deep
7. In order to find a possible solution to the problem of
8. churn prediction i.e. successfully apply a machine learning
9. technique to the available data, one needs a deep
10. In order to find a possible solution to the problem of
11. churn prediction i.e. successfully apply a machine learning
12. technique to the available data, one needs a deep
13. In order to find a possible solution to the problem of
14. churn prediction i.e. successfully apply a machine learning
15. technique to the available data, one needs a deep
16. In order to find a possible solution to the problem of
17. churn prediction i.e. successfully apply a machine learning
18. technique to the available data, one needs a deep

We will be working on historic data from users on their phone usage, as well as various features from very large log files. The clients who have churned are indicated in the dataset.

We have another dataset with the same features built on current clients. That will be used to deploy the model and predict who is likely to churn. Instead of just answering the yes, no question: “will they churn,” we have decided to build models instead of one:

1. A **first model that segments customers** into relevant groups (by using Clustering algorithms), for targeting.
2. A **second model that uses these segments (clusters) to predict the churn likeliness** of each unlabeled customer (by using classification algorithms), so that business units can then check scores on a daily basis and target these customers.

**Tools:** The Hardware and Software Requirement Specifications: -

|  |  |
| --- | --- |
| Minimum Requirements | |
| Computer (CPU) | Intel i3 or Higher |
| Memory | 4GB |
| Hard Disk | 10 GB |
| Display | 1366 768 |
| Input Device | Keyboard, Mouse |
| Internet connectivity | Ethernet or Wi-fi |
| Web Browser | Google Chrome or Microsoft Edge |

**Table No-1: Hardware Requirements**

|  |  |
| --- | --- |
| Minimum Requirements | |
| Front End | Python |
| Operating System | Windows 10 |
| Development Tool | Anaconda navigator (IDE), Jupyter Notebook |

**Table No-2: Software Requirements**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S No** | **Topic** | **Page No** |
| 1 | Certificate | 2 |
| 2 | Acknowledgement | 3 |
| 3 | Executive Summary | 4-7 |
| 4 | List of Tables | 9 |
| 5 | List of Figures | 10-11 |
| 6 | Chapter-1: Introduction | 12-26 |
| 7 | Chapter-2: Exploratory Data Analysis | 27-32 |
| 8 | Chapter-3: Data Preparation | 33-42 |
| 9 | Chapter-4: Data visualization | 43-60 |
| 10 | Chapter-5: Data Modeling | 61-76 |
| 11 | Chapter-6: Summary and Conclusions | 77-82 |
| 12 | References | 83 |
| 13 | Appendix | 84-96 |

**LIST OF TABLES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S No** | **Table No** | **Title** | **Page No** |
| 1 | 1.1 | Hardware Requirements | 20 |
| 2 | 1.2 | Software Requirements | 20 |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S No** | **Figure.no** | **Title** | **Page No** |
| 1 | 1.1 | Gantt Chart | 22 |
| 2 | 1.2 | Processes | 22 |
| 3 | 1.3 | Knowledge discovery Process | 23 |
| 4 | 2.1 | Dataset | 29 |
| 5 | 2.2 | Shape of Dataset | 29 |
| 6 | 2.3 | Description of Dataset | 31 |
| 7 | 3.1 | Data Preparation Process | 34 |
| 8 | 3.2 | Dataset Information | 37 |
| 9 | 3.3 | Total Charges changed Information of Dataset | 38 |
| 10 | 3.4 | Handling Missing Values | 40 |
| 11 | 3.5 | Manipulation of Data | 42 |
| 12 | 4.1 | Percentage of Churn Customer in Dataset | 46 |
| 13 | 4.2 | Distribution of Customer Attrition | 46 |
| 14 | 4.3 | KDE Plot for Tenure | 47 |
| 15 | 4.4 | KDE Plot for Monthly Charges | 48 |
| 16 | 4.5 | KDE Plot for Total Charges | 48 |
| 17 | 4.6 | KDE Plot for Difference in Monthly and Total Charges | 49 |
| 18 | 4.7 | Senior Citizen | 50 |
| 19 | 4.8 | Senior Citizen on the basis of gender | 50 |
| 20 | 4.9 | Customer Has Partner or Dependents | 51 |
| 21 | 4.10 | Customer Has Multiple Lines | 52 |
| 22 | 4.11 | Monthly Charges of Customer Has Multiple Lines | 52 |
| 23 | 4.12 | Customer Has Internet Service | 53 |
| 24 | 4.13 | Monthly Charges of Customer Has Internet Service | 54 |
| 25 | 4.14 | Total Customer Has Additional Service | 54 |
| 26 | 4.15 | Churn Customer Has Additional Service | 55 |
| 27 | 4.16 | Contract | 56 |
| 28 | 4.17 | Payment Method | 56 |
| 29 | 4.18 | Monthly Charges on Contract Basis | 57 |
| 30 | 4.19 | Monthly Charges on Payment Method Basis | 58 |
| 31 | 4.20 | Correlation Matrix | 60 |
| 32 | 5.1 | A visualization of the splits | 63 |
| 33 | 5.2 | Analyzing Multiple Iteration For KNN | 66 |
| 34 | 5.3 | Analyzing Multiple Iteration for Random Forest | 71 |
| 35 | 5.4 | Confusion Matrix Structure | 72 |
| 36 | 5.5 | Confusion Matrix | 73 |
| 37 | 6.1 | Features affecting Customer churn | 82 |

**Chapter 1**

**INTRODUCTION**

* 1. **Introduction**

In this chapter various initial aspects of the projects are covered such as the general description about the company, project, methodology used, needs for the new system, data collection method and project planning demonstration.

* 1. **Brief Description About Organization**

The name of the organization where the project is intended is Hackveda, VMDD Technologies. Hackveda, VMDD Technologies is an ISO 9001:2008, approved organization for Software Development. VMDD Technologies was established in 2008 by Mr. V.K.Shukla, RETD. Indian Air Force Officer. During this period VMDD Technologies has successfully delivered product and services to Defense Research Development Organization, Ministry of Defense (India), Indian Institute of Technology, National Physical Laboratory and various other Govt. and private organization.

VMDD Technologies is providing software product & services since 2011. VMDD started its cyber security & Training Service in 2011. Later Hackveda, VMDD Technologies Introduced android application development & training Services, commercial android applications development etc. Hackveda is also a Creative Learning, Certification, Development, Publishing and Research Center.

**1.3 General Description of the System under Study**

The **Enterprise Artificial Intelligence Project: Predicting Churn** is a python notebook Made and Developed in Jupyter Notebook only designed to provide user efficient working environment. This Notebook provides user friendly interface resulting in knowing each and every usability features of the Dataset. This system completes the work in a very less time resulting in less time consumption and high level of efficiency. This system is developed in such a way that even a naïve user can also operate the system easily. The calculations are made very quickly and the records are generated and can be used again for a longer period of time.

**1.4 User Requirements**

**1.4.1** **Problems In existing system:** As we know manual system are quite tedious, time consuming and less efficient and accurate in comparison to the computerized system.

So, following are some disadvantages of the old system:

1. Time consuming
2. Less accurate
3. Less efficient
4. Lot of paper work
5. Slow data processing
6. Not user-friendly environment
7. Difficult to keep old records

**1.4.2** **Need of the New System**:Without new system user requirements can’t be fulfilled. Following are the user’s requirements which old system don’t hold and therefore there is a need to develop a new efficient system: -

1. Deep Analysis
2. Showing great result with the help of Graph and plots
3. Fast Processing
4. Faster and accurate generation of effective Knowledge
5. Decision making

**1.4.3** **Proposed system advantages:** Following are the characteristics of proposed system-

(a) Efficiency: Provides appropriate interactions with the resources, data manipulation, Exploratory Data Analysis, Data Preparation, Data visualization, Model Building, Model Performance.

(b) Effectiveness: The effectiveness of proposed system is measured in producing a desired result.

The scope of this project is to provide user efficient working environment and more output can be generated through this. This project provides user friendly interface resulting in knowing each and every usability features of the project. This project completes the work in a very less time resulting in less time consumption and high level of efficiency. This project is developed in such a way that even a naïve user can also operate the model easily. The calculations are made very quickly and the records are generated and can be used again for a longer period of time

**1.5 Objective of the project:**

This project will help us to learn how to combine several models to build a churn prevention pipeline: segment customers, create insights to understand churn, and build a model to score new customers. The data is from a major telecom operator. Just like pretty much any company in the world, they’re concerned with keeping our customers happy, so they won’t leave. In other words, they’re looking at ways to reduce churn. To do this, they set up a task force of data analysis and people from business teams who came up with several business goals to reduce churn.

* Get to know customer better, by accessing the data about their plans and usage, and getting in touch with interesting profiles
* Target clients with more effective advertising based on their usage profiles
* Retrieve customers with very high likeliness of churn so we could get in touch and offer them special deals before they even thought of leaving

The only necessary condition for the usage of this Project is that there should be an Analysts who can Extract the necessary insights. He has the ability to manipulate, transform, and get the proper and meaningful Information from the dataset.

**1.6 Methodology:**

The methodology basically being followed as for the Data science and Machine learning, In order to find a possible solution to the problem of Customer attrition i.e. successfully apply a machine learning technique to the available data, one needs a deep understanding of the business rules of the telecommunications company and their specificity. Such knowledge enables the selection of attributes suitable for the problem at hand. The quality of the data can further be improved by subjecting it to preprocessing. Once a final dataset is derived, the classification algorithms can be successfully trained and their performances correspondingly evaluated. In the following subsections, we present the identified phases in our methodology.

**1.6.1** **Business Understanding**:

In this initial phase, the focus is set on understanding the project objectives and requirements from the telecommunications business perspective. The aim of the churn prediction is to identify the properties that make a customer churn in order to prevent it and retain the customer. To enable this, we consider customers that churned and analyze their data over a period while they still used the services of the telecommunications company.

**1.6.2** **Data Understanding**

For the purpose of this paper, a telecommunication company from Macedonia shared their data, through text files exported from certain tables in their database. We have anonymized the data (we only care about the user’s dynamics data, not their personal data). The obtained data covers 28 months period from 01.01.2012 to 30.04.2014 (approximately 34 million records). Additional data for the customer complaints is included in the dataset since it is a strong indicator for customer dissatisfaction.

**1.6.3** **Data Pre-processing**

The data pre-processing tasks include careful selection of data attributes and records. Because we deal with incomplete and noisy data, some additional data cleaning and transformation are also performed.

**1) Data Selection**

We first identify and extract the most relevant attributes for the research. The initial dataset consists of 68 (mainly numeric) attributes, that can be grouped in the following three categories:

**• Demographic attributes**: contain the primary features of the customer such as sex, age, nationality, place of residence, etc.

• **Contract attributes**: contain the attributes associated with the customer contract for a particular service such as type of service, date of conclusion of the contract, price of the service etc.

• **Customer behavior attributes**: describe the customer activities. The data subjected to our analysis spans over a period of one year from 01.05.2013 to 30.04.2014. A total of 22461 customers are included, of which 2629 customers are churns, while 19832 customers still use the services of the telecommunication company.

2) **Data Cleaning**

The presence of noise, unknown or empty values, outliers and invalid values may negatively affect the performance of the machine learning algorithm by using the raw data. The purpose of data cleaning is to reduce the number of inconsistent values, remove noise and incomplete entries and attributes. Since our dataset is sufficiently big, we removed all potentially problematic tuples.

3**) Data Transformation**

Data transformation techniques can significantly improve the overall performance of the churn prediction, which we have seen while experimenting with potential transformations. The prediction produces best results when data attributes are normalized (in the [0,1] range) and discretized (Sturges and k-proportional were used) and the results presented in this paper refer to such data.

4**) Feature Selection**

Features selection refers to the process of selecting a subset of relevant attributes of a set of attributes. This reduces the number of input attributes to the learning algorithm, thereby significantly reducing time and resources required to train the algorithm. For feature selection, we have chosen two techniques chi-squared, Information gain.

**1.6.4** **Machine learning approaches for churn prediction**

There are many techniques that have been proposed for customer churn prediction. In our approach, we will analyze four machine learning algorithms: Decision tree, k-nearest neighbors’ algorithm, naïve Bayes classifier and logistics regression.

**1) Decision Tree:**

A decision tree is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements. Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label.

**2) K-nearest neighbors:**

K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection. It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as [GMM](https://en.wikipedia.org/wiki/Mixture_model), which assume a Gaussian distribution of the given data). We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute.

**3) Naïve Bayes:**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), naive Bayes classifiers are a family of simple "[probabilistic classifiers](https://en.wikipedia.org/wiki/Probabilistic_classifier)" based on applying [Bayes' theorem](https://en.wikipedia.org/wiki/Bayes%27_theorem) with strong (naive) [independence](https://en.wikipedia.org/wiki/Statistical_independence) assumptions between the features. Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of [feature](https://en.wikipedia.org/wiki/Feature_vector) values, where the class labels are drawn from some finite set. There is not a single [algorithm](https://en.wikipedia.org/wiki/Algorithm) for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is [independent](https://en.wikipedia.org/wiki/Independence_(probability_theory)) of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible [correlations](https://en.wikipedia.org/wiki/Correlation_and_dependence) between the color, roundness, and diameter features. For some types of probability models, naive Bayes classifiers can be trained very efficiently in a [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) setting.

