### VISVESVARAYA TECHNOLOGICAL UNIVERSITY

##### Jnana Sangama, Belagavi-590018, Karnataka



**INTERNSHIP REPORT ON**

## “Machine Learning with Python and IBM Watson”

***Submitted in Partial Fulfillment of requirement for the award of the degree of***

#### Bachelor of Engineering in

**Computer Science & Engineering**

**Submitted by:**

##### NISHCHITH (3GN16CS049)

**Internship Carried Out at**

##### SmartBridge Educational Services Pvt. Ltd.

**Jubilee Hills, Hyderabad - 500092**

Under the guidance of:

##### Prof. Guru Prasad K



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING GURU NANAK DEV ENGINEERING COLLEGE,**

**BIDAR-585403, KARNATAKA**

**GURU NANAK DEV ENGINEERING COLLEGE,**

**BIDAR-585403, KARNATAKA**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the Internship work entitled **“Machine Learning with Python”** carried out by  **NISHCHITH (3GN16CS049**), a bonafide student of Guru Nanak Dev Engineering College in partial fulfillment for the reward of **Bachelor of Engineering** in **Computer Science and Engineering** under **Visvesvaraya Technological University**, **Belagavi** during the academic year **2019-2020** is true representation of Internship work completed satisfactorily.

##### .…………………. …………………… ...............................

**Guide Co-Ordinator HOD**

**EXTERNAL VIVA**

**Examiners: 1) \_ 2)**

**ACKNOWLEDGEMENT**

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without complementing those who made it possible and whose guidance and encouragement made our efforts successful. So, my sincere thanks to all those who have supported me in completing this Internship successfully.

Working in **SmartBridge Educational Services Pvt. Ltd.** has made me to know about the **Machine Learning** domain and **IBM Watson Studio** Cloud Platform tools**.** I wish to express my heartfelt gratitude to my mentor, guide and my team members for their valuable guidance, suggestions and cheerful encouragement during the period of my Internship.

My sincere thanks to **Dr. RAVINDRA EKLARKER**, Principal, GNDEC and **Dr. DAYANAND J**, Head of the Department of Computer Science and Engineering, GNDEC for their encouragement, support and guidance to the student community in all fields of education. I am grateful to our institution for providing us a congenial atmosphere to carry out the Internship successfully.

I would not forget to remember **Prof. STEVENRAJ N**, Professor and Internship Coordinator, for his encouragement and more over for his timely support and guidance till the completion of Internship.

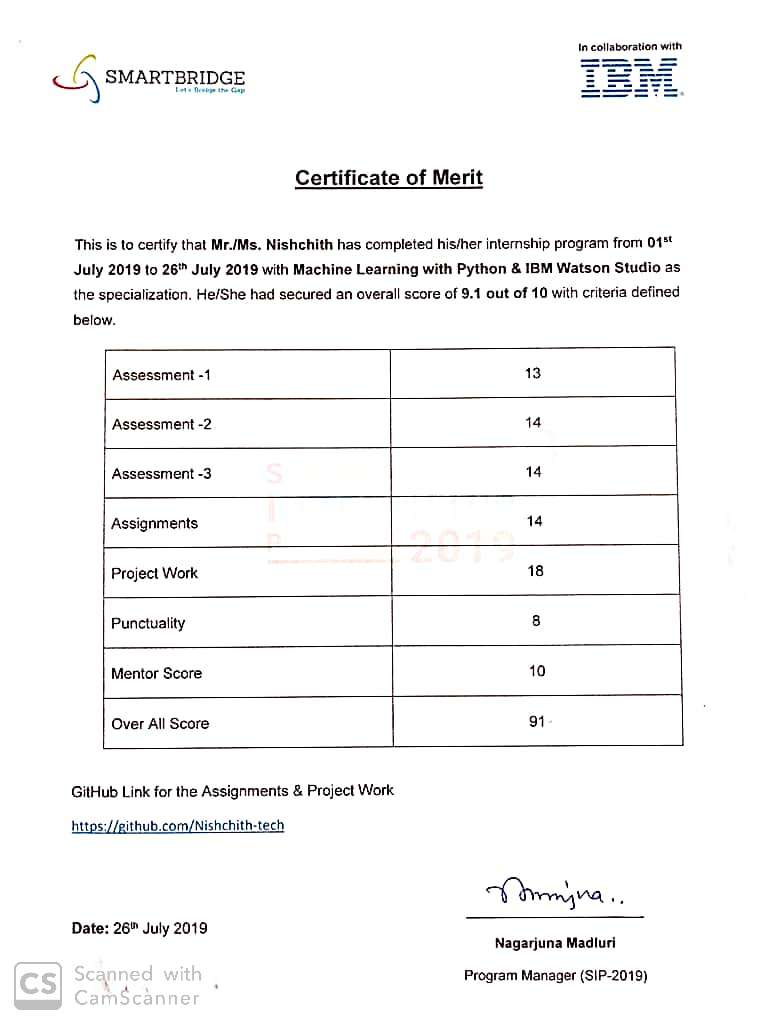
I avail this opportunity to express our profound sense of deep gratitude to my esteemed guide **Prof. GURUPRASAD K**, for his moral support, encouragement and valuable suggestions throughout the Internship.