**4) Logistics Regression:**

In [statistics](https://en.wikipedia.org/wiki/Statistics), the logistic model (or logit model) is a widely used [statistical model](https://en.wikipedia.org/wiki/Statistical_model) that in its basic form uses a [logistic function](https://en.wikipedia.org/wiki/Logistic_function) to model a [binary](https://en.wikipedia.org/wiki/Binary_variable) [dependent variable](https://en.wikipedia.org/wiki/Dependent_variable), although many more complex [extensions](https://en.wikipedia.org/wiki/Logistic_regression#Extensions) exist. In [regression analysis](https://en.wikipedia.org/wiki/Regression_analysis), logistic regression (or logit regression) is [estimating](https://en.wikipedia.org/wiki/Estimation_theory) the parameters of a logistic model (a form of [binary regression](https://en.wikipedia.org/wiki/Binary_regression)). Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail, win/lose, alive/dead or healthy/sick; these are represented by an [indicator variable](https://en.wikipedia.org/wiki/Indicator_variable), where the two values are labeled "0" and "1.

**5) SVM (Support Vector machine):**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), support-vector machines (SVMs, also support-vector networks) are [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) models with associated learning [algorithms](https://en.wikipedia.org/wiki/Algorithm) that analyze data used for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression analysis](https://en.wikipedia.org/wiki/Regression_analysis). Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-[probabilistic](https://en.wikipedia.org/wiki/Probabilistic_classification) [binary](https://en.wikipedia.org/wiki/Binary_classifier) [linear classifier](https://en.wikipedia.org/wiki/Linear_classifier) (although methods such as [Platt scaling](https://en.wikipedia.org/wiki/Platt_scaling) exist to use SVM in a probabilistic classification setting).

**6) Random Forest** **:** Random forests or random decision forests are an [ensemble learning](https://en.wikipedia.org/wiki/Ensemble_learning) method for [classification](https://en.wikipedia.org/wiki/Statistical_classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis) and other tasks that operates by constructing a multitude of [decision trees](https://en.wikipedia.org/wiki/Decision_tree_learning) at training time and outputting the class that is the [mode](https://en.wikipedia.org/wiki/Mode_(statistics)) of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of [overfitting](https://en.wikipedia.org/wiki/Overfitting) to their [training set](https://en.wikipedia.org/wiki/Test_set).

**1.62 Tools:** The Hardware and Software Requirement Specifications: -

|  |  |
| --- | --- |
| Minimum Requirements | |
| Computer (CPU) | Intel i3 or Higher |
| Memory | 4GB |
| Hard Disk | 10 GB |
| Display | 1366 768 |
| Input Device | Keyboard, Mouse |
| Internet connectivity | Ethernet or Wi-fi |
| Web Browser | Google Chrome or Microsoft Edge |

**Table No-1.1: Hardware Requirements**

|  |  |
| --- | --- |
| Minimum Requirements | |
| Front End | Python |
| Back End | Php MySQL |
| Operating System | Windows |
| Development Tool | Anaconda navigator (IDE) |

**Table No-1.2: Software Requirements**

**1.7 Data required & Data Collection Method**

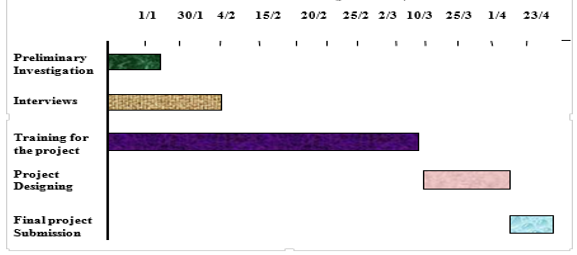
The process of gathering the data from the user is accomplished through following processes: -

**(a) Observation Method:** Observation method is a method under which data from the field is collected with the help of observation by the observer or by personally going to the field.  
**(b)** **Interview Method:** This method of collecting data involves presentation or oral-verbal stimuli and reply in terms of oral-verbal responses. This method is of Oral Verbal communication. Where interviewer asks questions (which are aimed to get information required for study) to respondent

**(c)** **Questionnaire Method:** This method of data collection is quite popular, particularly in case of big enquiries. The questionnaire is mailed to respondents who are expected to read and understand the questions and write down the reply in the space meant for the purpose in the questionnaire itself. The respondents have to answer the questions on their own.

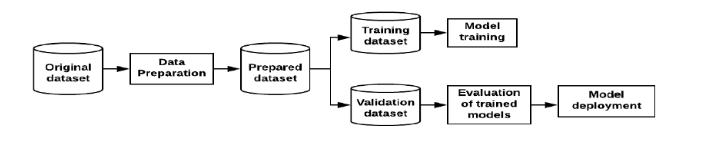
**1.8 Project Planning:**

1.81 When creating a software project schedule, the planner begins with a set of tasks (the work breaks down structure). Efforts, duration, and start date are then input for each task. In addition, task may be assigned to specific individuals. As a sequence of this input, a timeline chart, also called a Gantt chart, is generated. A Gantt chart can be developed for the entire project. Alternatively, separated it depicts a part of a software project schedule that emphasizes the concept scooping task for a new word processing software project. All project tasks are listed in the left-hand column. The horizontal bars occur at the same time on the calendar, task concurrency is implied.



**Figure No- 1.1: Gantt Chart**

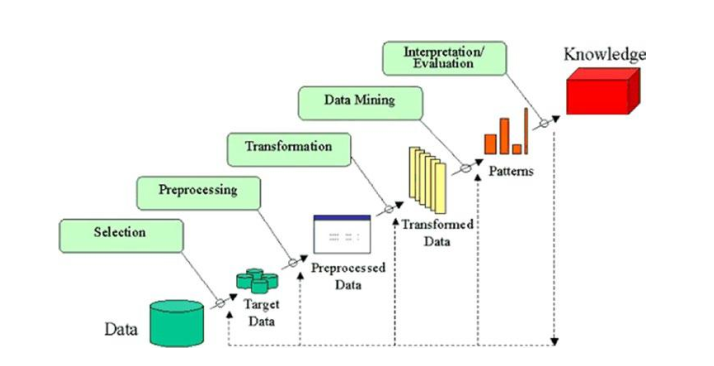
**1.9** **Identification of the Project Process**



**Figure No- 1.2: Processes**

**1.10 Flow diagram**

Flow diagram or we can say Flow Chart is graphical representation of flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data. The FD mention about how data flows through the system.There is a prominent difference between DFD and Flowchart. The flowchart depicts flow of control in program modules. DFDs depict flow of data in the system at various levels. DFD does not contain any control or branch elements.



**Figure 1.3 Knowledge discovery Process**

Some people don’t differentiate data mining from knowledge discovery while others view data mining as an essential step in the process of knowledge discovery. Here is the list of steps involved in the knowledge discovery process –

* **Data Cleaning** − In this step, the noise and inconsistent data is removed. The presence of noise, unknown or empty values, outliers and invalid values may negatively affect the performance of the machine learning algorithm by using the raw data. The purpose of data cleaning is to reduce the number of inconsistent values, remove noise and incomplete entries and attributes. Since our dataset is sufficiently big, we removed all potentially problematic tuples. Data cleansing or data cleaning is the process of detecting and correcting corrupt or inaccurate records from a record set, table, or database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data.
* **Data Integration** − In this step, multiple data sources are combined.Data integration involves combining [data](https://en.wikipedia.org/wiki/Data) residing in different sources and providing users with a unified view of them. This process becomes significant in a variety of situations, which include both commercial (such as when two similar companies need to merge their [databases](https://en.wikipedia.org/wiki/Database)) and scientific (combining research results from different [bioinformatics](https://en.wikipedia.org/wiki/Bioinformatics) repositories, for example) domains. Data integration appears with increasing frequency as the volume (that is, [big data](https://en.wikipedia.org/wiki/Big_data)) and the need to share existing data [explodes](https://en.wikipedia.org/wiki/Information_explosion). It has become the focus of extensive theoretical work, and numerous open problems remain unsolved. Data integration encourages collaboration between internal as well as external users
* **Data Selection** − In this step, data relevant to the analysis task are retrieved from the database.Data selection is defined as the process of determining the appropriate data type and source, as well as suitable instruments to collect data. Data selection precedes the actual practice of data collection. This definition distinguishes data selection from selective data reporting (selectively excluding data that is not supportive of a research hypothesis) and interactive/active data selection (using collected data for monitoring activities/events, or conducting secondary data analyses). The process of selecting suitable data for a research project can impact data integrity. The primary objective of data selection is the determination of appropriate data type, source, and instrument(s) that allow investigators to adequately answer research questions. This determination is often discipline-specific and is primarily driven by the nature of the investigation, existing literature, and accessibility to necessary data sources. Integrity issues can arise when the decisions to select ‘appropriate’ data to collect are based primarily on cost and convenience considerations rather than the ability of data to adequately answer research questions. Certainly, cost and convenience are valid factors in the decision-making process. However, researchers should assess to what degree these factors might compromises the integrity of the research endeavor.
* **Data Transformation** − In this step, data is transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations. In computing, Data transformation is the process of converting data from one format or structure into another format or structure. It is a fundamental aspect of most [data integration](https://en.wikipedia.org/wiki/Data_integration) and [data management](https://en.wikipedia.org/wiki/Data_management) tasks such as [data wrangling](https://en.wikipedia.org/wiki/Data_wrangling), [data warehousing](https://en.wikipedia.org/wiki/Data_warehousing), [data integration](https://en.wikipedia.org/wiki/Data_integration) and application integration. Data transformation can be simple or complex based on the required changes to the data between the source (initial) data and the target (final) data. Data transformation is typically performed via a mixture of manual and automated steps. Tools and technologies used for data transformation can vary widely based on the format, structure, complexity, and volume of the data being transformed. A [master data](https://en.wikipedia.org/wiki/Master_data) recast is another form of data transformation where the entire [database](https://en.wikipedia.org/wiki/Database) of data values is transformed or recast without extracting the data from the database.
* **Data Mining** − In this step, intelligent methods are applied in order to extract data patterns.Data mining is the process of discovering patterns in large [data sets](https://en.wikipedia.org/wiki/Data_set) involving methods at the intersection of [machine learning](https://en.wikipedia.org/wiki/Machine_learning), [statistics](https://en.wikipedia.org/wiki/Statistics), and [database systems](https://en.wikipedia.org/wiki/Database_system). Data mining is an [interdisciplinary](https://en.wikipedia.org/wiki/Interdisciplinary) subfield of [computer science](https://en.wikipedia.org/wiki/Computer_science) and [statistics](https://en.wikipedia.org/wiki/Statistics) with an overall goal to extract information (with intelligent methods) from a data set and transform the information into a comprehensible structure for further use.  Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD. Aside from the raw analysis step, it also involves database and [data management](https://en.wikipedia.org/wiki/Data_management) aspects, [datapreprocessing](https://en.wikipedia.org/wiki/Data_pre-processing), [model](https://en.wikipedia.org/wiki/Statistical_model) and [inference](https://en.wikipedia.org/wiki/Statistical_inference) considerations, interesting isometrics, [complexity](https://en.wikipedia.org/wiki/Computational_complexity_theory) considerations, post-processing of discovered structures, [visualization](https://en.wikipedia.org/wiki/Data_visualization), and [online updating](https://en.wikipedia.org/wiki/Online_algorithm).

**Pattern Evaluation** − In this step, data patterns are evaluated.  Pattern Evaluation is defined as identifying strictly increasing patterns representing knowledge based on given measures. Find interestingness score of each pattern. Uses summarization and Visualization to make data understandable by user. Generally, this step includes visualization, transformation, removing redundant patterns from the patterns we generated.

* **Knowledge Presentation** − In this step, knowledge is represented. Knowledge representation is defined as technique which utilizes visualization tools to represent data mining results. Generate reports, generate tables, Generate discriminant rules, classification rules, characterization rules, etc. As this step is beneficial to us. Also, it helps to use the knowledge acquired to take better decisions.

**Chapter 2**

**Exploratory Data Analysis**

2.1 Introduction

In [statistics](https://en.wikipedia.org/wiki/Statistics), exploratory data analysis (EDA) is an approach to analyzing [data sets](https://en.wikipedia.org/wiki/Data_set) to summarize their main characteristics, often with visual methods. A [statistical model](https://en.wikipedia.org/wiki/Statistical_model) can be used or not, but primarily EDA is for seeing what the data can tell us beyond the formal modelling or hypothesis testing task. Exploratory data analysis was promoted by [John Tukey](https://en.wikipedia.org/wiki/John_Tukey) to encourage statisticians to explore the data, and possibly formulate hypotheses that could lead to new data collection and experiments. EDA is different from [initial data analysis (IDA)](https://en.wikipedia.org/wiki/Data_analysis#Initial_data_analysis). which focuses more narrowly on checking assumptions required for model fitting and hypothesis testing, and handling missing values and making transformations of variables as needed. EDA encompasses IDA.

**For Example,** imagine your wolf pack decides to watch a movie you haven’t heard of. There is

absolutely no debate about that, it will lead to a state where you find yourself puzzled with lot of

questions which needs to be answered in order to make a decision. Being a good chieftain the first

question you would ask, what is the cast and crew of the movie? As a regular practice, you would

also watch the trailer of the movie on YouTube. Furthermore, you’d find out ratings and reviews

the movie has received from the audience. Whatever investigating measures you would take

before finally buying popcorn for your clan in theater, is nothing but what data scientists in their

lingo call ‘Exploratory Data Analysis’.

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations. So, to explain further I am going to import my dataset.

**2.2 Importing Dataset**

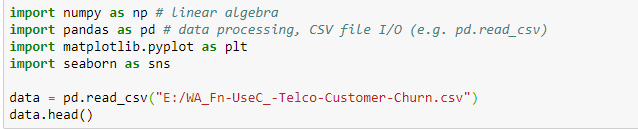
To starts with, I imported necessary libraries (for this example pandas, NumPy, matplotlib and seaborn) and loaded the data set.

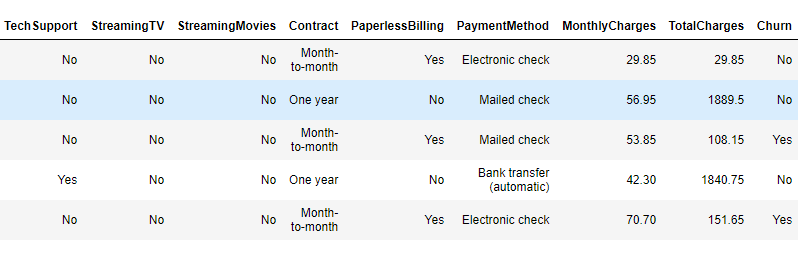
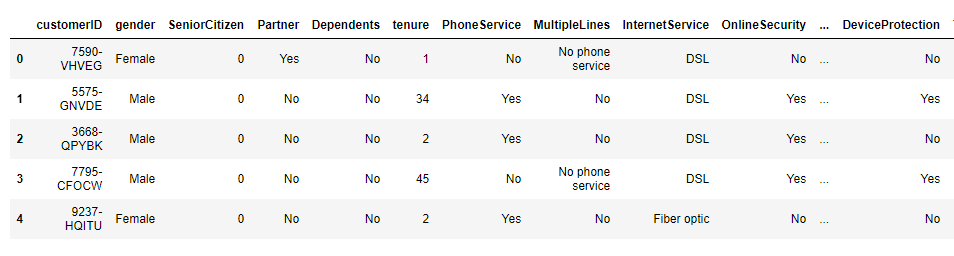
**2.2.1 Dataset**

Customer retention can be achieved with good customer service and products. But the most effective way for a company to prevent attrition of customers is to truly know them. The vast volumes of data collected about customers can be used to build churn prediction models. Knowing who is most likely to defect means that a company can priorities focused marketing effort on that subset of their customer base.

Preventing customer churn is critically important to the telecommunications sector, as the barriers to entry for switching services are so low. In this post, we will examine customer data from IBM Sample Data Sets with the aim of building and comparing several customer churn prediction models. You can download the dataset [here.](https://community.watsonanalytics.com/wp-content/uploads/2015/03/WA_Fn-UseC_-Telco-Customer-Churn.csv)

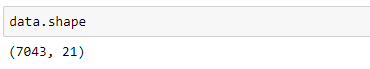
<https://community.watsonanalytics.com/wp-content/uploads/2015/03/WA_Fn-UseC_-Telco-Customer-Churn.csv>





**Figure: 2.1 Dataset**

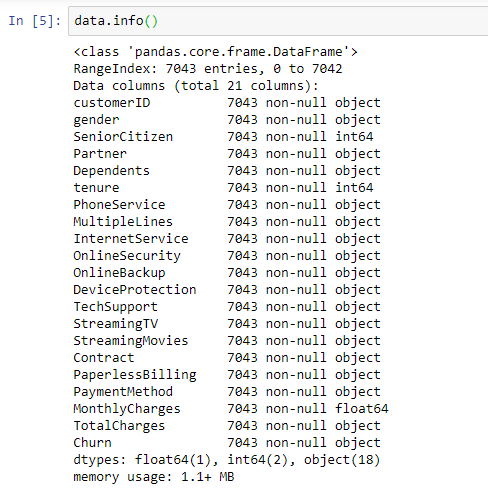
* In the First line we just imported the dataset from the directory.
* To take a closer look at the data took help of “. head ()” function of pandas library which returns first five observations of the data set. Similarly, “. tail ()” returns last five observations of the data set.

I found out the total number of rows and columns in the data set using “. shape”

**Figure: 2.2 Shape of Dataset**

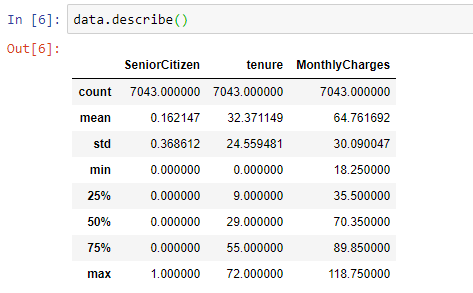
* Dataset comprises of 7043 observations and 21 characteristics.
* Out of which one is dependent variable and rest 20 are independent variables

It is also a good practice to know the columns and their corresponding data types, along with finding whether they contain null values or not.



* Data has some float and integer values and also contains object values But Pandas didn’t detect all of the values in the ‘TotalCharges’ column to be float64 type, so we probably have some non-numeric data in the column.
* ‘TotalCharges ‘column has some null/missing values. We are going to find that below in this report

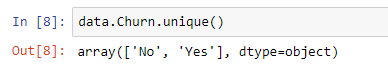
The describe () function in pandas is very handy in getting various summary statistics. This function returns the count, mean, standard deviation, minimum and maximum values and the quantiles of the data.



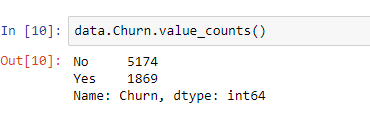
**Figure: 2.3 Description of Dataset**

* Here as you can notice mean value is less than median value of Monthly Charges column which is represented by 50% (50th percentile) in index column.
* There is notably a large difference between 75th %tile and max values of each column.
* Thus observations 1 and 2 suggests that there are extreme values-Outliers in our data set.

Few key insights just by looking at dependent variable are as follows:



* Target variable/Dependent variable is discrete and categorical in nature.
* Target variable only have yes and no so we can classify the rest of the data based in the dependent variable.



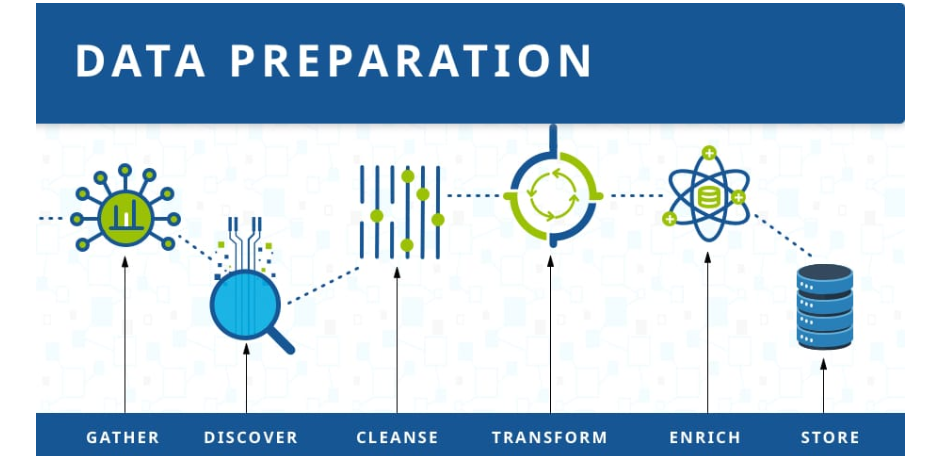
* This tells us vote count of Churn user score in descending order.
* “Churn” has most values concentrated in non-churn user this indicates that in this dataset the non-churn users are more than churn users.

**Chapter 3**

**Data Preparation**

**3.1 Introduction**

Data preparation is the act of manipulating (or pre-processing) [raw data](https://en.wikipedia.org/wiki/Raw_data) (which may come from disparate data sources) into a form that can readily and accurately analyzed, e.g. for business purposes. Data preparation is the first step in data analytics projects and can include many discrete tasks such as loading data or data ingestion, [data fusion](https://en.wikipedia.org/wiki/Data_fusion), [data cleaning](https://en.wikipedia.org/wiki/Data_cleaning), [data augmentation](https://en.wikipedia.org/wiki/Data_augmentation), and data delivery. Good data preparation allows for efficient analysis, limits errors and inaccuracies that can occur to data during processing, and makes all processed data more accessible to users. It’s also gotten easier with new tools that enable any user to cleanse and qualify data on their own. Data preparation is the process of cleaning and transforming raw data prior to processing and analysis. It is an important step prior to processing and often involves reformatting data, making corrections to data and the combining of data sets to enrich data. Data preparation is often a lengthy undertaking for data professionals or business users, but it is essential as a prerequisite to put data in context in order to turn it into insights and eliminate bias resulting from poor data quality. For example, the data preparation process usually includes standardizing data formats, enriching source data, and/or removing outliers.



**Figure:3.1 Data Preparation Process**

So, if we talk about this dataset Each row gives details for all of the 7,034 individual customers, e.g. the length of their tenure, internet service type, contract type, monthly charges. The target for prediction is the ‘Churn’ column, indicating whether or not the customer cancelled their service. As we seen in earlier chapter Pandas didn’t detect all of the values in the ‘TotalCharges’ column to be float64 type, so we probably have some non-numeric data in the column.

1. **Gather data**

The data preparation process begins with finding the right data. This can come from an existing data catalog or can be added ad-hoc. As we don’t need to perform this step as we already have our dataset.

2**. Discover and assess data**

After collecting the data, it is important to [discover](https://bi-survey.com/data-discovery) each dataset. This step is about getting to know the data and understanding what has to be done before the data becomes useful in a particular context. Discovery is a big task, but [Talend’s data preparation platform](https://www.talend.com/resources/self-service-data-preparation/) offers visualization tools which help users profile and browse their data. We have already performed this step in the Exploratory Data Analysis.

3. **Cleanse and validate data**

Cleaning up the data is traditionally the most time-consuming part of the data preparation process, but it’s crucial for removing faulty data and filling in gaps. Important tasks here include:

* Removing extraneous data and outliers.
* Filling in missing values.
* Conforming data to a standardized pattern.
* Masking private or sensitive data entries.

Once data has been cleansed, it must be validated by testing for errors in the data preparation process up to this point. Often times, an error in the system will become apparent during this step and will need to be resolved before moving forward.

4. **Transform and enrich data**

Transforming data is the process of updating the format or value entries in order to reach a well-defined outcome, or to make the data more easily understood by a wider audience. Enriching data refers to adding and connecting data with other related information to provide deeper insights.

5. **Store data**

Once prepared, the data can be stored or channeled into a third-party application—such as a business intelligence tool—clearing the way for processing and analysis to take place.

**3.2 Data Cleaning**

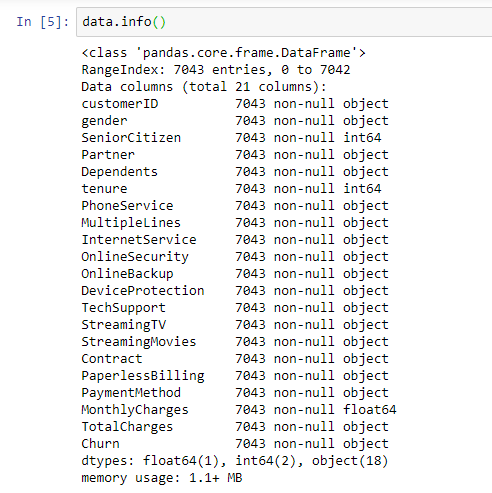
Data cleansing or data cleaning is the process of detecting and correcting (or removing) corrupt or inaccurate [records](https://en.wikipedia.org/wiki/Storage_record) from a record set, [table](https://en.wikipedia.org/wiki/Table_(database)), or [database](https://en.wikipedia.org/wiki/Database) and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the [dirty](https://en.wikipedia.org/wiki/Dirty_data) or coarse data. Data cleansing may be performed interactively with [data wrangling](https://en.wikipedia.org/wiki/Data_wrangling) tools, or as [batch processing](https://en.wikipedia.org/wiki/Batch_processing) through [scripting](https://en.wikipedia.org/wiki/Script_(computing)).

After cleansing, a [data set](https://en.wikipedia.org/wiki/Data_set) should be consistent with other similar data sets in the system. The inconsistencies detected or removed may have been originally caused by user entry errors, by corruption in transmission or storage, or by different [data dictionary](https://en.wikipedia.org/wiki/Data_dictionary) definitions of similar entities in different stores. Data cleaning differs from [data validation](https://en.wikipedia.org/wiki/Data_validation) in that validation almost invariably means data is rejected from the system at entry and is performed at the time of entry, rather than on batches of data.

The actual process of data cleansing may involve removing [typographical errors](https://en.wikipedia.org/wiki/Typographical_error) or validating and correcting values against a known list of entities. The validation may be strict (such as rejecting any address that does not have a valid [postal code](https://en.wikipedia.org/wiki/Postal_code)) or [fuzzy](https://en.wikipedia.org/wiki/Fuzzy_logic) (such as correcting records that partially match existing, known records). Some data cleansing solutions will clean data by cross checking with a validated data set. A common data cleansing practice is data enhancement, where data is made more complete by adding related information. For example, appending addresses with any phone numbers related to that address. Data cleansing may also involve activities like, harmonization of data, and standardization of data. For example, harmonization of short codes (st, rd, etc.) to actual words (street, road, etcetera). Standardization of data is a means of changing a reference data set to a new standard, ex, use of standard codes.

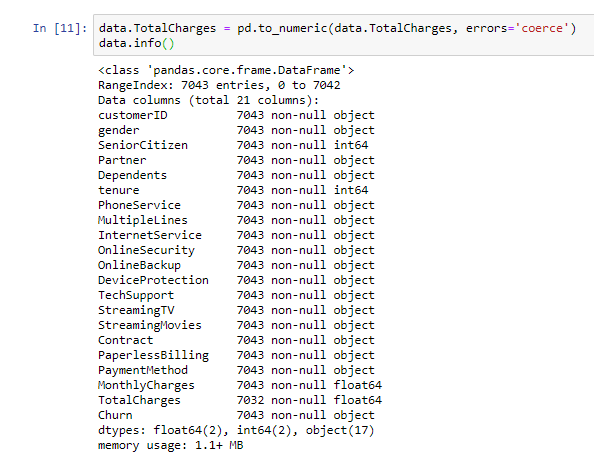
**3.2.1 Converting data from object type to numeric data type**

As we already know in our dataset column ‘TotalCharges’ contain some object whether it should have float datatype to fix the we can convert the column in to numeric with the use of pandas function to numeric ().