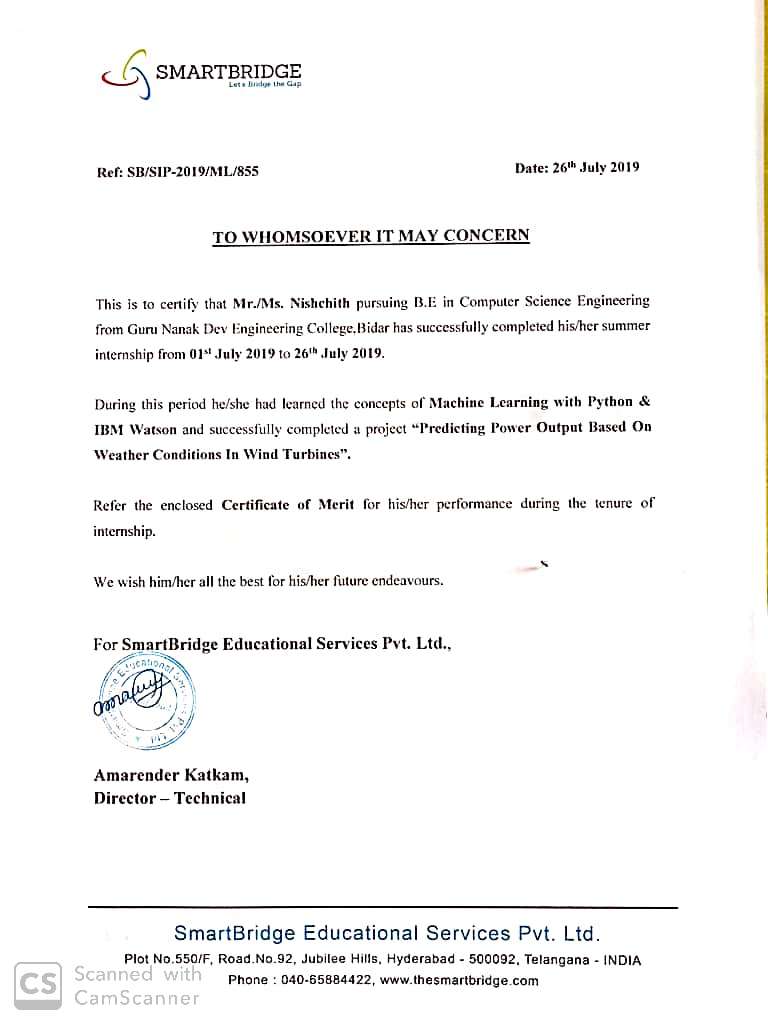
I extend my sincere thanks to my department faculty members of computer science and engineering and also non-teaching staff for supporting me directly or indirectly for the completion of this Internship.