**Figure: 3.2 Dataset Information**

Pandas didn’t detect all of the values in the ‘TotalCharges’ column to be float64 type, so we probably have some non-numeric data in the column. This also shows that Total charges have null values.



**Figure: 3.3 Total Charges changed Information of Dataset**

**3.3 Dealing with missing values**

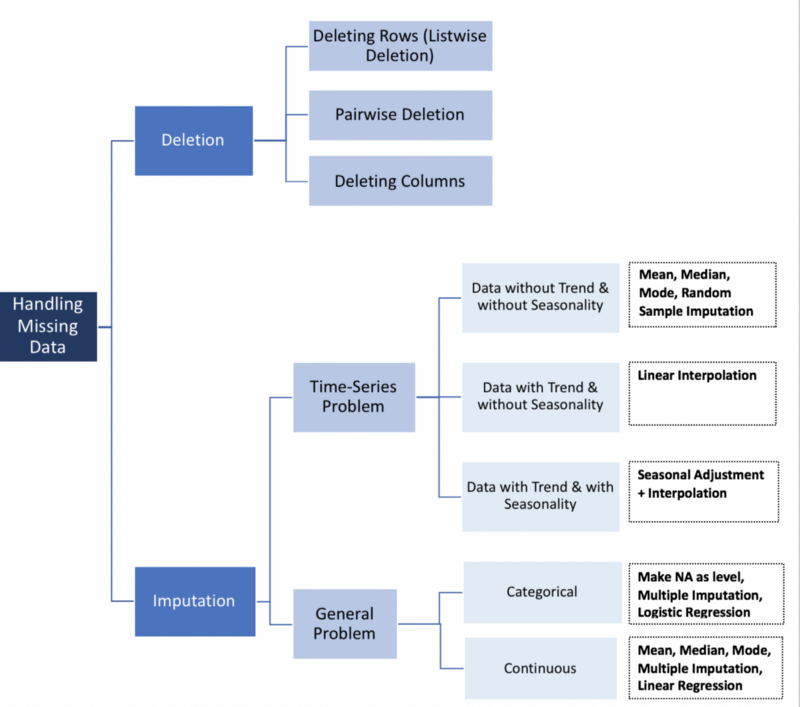
One of the most common problems I have faced in Data Cleaning/Exploratory Analysis is handling the missing values. Firstly, understand that there is NO good way to deal with missing data. I have come across different solutions for data imputation depending on the kind of problem — Time series Analysis, ML, Regression etc. and it is difficult to provide a general solution. In this blog, I am attempting to summarize the most commonly used methods and trying to find a structural solution.

#### Imputation vs Removing Data

Before jumping to the methods of data imputation, we have to understand the reason why data goes missing.

1. **Missing at Random (MAR):**Missing at random means that the propensity for a data point to be missing is not related to the missing data, but it is related to some of the observed data
2. **Missing Completely at Random (MCAR):** The fact that a certain value is missing has nothing to do with its hypothetical value and with the values of other variables.
3. **Missing not at Random (MNAR):**Two possible reasons are that the missing value depends on the hypothetical value (e.g. People with high salaries generally do not want to reveal their incomes in surveys) or missing value is dependent on some other variable’s value (e.g. Let’s assume that females generally don’t want to reveal their ages! Here the missing value in age variable is impacted by gender variable)

In the first two cases, it is safe to remove the data with missing values depending upon their occurrences, while in the third case removing observations with missing values can produce a bias in the model. So, we have to be really careful before removing observations. Note that imputation does not necessarily give better results.



**Figure:3.4 Handling Missing Values**

So, in this project we can deal with missing values by two ways:

1. First way is to delete those rows which contains missing values.

2. Fill the null values with zero.

There are 11 missing values in TotalCharges column. We can fill the missing values with median data, set it to 0 or delete these rows, it is up to you. I prefer deleting these columns because it is a small part compared to all data.



We don't need customerID column for analyzing, so we can also drop this column too

.

**3.3 Data Manipulation**

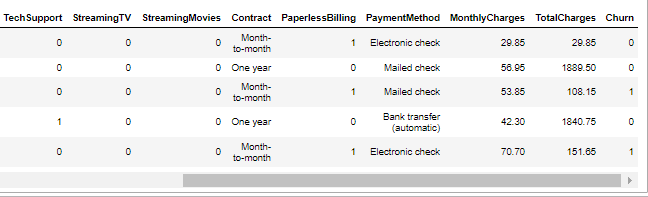
Data manipulation is the process of changing data to make it easier to read or be more organized. For example, a log of data could be organized in [alphabetical order](https://www.computerhope.com/jargon/l/lexisort.htm), making individual entries easier to locate. Data manipulation is often used on [web server](https://www.computerhope.com/jargon/w/webserve.htm) logs to allow a website owner to view their most popular pages as well as their traffic sources.

Users in the accounting field or similar fields often manipulate data to figure out product costs, sales trends, or potential tax obligations. Stock market analysts are frequently using data manipulation to predict trends in the stock market and how stocks might perform in the near future.

Computers may also use data manipulation to display information to users in a more meaningful way, based on code in a software program, web page, or data formatting defined by a user.

In our case we have categorical data,Categorical data is the [statistical data type](https://en.wikipedia.org/wiki/Statistical_data_type) consisting of categorical variables or of data that has been converted into that form, for example as [grouped data](https://en.wikipedia.org/wiki/Grouped_data). More specifically, categorical data may derive from observations made of [qualitative data](https://en.wikipedia.org/wiki/Qualitative_data) that are summarized as counts or [cross tabulations](https://en.wikipedia.org/wiki/Cross_tabulation), or from observations of [quantitative data](https://en.wikipedia.org/wiki/Quantitative_data) grouped within given intervals. Often, purely categorical data are summarized in the form of a [contingency table](https://en.wikipedia.org/wiki/Contingency_table). Handling Categorical data in python can be done through many ways but we go with the easy one. Replacing text columns to integers. The columns below include similar text values so they can be changed in one go.





**Figure:3.5 Manipulation of Data**

**Chapter 4**

**DATA VISUALIZATION**

**4.1 Introduction**

Data visualization is viewed by many disciplines as a modern equivalent of [visual communication](https://en.wikipedia.org/wiki/Visual_communication). It involves the creation and study of the [visual](https://en.wikipedia.org/wiki/Visual_system) representation of [data](https://en.wikipedia.org/wiki/Data).

To communicate information clearly and efficiently, data visualization uses [statistical graphics](https://en.wikipedia.org/wiki/Statistical_graphics), [plots](https://en.wikipedia.org/wiki/Plot_(graphics)), [information graphics](https://en.wikipedia.org/wiki/Infographic) and other tools. Numerical data may be encoded using dots, lines, or bars, to visually communicate a quantitative message. Effective visualization helps users analyze and reason about data and evidence. It makes complex data more accessible, understandable and usable. Users may have particular analytical tasks, such as making comparisons or understanding [causality](https://en.wikipedia.org/wiki/Causality), and the design principle of the graphic (i.e., showing comparisons or showing causality) follows the task. Tables are generally used where users will look up a specific measurement, while charts of various types are used to show patterns or relationships in the data for one or more variables.

Data visualization is both an art and a science. It is viewed as a branch of [descriptive statistics](https://en.wikipedia.org/wiki/Descriptive_statistics) by some, but also as a [grounded theory](https://en.wikipedia.org/wiki/Grounded_theory) development tool by others. Increased amounts of data created by Internet activity and an expanding number of sensors in the environment are referred to as "[big data](https://en.wikipedia.org/wiki/Big_data)" or [Internet of things](https://en.wikipedia.org/wiki/Internet_of_things). Processing, analyzing and communicating this data present ethical and analytical challenges for data visualization. The field of [data science](https://en.wikipedia.org/wiki/Data_science) and practitioners called data scientists help address this challenge.

Data visualization refers to the techniques used to communicate data or information by encoding it as visual objects (e.g., points, lines or bars) contained in graphics. The goal is to communicate information clearly and efficiently to users. It is one of the steps in [data analysis](https://en.wikipedia.org/wiki/Data_analysis) or [data science](https://en.wikipedia.org/wiki/Data_science). According to Friedman (2008) the "main goal of data visualization is to communicate information clearly and effectively through graphical means. It doesn't mean that data visualization needs to look boring to be functional or extremely sophisticated to look beautiful. To convey ideas effectively, both aesthetic form and functionality need to go hand in hand, providing insights into a rather sparse and complex data set by communicating its key-aspects in a more intuitive way. Yet designers often fail to achieve a balance between form and function, creating gorgeous data visualizations which fail to serve their main purpose — to communicate information".

**4.1.1 Characteristics of effective graphical displays**

Excellence in statistical graphics consists of complex ideas communicated with clarity, precision and efficiency. Graphical displays should:

* show the data
* induce the viewer to think about the substance rather than about methodology, graphic design, the technology of graphic production or something else
* avoid distorting what the data has to say
* present many numbers in a small space
* make large data sets coherent
* encourage the eye to compare different pieces of data
* reveal the data at several levels of detail, from a broad overview to the fine structure
* serve a reasonably clear purpose: description, exploration, tabulation or decoration
* be closely integrated with the statistical and verbal descriptions of a data set.

Graphics reveal data. Indeed, graphics can be more precise and revealing than conventional statistical computations

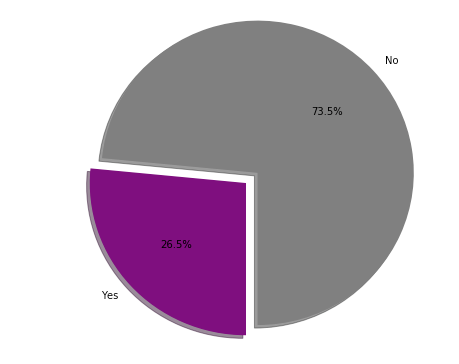
**4.2 EDA (Exploratory Data Analysis) with visual representation**

Perhaps one of the greatest disparities between those who live in the world of data science and those who don’t is the ability to translate the data so that everyone can understand it. One of the best ways for data scientists to present analysis of the data to those outside the industry is by generating visualizations. There are really only two times where visualizations can be generated. The first of those times is during the Exploratory Data Analysis (EDA) where you have initially received the data, cleaned it, sorted it, and created whatever other features you please for a nice clean and full dataset. From that point you can begin to see the relationship between certain variables, the correlation (if any) between features, and generate many more visualization in order to better display the data at hand. The other usually takes place at the conclusion of your analysis where you may want to see visually how your model prediction compared to the that of your EDA visualizations or to see if the inferences your inferred are now able to make are new discoveries or a continuance of the data displayed in the EDA. EDA is an approach to data analysis that uses a variety of techniques, largely visual, to: Maximize insight, reveal underlying structure, check for outliers, test assumptions, and determine optimal factors. It is an approach to analysis of your data that delays the usual assumptions of what kind of model should be used and instead allows the data to speak for itself.

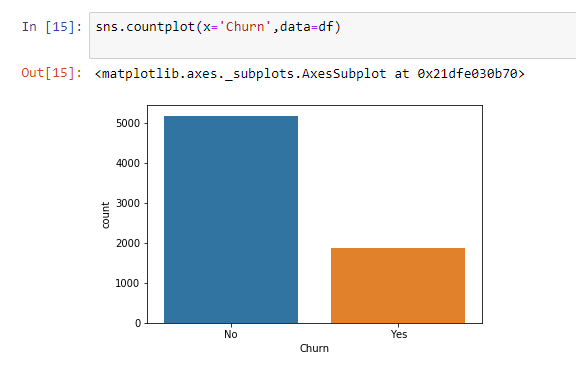
**4.2.1** **Target variable**

We are trying to predict if the client left the company in the previous month. Therefore, we have a binary classification problem with a slightly unbalanced target:

* Churn: No - 73.5%
* Churn: Yes - 26.5%



**Figure: 4.1 Percentage of Churn Customer in Dataset**

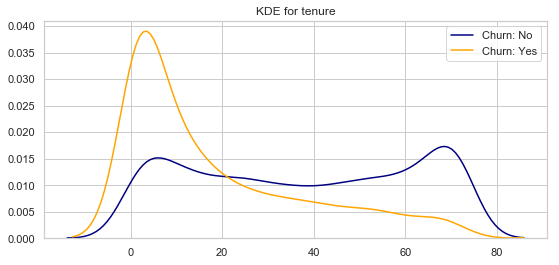


**Figure: 4.2 Distribution of Customer Attrition**

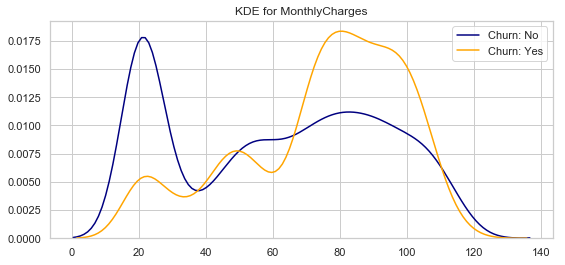
This is the distribution of Churn values. As you can see above, the data set is imbalanced. But for now, it will be ignored. At the time of Modelling it will be fixed.

**4.2.2. Numerical features**

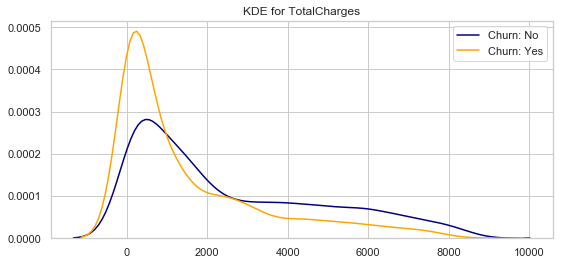
There are only three numerical columns: tenure, monthly charges and total charges. The probability density distribution can be estimate using the seaborn kdeplot function.