##### NISHCHITH (3GN16CS049)

**INTERNSHIP MERIT CERTIFICATE**

**CERTIFICATE OF MERIT**

****

****

****

****

|  |  |  |
| --- | --- | --- |
| CONTENTS: - | | |
| **Chapter** | **Particulars** | **Page no.** |
| 1. | COMPANY PROFILE | 1 |
| 2. | ABOUT THE COMPANY | 2 |
| 3. | TASK PERFORMED | 3 |
| 4. | MINI-PROJECT | 22 |
| 5. | REFLECTION | 25 |
| 6. | CONCLUSION | 27 |

**CHAPTER 1: COMPANY PROFILE**

1. **COMPANY PROFILE**

|  |  |  |
| --- | --- | --- |
| 1. | **Company Name:** | SmartBridge Educational Services Pvt. Ltd. |
| 2. | **Company Logo:** |  |
| 3. | **Location:** | Hyderabad |
| 4. | **Company Address:** | Plot No 132, Bapuji Nagar, Habsiguda, Nacharam  Main Road, Hyderabad- 500076 |
| 5. | **E-mail ID:** | [info@thesmartbridge.com](mailto:info@thesmartbridge.com) |
| 6. | **Stream:** | Machine Learning with Python and IBM Studio |
| 7. | **Contact No.:** | **+91 8885544987** |
| 8. | **Website:** | <https://www.thesmartbridge.com/> |

**CHAPTER 2: ABOUT THE COMPANY**

1. **ABOUT THE COMPANY**

SMARTBRIDGE is an EdTech organization with a vision to bridge the gap between academia & industry. Our outcome-based experiential learning programs on emerging technologies (Internet of Things, Machine Learning, Data Science, Artificial Intelligence, Robotics) are building skilled entry - level engineers, for the corporate world.

SmartBridge is in mission to build technology communities in academia to encourage students towards innovation & entrepreneurship. Since inception, we have trained thousands of students, faculty and working professionals on emerging technologies via technical bootcamps, hackathons, Summer & Winter Internship Programs.

Smartbridge Educational Services Private Limited has been created with ingenuity to bridge the existing gaps in the transition phase of young graduates to working professionals. Our team works at continuous identification of the changing needs in the corporate sector with respect to the talent ramp up and acquisition.

Ever since the inception, we have shown high end captive results to many young graduates in achieving their goals. Our end goals are simple. Fresh graduates should be adequately equipped in terms of skill and confidence before setting sails to their dream careers with the corporates to achieve their Dreams.

Our ever-evolving strategies go in sync with what the corporates are really looking at with new talent. We empower the young workforce with the same.

# CHAPTER 3: TASK PERFORMED

## TASK PERFORMED

#### WEEK 1: Intensive Hands-on Training

* + 1. **Introduction to Machine Learning:**

The journey of AI began in the 1950's when the computing power was a fraction of what it is today. AI started out with the predictions made by the machine in a fashion a statistician does predictions using his calculator. Thus, the initial entire AI development was based mainly on statistical techniques.

Statistical Techniques The development of today’s AI applications started with using the age-old traditional statistical techniques. There are several other such statistical techniques which are successfully applied in developing so-called AI programs. We say “so-called” because the AI programs that we have today are much more complex and use techniques far beyond the statistical techniques used by the early AI programs.

Some of the examples of statistical techniques that are used for developing AI applications in those days and are still in practice are listed here:

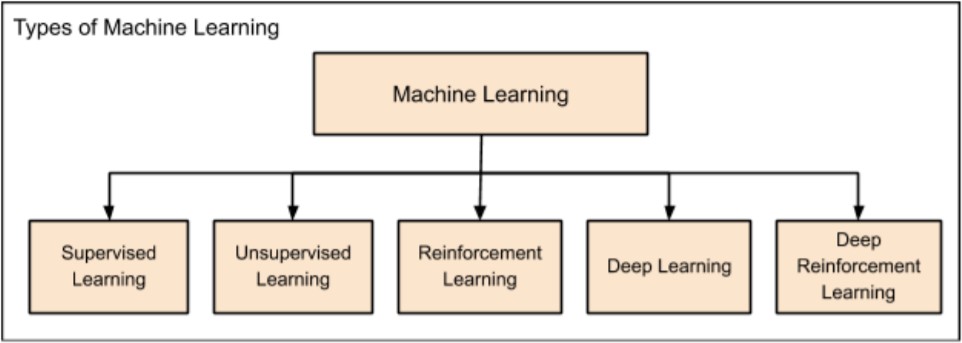
* + - * Regression
      * Classification
      * Clustering
      * Probability Theories
      * Decision Trees

Here we have listed only some primary techniques that are enough to get you started on AI without scaring you of the vastness that AI demands. If you are developing AI applications based on limited data, you would be using these statistical techniques. However, today the data is abundant. To analyse the kind of huge data that we possess statistical techniques are of not much help as they have some limitations of their own. More advanced methods such as deep learning are hence developed to solve many complex problems. As we move ahead in this tutorial, we will understand what Machine Learning is and how it is used for developing such complex AI applications.

Machine Learning with Python and IBM Watson Studio

#### Categories of Machine Learning:

Machine Learning is broadly categorized under the following headings:



Machine learning evolved from left to right as shown in the above diagram.

* + - * Initially, researchers started out with Supervised Learning. This is the case of housing price prediction discussed earlier.
      * This was followed by unsupervised learning, where the machine is made to learn on its own without any supervision.
      * Scientists discovered further that it may be a good idea to reward the machine when it does the job the expected way and there came the Reinforcement Learning.
      * Very soon, the data that is available these days has become so humongous that the conventional techniques developed so far failed to analyse the big data and provide us the predictions.
      * Thus, came the deep learning where the human brain is simulated in the Artificial Neural Networks (ANN) created in our binary computers.
      * The machine now learns on its own using the high computing power and huge memory resources that are available today.
      * It is now observed that Deep Learning has solved many of the previously unsolvable problems.
      * The technique is now further advanced by giving incentives to Deep Learning networks as awards and there finally comes Deep Reinforcement Learning.

Machine Learning with Python and IBM Watson Studio

#### Supervised Learning:

Supervised learning is analogous to training a child to walk. You will hold the child’s hand, show him how to take his foot forward, walk yourself for a demonstration and so on, until the child learns to walk on his own.

##### Regression:

In the case of supervised learning, you give concrete known examples to the computer. You say that for given feature value x1 the output is y1, for x2 it is y2, for x3 it is y3, and so on. Based on this data, you let the computer figure out an empirical relationship between x and y.

Once the machine is trained in this way with a sufficient number of data points, now you would ask the machine to predict Y for a given X. Assuming that you know the real value of Y for this given X, you will be able to deduce whether the machine’s prediction is correct.

Thus, you will test whether the machine has learned by using the known test data. Once you are satisfied that the machine is able to do the predictions with a desired level of accuracy (say 80 to 90%) you can stop further training the machine.

Now, you can safely use the machine to do the predictions on unknown data points, or ask the machine to predict Y for a given X for which you do not know the real value of Y. This training comes under the regression that we talked about earlier.

##### Classification:

We may also use machine learning techniques for classification problems. In classification problems, you classify objects of similar nature into a single group. For example, in a set of 100 students say, you may like to group them into three groups based on their heights - short, medium and long. Measuring the height of each student, you will place them in a proper group.