**Figure: 4.3 KDE Plot for Tenure**



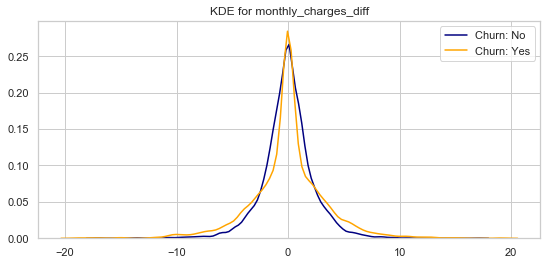
**Figure: 4.4 KDE Plot for Monthly Charges**

 **Figure: 4.5 KDE Plot for Total Charges**

From the plots above we can conclude that:

* Recent clients are more likely to churn
* Clients with higher Monthly Charges are also more likely to churn
* Tenure and Monthly Charges are probably important features

Another feature we can consider is the difference between the Monthly Charges and the TotalCharges divided by the tenure

****

**Figure: 4.6 KDE Plot for Difference in Monthly and Total Charges**

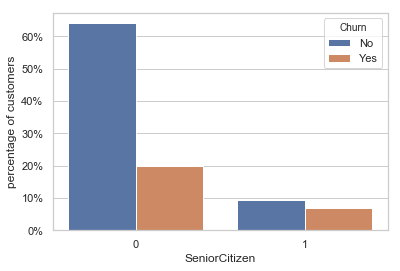
Not a promising feature at first glance, but it might be useful when combined with categorical features.

## 4.2.3. Categorical features

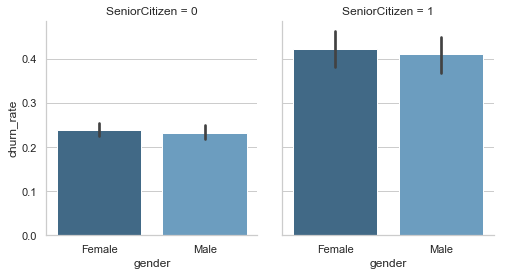
This dataset has 16 categorical features:

* Six binary features (Yes/No)
* Nine features with three unique values each (categories)
* One feature with four unique values

**1. Gender and Age (SeniorCitizen)**

****

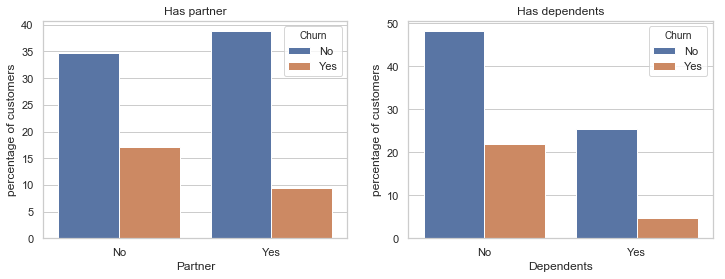
**Figure: 4.7 Senior Citizen**

****

**Figure: 4.8 Senior Citizen on the basis of gender**

* Gender is not an indicative of churn.
* Senior Citizens are only 16% of customers, but they have a much higher churn rate: 42% against 23% for non-senior customers.
* There are no special relations between this categorical value and the main numerical features.

### 2. Partner and dependents



**Figure: 4.9 Customer Has Partner or Dependents**

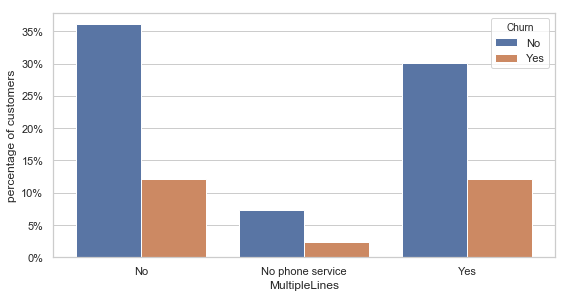
* Customers that doesn't have partners are more likely to churn
* Customers without dependents are also more likely to churn

**3. Phone and Internet services**

Now let's look at the services that customers are using. There are only two main services: phone and internet but the former has many additionals like online backup and security.

**Phone services**

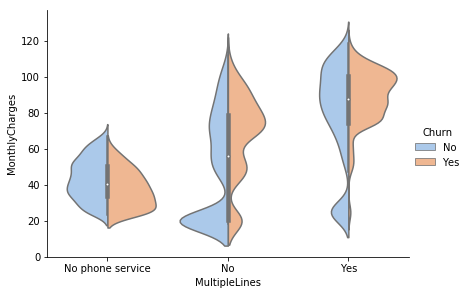
There are only two features here: if the client has phone and if he has more than one line. Both can be summed up in one chart:



**Figure: 4.10 Customer Has Multiple Lines**

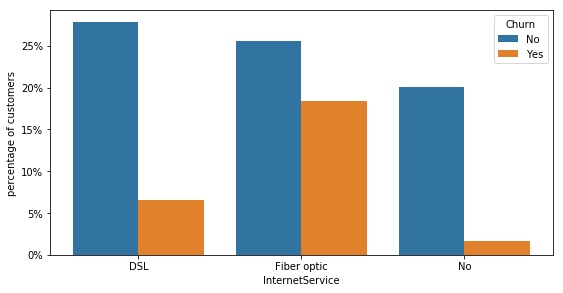
* Few customers don’t have phone service
* Customers with multiple lines have a slightly higher churn rate

Let's see how multiple lines affects the monthly charges:



**Figure: 4.11 Monthly Charges of Customer Has Multiple Lines**

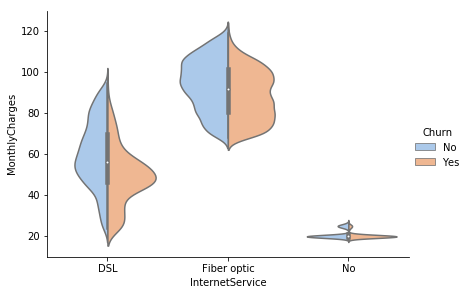
**Internet services**

****

**Figure: 4.12 Customer Has Internet Service**

* Clients without internet have a very low churn rate
* Customers with fiber are more probable to churn than those with DSL connection

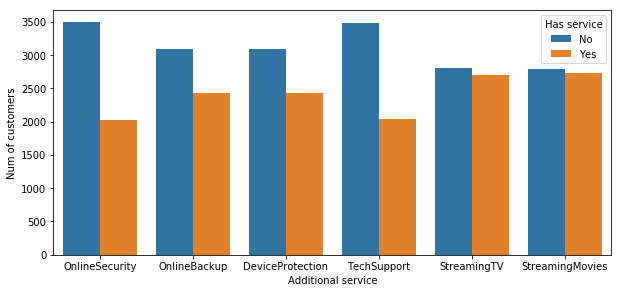
Comparing the Internet service with monthly charges:

****

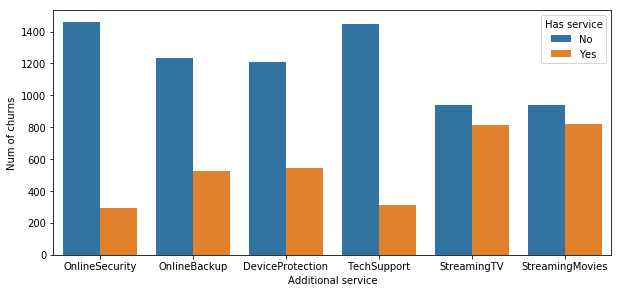
**Figure: 4.13 Monthly Charges of Customer Has Internet Service**

It's interesting how customers with DSL (slower connection) and higher charges are less probable to churn.

**Additional services**

****

**Figure: 4.14 Total Customer Has Additional Service**

****

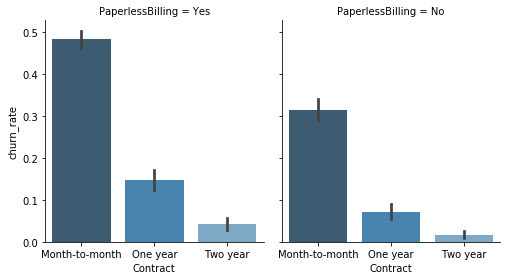
**Figure: 4.15 Churn Customer Has Additional Service**

The first plot shows the total number of customers for each additional service, while the second shows the number of clients that churn. We can see that:

* Customers with the first 4 additional Services (security to tech support) are more unlikely to churn
* Streaming service is not predictive for churn

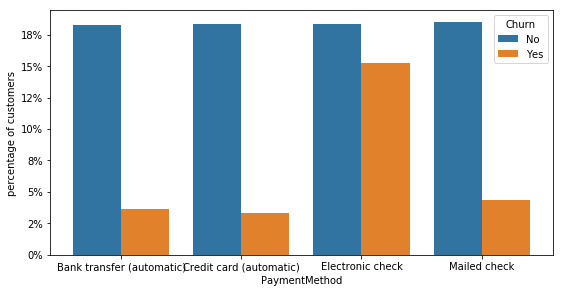
**4. Contract and Payment**

**Contract Method**

****

**Figure: 4.16 Contract**

**Payment Method**

****

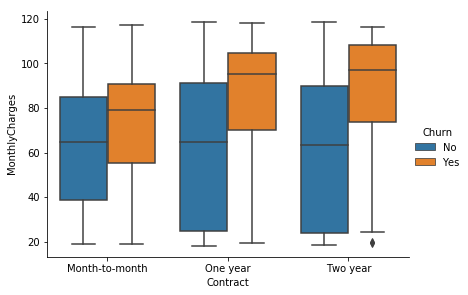
**Figure: 4.17 Payment Method**

A few observations:

* Customers with paperless billing are more probable to churn
* The preferred payment method is Electronic check with around 35% of customers. This method also has a very high churn rate
* Short term contracts have higher churn rates

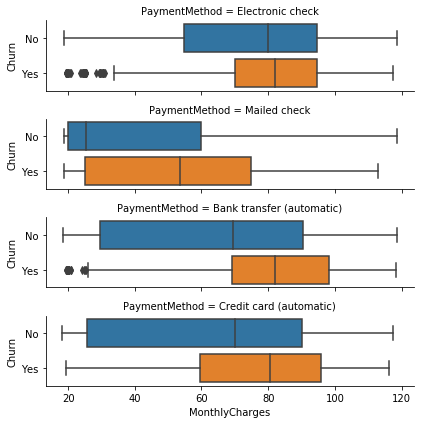
One and two year contracts probably have contractual fines and therefore customers have to wait until the end of contract to churn. A time-series dataset would be better to understand this kind of behavior. Now let's have a look at the relation with numerical features:

**On Contract basis**

****

**Figure: 4.18 Monthly Charges on Contract Basis**

**On Payment basis**

****

**Figure: 4.19 Monthly Charges on Payment Method Basis**

**5. Correlation Between features**

Let's see the correlation between churn and the remaining columns. Customers having month-to-month contract, having fiber optic internet service and using electronic payment are tend to churn more whereas people having two-year contract and having internet service are tend to not churn. The best way to present the Correlation between columns is by Heatmap of seaborn library. To decide which features of the data to include in our predictive churn model, Examine the correlation between churn and each customer feature. There is a good way to quickly check correlations among columns is by visualizing the correlation matrix as a heatmap.

In a correlation matrix or heatmap there are some features that highly correlated on a scale of 0 to 1 and there are negative values as well positive values shows the positive relation between the features and negative values shows the negative relation between the feature. So, positive values are positively correlated with each other and negative values are negatively correlated. Correlation matrix also shows the features that affecting the target variable and in what way positive or negative.

**Avoiding multicollinearity**

Total charges and monthly charges are highly correlated. We try to avoid strongly correlated explanatory variables in regression models. Correlation of explanatory variables is known as multicollinearity, and perfect multicollinearity occurs when the correlation between two independent variables is equal to 1 or -1.

**To see the code of plots and Graph presented in Chapter 4 see appendix ‘A’**

****

**Figure: 4.20 Correlation Matrix**

**Chapter 5**

**MACHINE LEARNING ALGORITHMS**

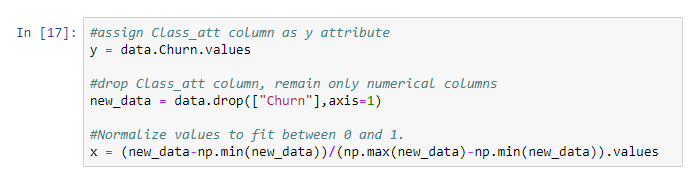
**5.1 Introduction**

In this chapter the implementation of machine learning algorithm and finding the best model fit for the dataset. Machine learning is a concept which allows the machine to learn from examples and experience, and that too without being explicitly programmed. So instead of you writing the code, what you do is you feed data to the generic algorithm, and the algorithm/ machine builds the logic based on the given data.

Machine Learning algorithm is trained using a training data set to create a model. When new input data is introduced to the ML algorithm, it makes a prediction on the basis of the model.  
The prediction is evaluated for accuracy and if the accuracy is acceptable, the Machine Learning algorithm is deployed. If the accuracy is not acceptable, the Machine Learning algorithm is trained again and again with an augmented training data set. Before implementing the algorithms independent and dependent variables are to be prepared so that there should be no problem while executing the algorithms. So, lets start prepare our dependent and independent variables.

**5.2 Prepare x and y**

First, separate x and y values. y would be our class which is Churn column in this dataset. x would be the remaining columns. Also, apply normalization to x in order to scale all values between 0 and 1.