Now, when a new student comes in, you will put him in an appropriate group by measuring his height. By following the principles in regression training, you will train the machine to classify a student based on his feature – the height. When the machine learns how the groups are formed, it will be able to classify any unknown new student correctly. Once again, you would use the test data to verify that the

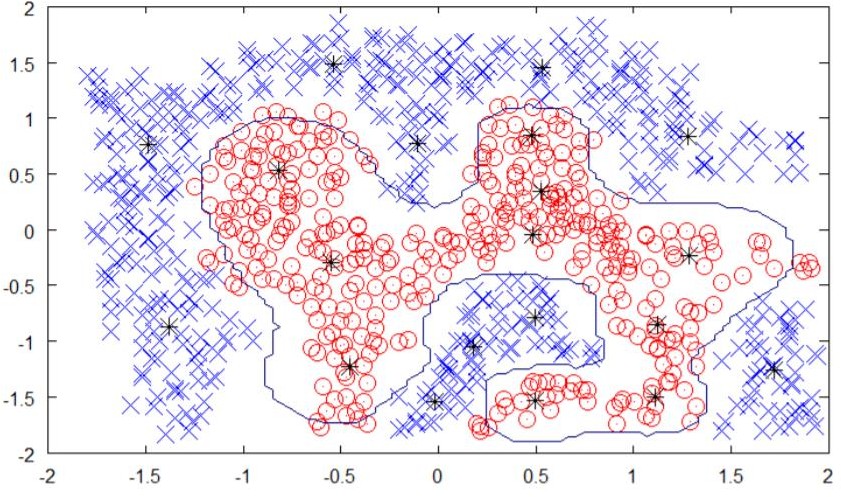
Machine Learning with Python and IBM Watson Studio

machine has learned your technique of classification before putting the developed model in production.

Supervised Learning is where the AI really began its journey. This technique was applied successfully in several cases. You have used this model while doing the hand-written recognition on your machine. Several algorithms have been developed for supervised learning.

##### Unsupervised Learning:

In unsupervised learning, we do not specify a target variable to the machine, rather we ask machine “What can you tell me about X?”. More specifically, we may ask questions such as given a huge data set X, “What are the five best groups we can make out of X?” or “What features occur together most frequently in X?”. To arrive at the answers to such questions, you can understand that the number of data points that the machine would require to deduce a strategy would be very large. In case of supervised learning, the machine can be trained with even about few thousands of data points. However, in case of unsupervised learning, the number of data points that is reasonably accepted for learning starts in a few millions. These days, the data is generally abundantly available. The data ideally requires curating. However, the amount of data that is continuously flowing in a social area network, in most cases data curation is an impossible task.

The following figure shows the boundary between the yellow and red dots as determined by unsupervised machine learning. You can see it clearly that the machine would be able to determine the class of each of the black dots with a fairly good accuracy.

The unsupervised learning has shown a great success in many modern AI applications, such as face detection, object detection, and so on.

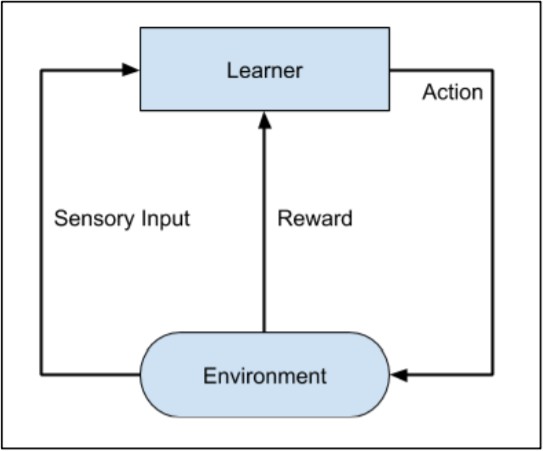
Machine Learning with Python and IBM Watson Studio

##### Reinforcement Learning:

Consider training a pet dog, we train our pet to bring a ball to us. We throw the ball at a certain distance and ask the dog to fetch it back to us. Every time the dog does this right, we reward the dog. Slowly, the dog learns that doing the job rightly gives him a reward and then the dog starts doing the job right way every time in future. Exactly, this concept is applied in “Reinforcement” type of learning. The technique was initially developed for machines to play games. The machine is given an algorithm to analyze all possible moves at each stage of the game. The machine may select one of the moves at random.

If the move is right, the machine is rewarded, otherwise it may be penalized. Slowly, the machine will start differentiating between right and wrong moves and after several iterations would learn to solve the game puzzle with a better accuracy. The accuracy of winning the game would improve as the machine plays more and more games.

The entire process may be depicted in the following diagram:



This technique of machine learning differs from the supervised learning in that you need not supply the labelled input/output pairs. The focus is on finding the balance between exploring the new solutions versus exploiting the learned solutions.

Machine Learning with Python and IBM Watson Studio

##### Deep Learning:

The deep learning is a model based on Artificial Neural Networks (ANN), more specifically Convolutional Neural Networks (CNN)s. There are several architectures used in deep learning such as deep neural networks, deep belief networks, recurrent neural networks, and convolutional neural networks.

These networks have been successfully applied in solving the problems of computer vision, speech recognition, natural language processing, bioinformatics, drug design, medical image analysis, and games. There are several other fields in which deep learning is proactively applied. The deep learning requires huge processing power and humongous data, which is generally easily available these days.

##### Deep Reinforcement Learning:

The Deep Reinforcement Learning (DRL) combines the techniques of both deep and reinforcement learning. The reinforcement learning algorithms like Q-learning are now combined with deep learning to create a powerful DRL model. The technique has been with a great success in the fields of robotics, video games, finance and healthcare. Many previously unsolvable problems are now solved by creating DRL models. There is lots of research going on in this area and this is very actively pursued by the industries.

Machine Learning with Python and IBM Watson Studio

##### Python Programming:

* + 1. **Data types**

Python has five standard Data Types:

* + - * Numbers
      * String
      * List
      * Tuple
      * Dictionary

Python sets the variable type based on the value that is assigned to it. Unlike more riggers languages, Python will change the variable type if the variable value is set to another value. For example:

var = 123 # This will create a number integer assignment var = 'john' # the `var` variable is now a string type.

##### Numbers

Python numbers variables are created by the standard Python method:

var = 382

Most of the time using the standard Python number type is fine. Python will automatically convert a number from one type to another if it needs. But, under certain circumstances that a specific number type is needed (i.e. complex, hexadecimal), the format can be forced into a format by using additional syntax in the table below:

|  |  |  |
| --- | --- | --- |
| **Type** | **Format** | **Description** |
| int | a = 10 | Signed Integer |
| long | a = 345L | (L) Long integers, they can also be represented in octal and hexadecimal |
| float | a = 45.67 | (.) Floating point real values |
| complex | a = 3.14J | (J) Contains integer in the range 0 to 255. |

Most of the time Python will do variable conversion automatically. You can also use Python conversion functions (int(), long(), float(), complex()) to convert data from one type to another. In addition, the type function returns information about how your data is stored within a variable.

Machine Learning with Python and IBM Watson Studio

message = "Good morning" num = 85

pi = 3.14159

print(type(message)) # This will return a string print(type(n)) # This will return an integer print(type(pi)) # This will return a float

##### String

Create string variables by enclosing characters in quotes. Python uses single quotes ' double quotes " and triple quotes """ to denote literal strings. Only the triple quoted strings """ also will automatically continue across the end of line statement.

firstName = 'john' lastName = "smith"

message = """This is a string that will span across multiple lines. Using newline characters

and no spaces for the next lines. The end of lines within this string also count as a newline when printed"""

Strings can be accessed as a whole string, or a substring of the complete variable using brackets ‘[]’. Here are a couple examples:

var1 = 'Hello World!' var2 = 'RhinoPython'

print var1[0] # this will print the first character in the string an `H` print var2[1:5] # this will print the substring 'hinoP`

Python can use a special syntax to format multiple strings and numbers. The string formatter is quickly covered here because it is seen often and it is important to recognize the syntax.

print "The item {} is repeated {} times".format(element,count))

Machine Learning with Python and IBM Watson Studio

The {} are placeholders that are substituted by the variables element and count in the final string. This compact syntax is meant to keep the code more readable and compact.