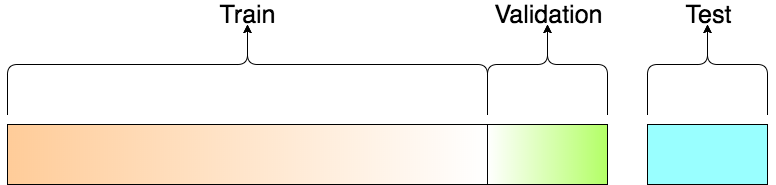


**5.3 Splitting Data**

The dataset going to be split into Training dataset and Test dataset.

**Training Dataset:** The sample of data used to fit the model. The actual dataset that we use to train the model (weights and biases in the case of Neural Network). The model sees and learns from this data.

**Test Dataset:** The sample of data used to provide an unbiased evaluation of a final model fit on the training dataset. The Test dataset provides the gold standard used to evaluate the model. It is only used once a model is completely trained (using the train and validation sets). The test set is generally what is used to evaluate competing models (For example on many Kaggle competitions, the validation set is released initially along with the training set and the actual test set is only released when the competition is about to close, and it is the result of the model on the Test set that decides the winner). Many a times the validation set is used as the test set, but it is not good practice. The test set is generally well curated. It contains carefully sampled data that spans the various classes that the model would face, when used in the real world.



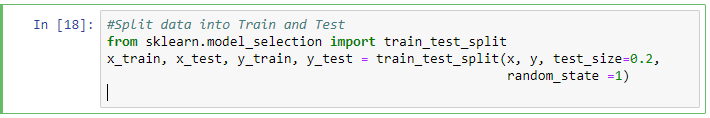
**Figure no – 5.1: A visualization of the splits**

**Validation Dataset:**  The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration.

The validation set is used to evaluate a given model, but this is for frequent evaluation. We as machine learning engineers use this data to fine-tune the model hyperparameters. Hence the model occasionally sees this data, but never does it “Learn” from this. We (mostly humans, at-least as of 2017) use the validation set results and update higher level hyperparameters. So, the validation set in a way affects a model, but indirectly. Models with very few hyperparameters will be easy to validate and tune, so you can probably reduce the size of your validation set, but if your model has many hyperparameters, you would want to have a large validation set as well (although you should also consider cross validation).

**Cross Validation:** Many a times, people first split their dataset into 2 — Train and Test. After this, they keep aside the Test set, and randomly choose X% of their Train dataset to be the actual Train set and the remaining (100-X) % to be the Validation set, where X is a fixed number (say 80%), the model is then iteratively trained and validated on these different sets. There are multiple ways to do this, and is commonly known as Cross Validation. Basically, you use your training set to generate multiple splits of the Train and Validation sets. Cross validation avoids over fitting and is getting more and more popular, with K-fold Cross Validation being the most popular method of cross validation.

So, Split the data set as train and test with %20-%80 ratio.



## 5.4 Apply Machine Learning Algorithms

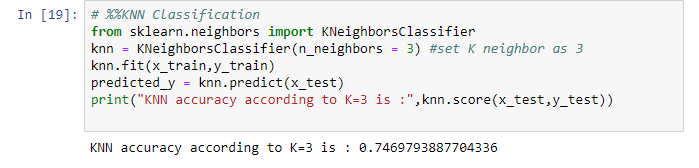
Let's start to apply some machine learning algorithms and find the accuracy of each Model.

**5.4.1 KNN Classification**

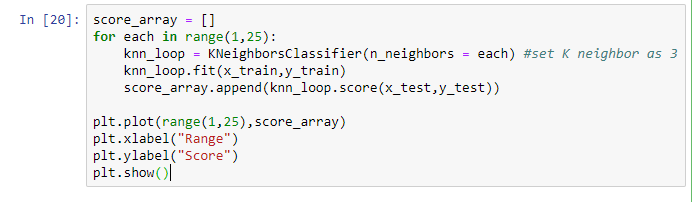
The k-nearest neighbors algorithm (k-NN) is a [non-parametric](https://en.wikipedia.org/wiki/Non-parametric_statistics) method used for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression](https://en.wikipedia.org/wiki/Regression_analysis). In both cases, the input consists of the k closest training examples in the [feature space](https://en.wikipedia.org/wiki/Feature_space). The output depends on whether k-NN is used for classification or regression:

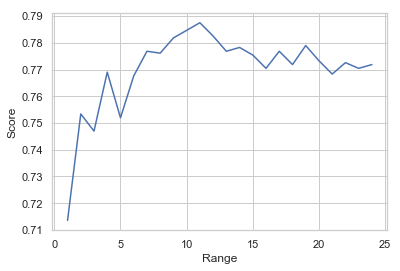
* In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive [integer](https://en.wikipedia.org/wiki/Integer), typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.
* In k-NN regression, the output is the property value for the object. This value is the average of the values of k nearest neighbors

Applying KNN Classification on to the dataset



We assume K = 3 for first iteration, but actually we don't know what is the optimal K value that gives maximum accuracy. So, we can write a for loop that iterates for example 25 times and gives the accuracy at each iteration. So that we can find the optimal K value.





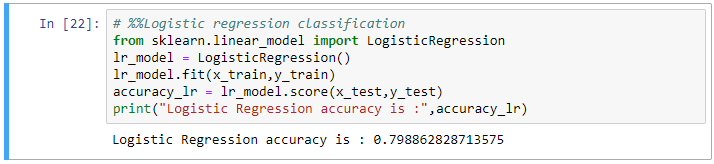
**Figure: 5.2 Analyzing Multiple Iteration For KNN**

As you can see above, if we use K = 11, then we get maximum score of %78.7

**5.4.2 Logistic Regression Classification**

[Logistic regression](http://www.statisticssolutions.com/academic-solutions/membership-resources/member-profile/data-analysis-plan-templates/data-analysis-plan-logistic-regression/) is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).  Like all regression analyses, the logistic regression is a predictive analysis.  Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Sometimes logistic regressions are difficult to interpret; the Intellectus Statistics tool easily allows you to conduct the analysis, then in plain English interprets the output.



**5.4.3 SVM (Support Vector Machine) Classification**

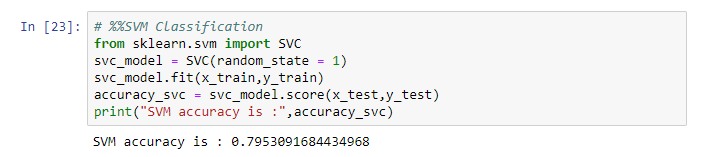
The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.

To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. It becomes difficult to imagine when the number of features exceeds

Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.

Applying SVM on the dataset



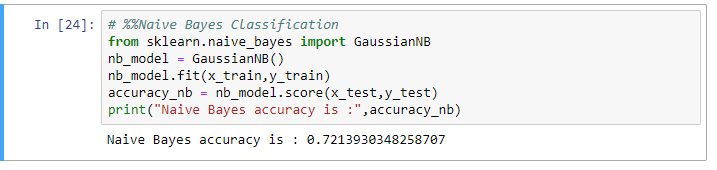
**5.4.4 Naive Bayes Classification**

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of [feature](https://en.wikipedia.org/wiki/Feature_vector) values, where the class labels are drawn from some finite set. There is not a single [algorithm](https://en.wikipedia.org/wiki/Algorithm) for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is [independent](https://en.wikipedia.org/wiki/Independence_(probability_theory)) of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible [correlations](https://en.wikipedia.org/wiki/Correlation_and_dependence) between the color, roundness, and diameter features.

For some types of probability models, naive Bayes classifiers can be trained very efficiently in a [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) setting. In many practical applications, parameter estimation for naive Bayes models uses the method of [maximum likelihood](https://en.wikipedia.org/wiki/Maximum_likelihood); in other words, one can work with the naive Bayes model without accepting [Bayesian probability](https://en.wikipedia.org/wiki/Bayesian_probability) or using any Bayesian methods.

An advantage of naive Bayes is that it only requires a small number of training data to estimate the parameters necessary for classification

Applying Naïve Bayes Classification on the dataset



**5.4.5 Decision Tree Classification**

A decision tree is a [decision support](https://en.wikipedia.org/wiki/Decision_support_system) tool that uses a [tree-like](https://en.wikipedia.org/wiki/Tree_(graph_theory)) [model](https://en.wikipedia.org/wiki/Causal_model) of decisions and their possible consequences, including [chance](https://en.wikipedia.org/wiki/Probability) event outcomes, resource costs, and [utility](https://en.wikipedia.org/wiki/Utility). It is one way to display an [algorithm](https://en.wikipedia.org/wiki/Algorithm) that only contains conditional control statements.

Decision trees are commonly used in [operations research](https://en.wikipedia.org/wiki/Operations_research), specifically in [decision analysis](https://en.wikipedia.org/wiki/Decision_analysis), to help identify a strategy most likely to reach a [goal](https://en.wikipedia.org/wiki/Goal), but are also a popular tool in [machine learning](https://en.wikipedia.org/wiki/Decision_tree_learning).

A decision tree is a [flowchart](https://en.wikipedia.org/wiki/Flowchart)-like structure in which each internal node represents a "test" on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represent classification rules.

In [decision analysis](https://en.wikipedia.org/wiki/Decision_analysis), a decision tree and the closely related [influence diagram](https://en.wikipedia.org/wiki/Influence_diagram) are used as a visual and analytical decision support tool, where the [expected values](https://en.wikipedia.org/wiki/Expected_value)(or [expected utility](https://en.wikipedia.org/wiki/Expected_utility)) of competing alternatives are calculated.

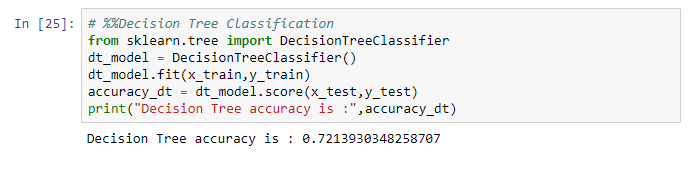
A decision tree consists of three types of nodes:

* Decision nodes – typically represented by squares
* Chance nodes – typically represented by circles
* End nodes – typically represented by triangles

Decision trees are commonly used in [operations research](https://en.wikipedia.org/wiki/Operations_research) and [operations management](https://en.wikipedia.org/wiki/Operations_management). If, in practice, decisions have to be taken online with no recall under incomplete knowledge, a decision tree should be paralleled by a [probability](https://en.wikipedia.org/wiki/Probability) model as a best choice model or online selection model [algorithm](https://en.wikipedia.org/wiki/Algorithm). Another use of decision trees is as a descriptive means for calculating [conditional probabilities](https://en.wikipedia.org/wiki/Conditional_probability).

Decision trees, [influence diagrams](https://en.wikipedia.org/wiki/Influence_diagrams), [utility functions](https://en.wikipedia.org/wiki/Utility_function), and other [decision analysis](https://en.wikipedia.org/wiki/Decision_analysis) tools and methods are taught to undergraduate students in schools of business, health economics, and public health, and are examples of operations research or [management science](https://en.wikipedia.org/wiki/Management_science) methods.

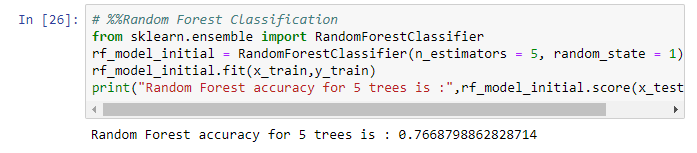
Applying Decision Tree Classification on the dataset



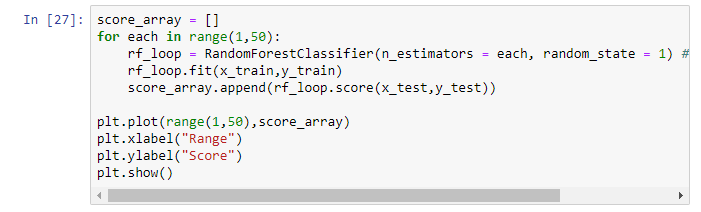
**5.4.6 Random Forest Classification**

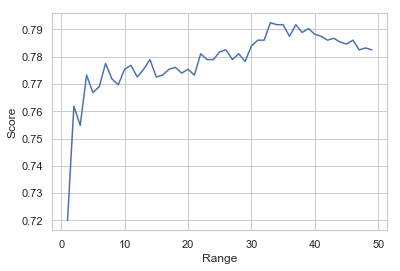
Random forests  or  random decision forests  are an  [ensemble learning](https://en.wikipedia.org/wiki/Ensemble_learning) method for [classification](https://en.wikipedia.org/wiki/Statistical_classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis) and other tasks that operates by constructing a multitude of [decision trees](https://en.wikipedia.org/wiki/Decision_tree_learning) at training time and outputting the class that is the [mode](https://en.wikipedia.org/wiki/Mode_(statistics)) of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of [overfitting](https://en.wikipedia.org/wiki/Overfitting) to their [training set](https://en.wikipedia.org/wiki/Test_set).

Applying Random Forest Classification



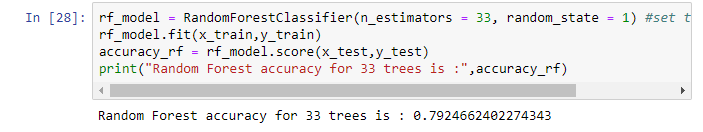
I set tree number as 5 initially. But I want to find the appropriate tree number. Let's try to find the best number with trying 1 to 50.



****

**Figure: 5.3 Analyzing Multiple Iteration for Random Forest**

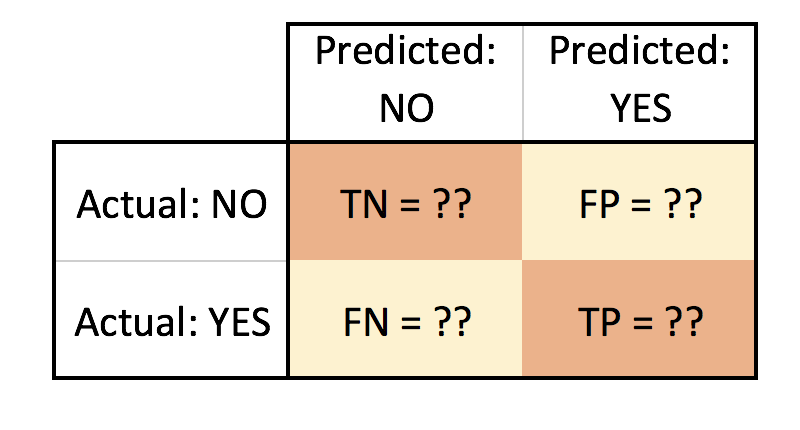
As you can see, the highest accuracy is at n\_estimators = 33.



**5.4.7 Confusion matrix**

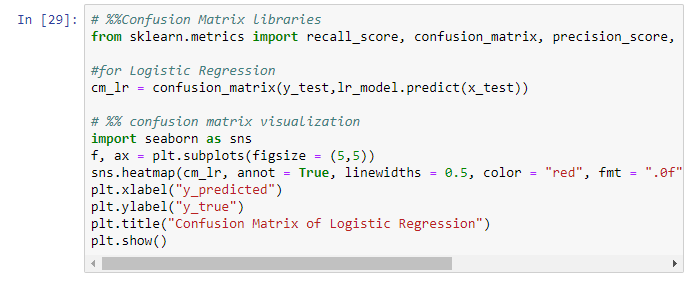
Logistic regression and SVC classification algorithms have the highest accuracy. But as I mentioned before, our data is imbalanced. So, it is important to look at the confusion matrix according to these two algorithms. With imbalanced datasets, the highest accuracy does not give the best model. Assume we have 1000 total rows; 10 rows are churn and 990 rows are non-churn. If we find all these 10 churn rows as non-churn, then the accuracy will be still %99. Although it is a wrong model, if we do not look at the confusion matrix, then we cannot see the mistake.

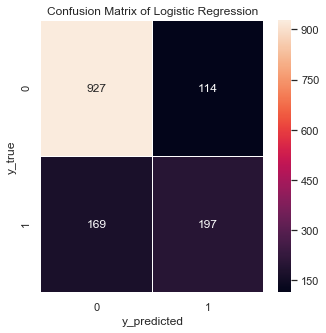
Confusion matrix gives us FN (false negative), FP (false positive), TN (true negative) and TP (true positive) values



**Figure: 5.4 Confusion Matrix Structure**

Implementation of Confusion Matrix





**Figure: 5.5 Confusion Matrix**

For logistic regression confusion matrix;

TN = 927

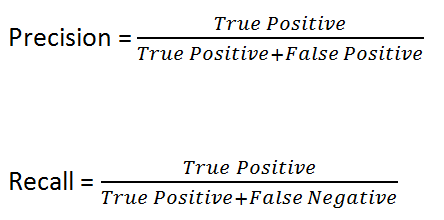
FP = 114

FN = 169

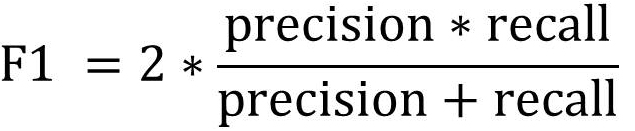
TP = 197

This means; there are total 927+114 = 1041 actual non-churn values and the algorithm predict 927 of them as non-churn and 114 of them churn. Also, there are total 169 + 197 = 366 actual churn values and the algorithm predict 169 of them as non-churn and 197 of them as churn.

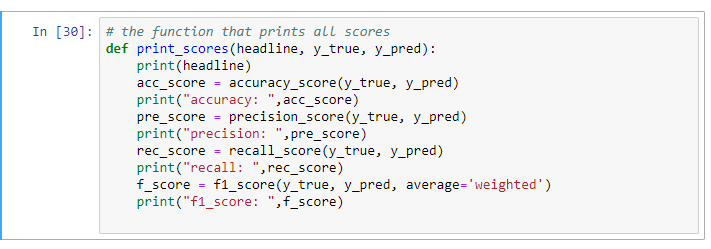
Accuracy should not be used as solely metric for imbalance datasets. There are some other metrics named as recall and precision.



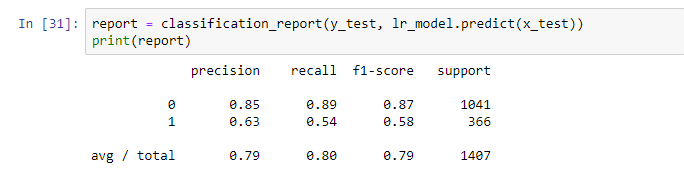
Sometimes we get high recall and low precision or vice versa. There is another metric that combines both precision and recall like below. We will use F1 score to identify the best algorithm score



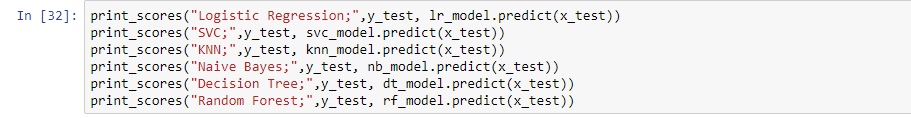
Below written a function that calculates and print both accuracy, recall, precision and weighted F1 score.

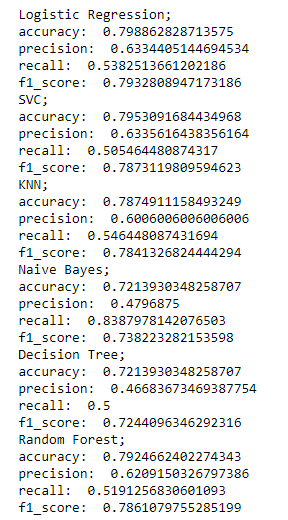


We can also use classification report function from sklearn library to show all these metrics



Now printing all results of each algorithm





**Chapter 6**

**SUMMARY & CONCLUSION**

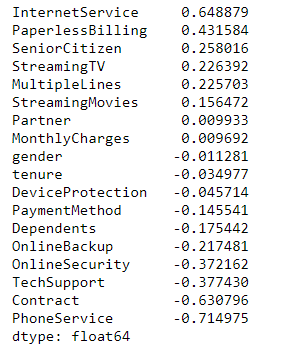
**6.1 Summary**

Churn prediction is a function that involves systematic analysis of customer data for  
identifying and analyzing patterns and trends of customer loyalty and blend. The detected  
patterns and trends can be used by telecommunication industries to improve customer  
relationship and at the same time improve net profit. Identification of churners and non-churners is a time consuming and critical task, that has to be performed carefully, as the  
future growth of the company relies on the result of such an analysis. This task is  
considered challenging because of two reasons, (i) customer information volume has  
increased and (ii) the data available is inconsistent and incomplete thus making the task  
of formal analysis a difficult task. Further, due to its vast size, investigation and analysis of  
customer database takes longer duration due to the complexity of these issues.  
As information science and technology progress, sophisticated data mining and  
artificial intelligence tools are increasingly accessible to the telecommunication sector.  
These techniques combined with state-of-the-art computers can process thousands of  
instructions in seconds, saving precious time. In addition, installing and running software  
often costs less than hiring and training personnel. Computers are also less prone to errors than human investigators, especially those who work long hours. The current needs of telecom companies are a tool that can be used to help them to understand customer patterns and locate churners and possible actions that can be taken to convert the churners to non-churners. This tool is called as ‘Churn Prediction and actionable knowledge discovery and the main goal is to provide timely and pertinent customer information to decision-makers in a company. The present research work focus on developing such a system that can be used by telecom industry easily discover customer patterns and trends, make forecasts, find relationships and possible explanations and identify possible churners. The proposed system proposes the use of data mining techniques during the design and development. To obtain an extra edge over competitive business, telecommunication industries are relying more and more on CRM combined with data mining techniques. In this study, customer’s churning behavior is predicted along with actionable knowledge discovery. The proposed system consists of three main steps, namely, data preprocessing, Churn Prediction and actionable knowledge discovery. Each of the three steps is treated as a separate research phase and the phases are interconnected to each other, where the output of one phase is taken as input by the next phase. As incomplete dataset and presence of outliers during churn management process reduces the efficiency of prediction, a preprocessing step handling the missing values and outlier removal is used. In the first phase, preprocessing, a missing value handling procedure and outlier detection algorithm is proposed. The missing value handling procedure enhances the operation of a Pandas Library and with help this particular library handling missing values. An iterative procedure for handling missing value is also proposed. In the same phase, the visualization also takes place that helps us to understand the customer data. This Process is enhanced through the use of different set of plots and graphs and getting good insights of customer data. The use of plots and graphs makes process more scalable and quicker in optimizing. In the second phase of the study, classification algorithms are analyzed for Churn Prediction. In this research work, there are several classification algorithms used to show that which algorithm perform better work in detecting churn user and which model have more accuracy. The proposed algorithms perform Churn Prediction in two steps. The first step performs Classification process with several algorithm, while the second step shows that as the dataset is imbalanced so the prediction that is generated by the algorithms is not 100% accurate so showing accuracy F1 score would be much effective than accuracy in this scenario that covers in this step. The first step, u**s**e**s** amultiple classification algorithm that helps in predicting the churners and with their accuracy rate in this step six classification algorithm used: K-Nearest Neighbor, Logistic Regression, Support Vector Machine (SVM), Naïve Bayes, Decision Tree (DT), Random Forest.The second step analyzes the mistake that accruing in the algorithms with the use of Confusion matrix that shows how much correct prediction model has done as the dataset is imbalanced so F1 score is showing better result than accuracy. The final phase of the study focuses on discovering action sets, which can be used as knowledge to convert churners to loyal customers and plan promotional activities to maintain loyal customers. For this purpose, the system uses to show the weights of different features that somehow effect the churners and with help of this knowledge the planning of promotional activities can be mad and that certainly increase the number of non-churners in the industry and convert the churners to loyal customers. The performance evaluation stage was conducted in three stages, each analyzing the performance of the missing value handling Features, customer loyalty assessment model and actionable knowledge discovery system. A telecom customer dataset from IDEA customer care was used during experiments. The missing value handling features was evaluated using two performance method, pandas, one hot encoding. The customer loyalty assessment procedure used accuracy and speed to analyze the algorithms. The actionable knowledge discovery system analysis was based on error rate, accuracy achieved and speed of discovery. The various experiments conducted proved that the proposed algorithms and the proposed CRM system for Churn Prediction and actionable knowledge discovery are efficient. Experimental results showed that the system is effective in terms of analysis accuracy and speed in identifying common customer behavior patterns and future churn prediction. The developed system has promising value in the current constantly changing telecommunication industry and can be used as effectively by companies to improve customer relationship and improve business opportunities.

**6.2 Conclusion**

Churn prediction as the customer information is given in the dataset there are some results of the analysis that is performed in this dataset there are three numerical features are good predictors for churn, specially tenure and the monthly charges also showing some impact on the target variable but total charges not showing a good impact but it has his importance as the probability density showing in the kdeplot . As we've seen, in our analysis customers with Fiber optic are very likely to churn as the technology is new that shows the need of trust between the company and customer, while those with long term contracts are not as they trust the company. On the other hand, gender and streaming are not important features and It might be interesting to drop additional services with the label 'No internet service', since they are highly correlated as shown in the correlation matrix.

Since data set is imbalanced, so accuracy would be less preferred as compared to F1 score. In this project F1 score is preferred rather than accuracy. Logistic Regression shows us the highest F1 Score, so it shows that this is the best model in all of the classification algorithms that we have used in our project according to the F1 score. Naive Bayes is the worst model because it gives the lowest F1 score. Gender has no impact on churn. People having month-to-month contract tend to churn more than people having long term contracts. As the tenure increases, the probability of churn decreases. As monthly charges increase, the probability of churn increases. As the dataset have different features there are the reasons that a customer can be a churn user of not as figure 6.1 shows there are certain features that effect the Customer attrition or Customer churn in a positive way as shown below there is 64% chances that a customer can leave the company because of Internet Services ,43% chances for Paperless Billing ,25% chances for Senior Citizen, and so on as for the negative values they didn’t effect the customer to be churn so they can be the reason for the customer to not become churn.



**Figure: 6.1 Features affecting Customer churn**

**FUTURE RESEARCH DIRECTIONS**

The following can be considered to improve the proposed customer loyalty assessment model and actionable knowledge discovery system.

* The proposed models can be further enhanced, if the processes can be parallelizing. This is feasible, by identifying operations that are independent to each other and propose a parallel architecture to improve the performance.
* The Missing value can be handled with different approach.
* Amount of memory used in prediction and action discovery is another area which can be analyzed in future.
* Classification process can be improved by using advanced techniques like ensemble  
  clustering or ensemble classification.