Python is currently transitioning to the format syntax above, but python can use an older syntax, which is being phased out, but is still seen in some example code:

print "The item %i is repeated %i times"% (element,count)

##### List

Lists are a very useful variable type in Python. A list can contain a series of values. List variables are declared by using brackets [ ] following the variable name.

A = [ ] # This is a blank list variable

B = [1, 23, 45, 67] # this list creates an initial list of 4 numbers. C = [2, 4, 'john'] # lists can contain different variable types.

All lists in Python are zero-based indexed. When referencing a member or the length of a list the number of list elements is always the number shown plus one.

mylist = ['Rhino', 'Grasshopper', 'Flamingo', 'Bongo']

B = len(mylist) # This will return the length of the list which is 3. The index is 0, 1, 2, 3.

print mylist[1] # This will return the value at index 1, which is 'Grasshopper' print mylist[0:2] # This will return the first 3 elements in the list.

You can assign data to a specific element of the list using an index into the list. The list index starts at zero. Data can be assigned to the elements of an array as follows:

mylist = [0, 1, 2, 3] mylist[0] = 'Rhino' mylist[1] = 'Grasshopper' mylist[2] = 'Flamingo' mylist[3] = 'Bongo'

print mylist[1]

Machine Learning with Python and IBM Watson Studio

Lists aren’t limited to a single dimension. Although most people can’t comprehend more than three or four dimensions. You can declare multiple dimensions by separating an with commas. In the following example, the MyTable variable is a two-dimensional array :

MyTable = [[], []]

In a two-dimensional array, the first number is always the number of rows; the second number is the number of columns.

##### TUPLE

Tuples are a group of values like a list and are manipulated in similar ways. But, tuples are fixed in size once they are assigned. In Python the fixed size is considered immutable as compared to a list that is dynamic and mutable. Tuples are defined by parenthesis ().

myGroup = ('Rhino', 'Grasshopper', 'Flamingo', 'Bongo')

Here are some advantages of tuples over lists:

* + - * 1. Elements to a tuple. Tuples have no append or extend method.
        2. Elements cannot be removed from a tuple.
        3. You can find elements in a tuple, since this doesn’t change the tuple.
        4. You can also use the in operator to check if an element exists in the tuple.
        5. Tuples are faster than lists. If you’re defining a constant set of values and all you’re ever going to do with it is iterate through it, use a tuple instead of a list.
        6. It makes your code safer if you “write-protect” data that does not need to be changed.

It seems tuples are very restrictive, so why are they useful? There are many data structures in Rhino that require a fixed set of values. For instance, a Rhino point is a list of 3 numbers [34.5, 45.7, 0]. If this is set as tuple, then you can be assured the original 3 number structure stays as a point (34.5, 45.7, 0). There are other data structures such as lines, vectors, domains and other data in Rhino that also require a certain set of values that do not change. Tuples are great for this.

Machine Learning with Python and IBM Watson Studio

##### DICTIONARY

Dictionaries in Python are lists of Key: Value pairs. This is a very powerful datatype to hold a lot of related information that can be associated through keys. The main operation of a dictionary is to extract a value based on the key name. Unlike lists, where index numbers are used, dictionaries allow the use of a key to access its members. Dictionaries can also be used to sort, iterate and compare data.

Dictionaries are created by using braces ({}) with pairs separated by a comma (,) and the key values associated with a colon (:). In Dictionaries the Key must be unique. Here is a quick example on how dictionaries might be used:

room\_num = {'john': 425, 'tom': 212}

room\_num['john'] = 645 # set the value associated with the 'john' key to 645 print (room\_num['tom']) # print the value of the 'tom' key. room\_num['isaac'] = 345 # Add a new key 'isaac' with the associated value print (room\_num.keys()) # print out a list of keys in the dictionary

print ('isaac' in room\_num) # test to see if 'issac' is in the dictionary. This returns true.