**REFERENCES**

1. [www.google.com](http://www.google.com)
2. [What is Exploratory Data Analysis? – Towards Data Science](https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15)
3. [Visualizing your Exploratory Data Analysis – Towards Data Science](https://towardsdatascience.com/visualizing-your-exploratory-data-analysis-d2d6c2e3b30e)
4. [Python Seaborn Tutorial For Beginners (article) - Data Camp](https://www.datacamp.com/community/tutorials/seaborn-python-tutorial)
5. [A step-by-step guide for creating advanced Python data visualizations with Seaborn / Matplotlib](https://towardsdatascience.com/a-step-by-step-guide-for-creating-advanced-python-data-visualizations-with-seaborn-matplotlib-1579d6a1a7d0)
6. <https://en.wikipedia.org/wiki/Data_preparation>
7. <https://en.wikipedia.org/wiki/Exploratory_data_analysis>
8. <https://en.wikipedia.org/wiki/Data_visualization>
9. <https://en.wikipedia.org/wiki/Machine_learning>
10. <https://en.wikipedia.org/wiki/Statistical_classification>
11. [www.TowardsDataScience.com](http://www.TowardsDataScience.com)
12. <https://web.stanford.edu/~hastie/Papers/ESLII.pdf>
13. <http://users.isr.ist.utl.pt/~wurmd/Livros/school/Bishop%20-%20Pattern%20Recognition%20And%20Machine%20Learning%20-%20Springer%20%202006.pdf>
14. <http://bedford-computing.co.uk/learning/wp-content/uploads/2015/10/Python-for-Data-Analysis.pdf>
15. <https://doc.lagout.org/science/Artificial%20Intelligence/Machine%20learning/Machine%20Learning%20for%20Hackers_%20Case%20Studies%20and%20Algorithms%20to%20Get%20You%20Started%20%5BConway%20%26%20White%202012-02-25%5D.pdf>
16. <http://web4.cs.ucl.ac.uk/staff/D.Barber/textbook/090310.pdf>
17. <http://mmds.org/#ver21>
18. <http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/understanding-machine-learning-theory-algorithms.pdf>

**APPENDIX ‘A’**

**Code**

**(a) Exploratory Data Analysis (EDA)**

import numpy as np # linear algebra

import pandas as pd # data processing, CSV file I/O (e.g. pd.read\_csv)

import matplotlib.pyplot as plt

import seaborn as sns

data = pd.read\_csv("E://DATASETS//WA\_Fn-UseC\_-Telco-Customer-Churn.csv")

data.head()

data.shape

data.describe()

data.Churn.unique()

data.Churn.value\_counts()

data.info()

data.TotalCharges = pd.to\_numeric(data.TotalCharges, errors='coerce')

#delete rows including null values

data.dropna(inplace = True)

# Deleting column customerID

data.drop(["customerID"],axis=1,inplace = True)

#data manipulation

data.gender = [1 if each == "Male" else 0 for each in data.gender]

columns\_to\_convert = ['Partner',

'Dependents',

'PhoneService',

'MultipleLines',

'OnlineSecurity',

'OnlineBackup',

'DeviceProtection',

'TechSupport',

'StreamingTV',

'StreamingMovies',

'PaperlessBilling',

'Churn']

for item in columns\_to\_convert:

data[item] = [1 if each == "Yes" else 0 if each == "No" else -1 for each in data[item]]

data.head()

**(b)Data Visualization**

sns.countplot(x="Churn",data=data)

#Probability Density Distribution estimated by kdeplot

#(kernel density estimation plot) to measure the pdf or density of continous random Variable

def kdeplot(feature):

plt.figure(figsize=(9, 4))

plt.title("KDE for {}".format(feature))

ax0 = sns.kdeplot(df[df['Churn'] == 'No'][feature].dropna(), color= 'navy', label= 'Churn: No')

ax1 = sns.kdeplot(df[df['Churn'] == 'Yes'][feature].dropna(), color= 'orange', label= 'Churn: Yes')

kdeplot('tenure')

kdeplot('MonthlyCharges')

kdeplot('TotalCharges')

# Calculate features

df['total\_charges\_to\_tenure\_ratio'] = df['TotalCharges'] / df['tenure']

df['monthly\_charges\_diff'] = df['MonthlyCharges'] - df['total\_charges\_to\_tenure\_ratio']

kdeplot('monthly\_charges\_diff')

#Calculating Percentage of customer

def barplot\_percentages(feature, orient='v', axis\_name="percentage of customers"):

ratios = pd.DataFrame()

g = df.groupby(feature)["Churn"].value\_counts().to\_frame()

g = g.rename({"Churn": axis\_name}, axis=1).reset\_index()

g[axis\_name] = g[axis\_name]/len(df)

if orient == 'v':

ax = sns.barplot(x=feature, y= axis\_name, hue='Churn', data=g, orient=orient)

ax.set\_yticklabels(['{:,.0%}'.format(y) for y in ax.get\_yticks()])

else:

ax = sns.barplot(x= axis\_name, y=feature, hue='Churn', data=g, orient=orient)

ax.set\_xticklabels(['{:,.0%}'.format(x) for x in ax.get\_xticks()])

ax.plot()

barplot\_percentages("SeniorCitizen")

#senior citizen on the basis of gender

df['churn\_rate'] = df['Churn'].replace("No", 0).replace("Yes", 1)

g = sns.FacetGrid(df, col="SeniorCitizen", height=4, aspect=.9)

ax = g.map(sns.barplot, "gender", "churn\_rate", palette = "Blues\_d", order= ['Female', 'Male'])

#customer having partner or dependents

fig, axis = plt.subplots(1, 2, figsize=(12,4))

axis[0].set\_title("Has partner")

axis[1].set\_title("Has dependents")

axis\_y = "percentage of customers"

# Plot Partner column

gp\_partner = df.groupby('Partner')["Churn"].value\_counts()/len(df)\*100

gp\_partner = gp\_partner.to\_frame().rename({"Churn": axis\_y}, axis=1).reset\_index()

ax = sns.barplot(x='Partner', y= axis\_y, hue='Churn', data=gp\_partner, ax=axis[0])

# Plot Dependents column

gp\_dep = df.groupby('Dependents')["Churn"].value\_counts()/len(df)\*100

gp\_dep = gp\_dep.to\_frame().rename({"Churn": axis\_y}, axis=1).reset\_index()

ax = sns.barplot(x='Dependents', y= axis\_y, hue='Churn', data=gp\_dep, ax=axis[1])

# percentage of customer of multiple lines

plt.figure(figsize=(9, 4.5))

barplot\_percentages("MultipleLines", orient='v')

#for violin plot

ax = sns.catplot(x="MultipleLines", y="MonthlyCharges", hue="Churn", kind="violin",

split=True, palette="pastel", data=df, height=4.2, aspect=1.4)

#Internet services

plt.figure(figsize=(9, 4.5))

barplot\_percentages("InternetService", orient="v")

ax = sns.catplot(x="InternetService", y="MonthlyCharges", hue="Churn", kind="violin",

split=True, palette="pastel", data=df, height=4.2, aspect=1.4);

#Additional services

cols = ["OnlineSecurity", "OnlineBackup", "DeviceProtection", "TechSupport", "StreamingTV", "StreamingMovies"]

df1 = pd.melt(df[df["InternetService"] != "No"][cols]).rename({'value': 'Has service'}, axis=1)

plt.figure(figsize=(10, 4.5))

ax = sns.countplot(data=df1, x='variable', hue='Has service')

ax.set(xlabel='Additional service', ylabel='Num of customers')

plt.show()

# no. of churn user

plt.figure(figsize=(10, 4.5))

df1 = df[(df.InternetService != "No") & (df.Churn == "Yes")]

df1 = pd.melt(df1[cols]).rename({'value': 'Has service'}, axis=1)

ax = sns.countplot(data=df1, x='variable', hue='Has service', hue\_order=['No', 'Yes'])

ax.set(xlabel='Additional service', ylabel='Num of churns')

plt.show()

#Contract and Payment

g = sns.FacetGrid(df, col="PaperlessBilling", height=4, aspect=.9)

ax = g.map(sns.barplot, "Contract", "churn\_rate", palette = "Blues\_d", order= ['Month-to-month', 'One year', 'Two year'])

ax = sns.catplot(x="Contract", y="MonthlyCharges", hue="Churn", kind="box", data=df, height=4.2, aspect=1.4)

plt.figure(figsize=(9, 4.5))

barplot\_percentages("PaymentMethod", orient='v')

ax = sns.catplot(y="Churn", x="MonthlyCharges", row="PaymentMethod", kind="box", data=df, height=1.5, aspect=4, orient='h')

# Correlations between customer data features and customer churn

corr = data.corr()

sns.heatmap(corr, xticklabels=corr.columns.values, yticklabels=corr.columns.values, annot = True, annot\_kws={'size':12},fmt='.2f')

heat\_map=plt.gcf()

heat\_map.set\_size\_inches(20,15)

plt.xticks(fontsize=10)

plt.yticks(fontsize=10)

plt.show()

**(c) Machine Learning algorithms**

# converting categorical to numerical

data = pd.get\_dummies(data=data)

data.head()

#assign Class\_att column as y attribute

y = data.Churn.values

#drop Class\_att column, remain only numerical columns

new\_data = data.drop(["Churn"],axis=1)

#Normalize values to fit between 0 and 1.

x = (new\_data-np.min(new\_data))/(np.max(new\_data)-np.min(new\_data)).values

#Split data into Train and Test

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 =1)

# %%KNN Classification

from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n\_neighbors = 3) #set K neighbor as 3

knn.fit(x\_train,y\_train)

predicted\_y = knn.predict(x\_test)

print("KNN accuracy according to K=3 is :",knn.score(x\_test,y\_test))

score\_array = []

for each in range(1,25):

knn\_loop = KNeighborsClassifier(n\_neighbors = each) #set K neighbor as 3

knn\_loop.fit(x\_train,y\_train)

score\_array.append(knn\_loop.score(x\_test,y\_test))

plt.plot(range(1,25),score\_array)

plt.xlabel("Range")

plt.ylabel("Score")

plt.show()

knn\_model = KNeighborsClassifier(n\_neighbors = 11) #set K neighbor as 11

knn\_model.fit(x\_train,y\_train)

predicted\_y = knn\_model.predict(x\_test)

accuracy\_knn = knn\_model.score(x\_test,y\_test)

print("KNN accuracy according to K=11 is :",accuracy\_knn)

# %%Logistic regression classification

from sklearn.linear\_model import LogisticRegression

lr\_model = LogisticRegression()

lr\_model.fit(x\_train,y\_train)

accuracy\_lr = lr\_model.score(x\_test,y\_test)

print("Logistic Regression accuracy is :",accuracy\_lr)

# %%SVM Classification

from sklearn.svm import SVC

svc\_model = SVC(random\_state = 1)

svc\_model.fit(x\_train,y\_train)

accuracy\_svc = svc\_model.score(x\_test,y\_test)

print("SVM accuracy is :",accuracy\_svc)

# %%Naive Bayes Classification

from sklearn.naive\_bayes import GaussianNB

nb\_model = GaussianNB()

nb\_model.fit(x\_train,y\_train)

accuracy\_nb = nb\_model.score(x\_test,y\_test)

print("Naive Bayes accuracy is :",accuracy\_nb)

# %%Decision Tree Classification

from sklearn.tree import DecisionTreeClassifier

dt\_model = DecisionTreeClassifier()

dt\_model.fit(x\_train,y\_train)

accuracy\_dt = dt\_model.score(x\_test,y\_test)

print("Decision Tree accuracy is :",accuracy\_dt)

# %%Random Forest Classification

from sklearn.ensemble import RandomForestClassifier

rf\_model\_initial = RandomForestClassifier(n\_estimators = 5, random\_state = 1)

rf\_model\_initial.fit(x\_train,y\_train)

print("Random Forest accuracy for 5 trees is :",rf\_model\_initial.score(x\_test,y\_test))

score\_array = []

for each in range(1,50):

rf\_loop = RandomForestClassifier(n\_estimators = each, random\_state = 1) #set K neighbor as 3

rf\_loop.fit(x\_train,y\_train)

score\_array.append(rf\_loop.score(x\_test,y\_test))

plt.plot(range(1,50),score\_array)

plt.xlabel("Range")

plt.ylabel("Score")

plt.show()

rf\_model = RandomForestClassifier(n\_estimators = 33, random\_state = 1) #set tree number as 33

rf\_model.fit(x\_train,y\_train)

accuracy\_rf = rf\_model.score(x\_test,y\_test)

print("Random Forest accuracy for 33 trees is :",accuracy\_rf)

# %%Confusion Matrix libraries

from sklearn.metrics import recall\_score, confusion\_matrix, precision\_score, f1\_score, accuracy\_score, classification\_report

#for Logistic Regression

cm\_lr = confusion\_matrix(y\_test,lr\_model.predict(x\_test))

# %% confusion matrix visualization

import seaborn as sns

f, ax = plt.subplots(figsize = (5,5))

sns.heatmap(cm\_lr, annot = True, linewidths = 0.5, color = "red", fmt = ".0f", ax=ax)

plt.xlabel("y\_predicted")

plt.ylabel("y\_true")

plt.title("Confusion Matrix of Logistic Regression")

plt.show()

# the function that prints all scores

def print\_scores(headline, y\_true, y\_pred):

print(headline)

acc\_score = accuracy\_score(y\_true, y\_pred)

print("accuracy: ",acc\_score)

pre\_score = precision\_score(y\_true, y\_pred)

print("precision: ",pre\_score)

rec\_score = recall\_score(y\_true, y\_pred)

print("recall: ",rec\_score)

f\_score = f1\_score(y\_true, y\_pred, average='weighted')

print("f1\_score: ",f\_score)

report = classification\_report(y\_test, lr\_model.predict(x\_test))

print(report)

print\_scores("Logistic Regression;",y\_test, lr\_model.predict(x\_test))

print\_scores("SVC;",y\_test, svc\_model.predict(x\_test))

print\_scores("KNN;",y\_test, knn\_model.predict(x\_test))

print\_scores("Naive Bayes;",y\_test, nb\_model.predict(x\_test))

print\_scores("Decision Tree;",y\_test, dt\_model.predict(x\_test))

print\_scores("Random Forest;",y\_test, rf\_model.predict(x\_test))

#Features Affecting Customer

weights=pd.Series(lr\_model.coef\_[0],index=x.columns.values)

weights.sort\_values(ascending = False)