Dictionaries can be more complex to understand, but they are great to store data that is easy to access.

Machine Learning with Python and IBM Watson Studio

* + 1. **Python Libraries**

##### NUMPY

Numpy is a general-purpose array-processing package. It provides a high- performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

* + - * + A powerful N-dimensional array object
        + Sophisticated (broadcasting) functions
        + Tools for integrating C/C++ and Fortran code
        + Useful linear algebra, Fourier transform, and random number capabilities Besides its obvious scientific uses, Numpy can also be used as an efficient multi- dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases.

##### PANDAS

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem.

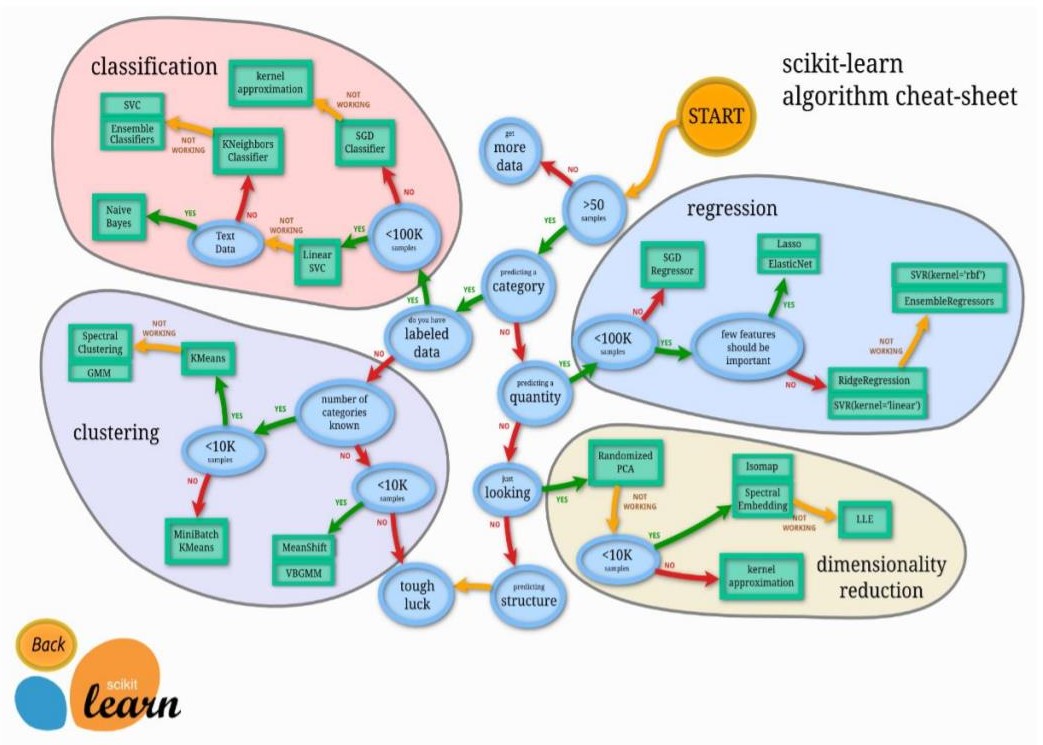
Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

##### MATPLOTLIB

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and [IPython](http://ipython.org/) shells, the [Jupyter](http://jupyter.org/) notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the [sample plots](https://matplotlib.org/tutorials/introductory/sample_plots.html) and [thumbnail gallery](https://matplotlib.org/gallery/index.html). For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc., via an object-oriented interface or via a set of functions familiar to MATLAB users.

Machine Learning with Python and IBM Watson Studio

### Sci-Kit Learn



This standard library has many pre-built classes of various Data-Science Algorithms which provide the ready-to-use implementation of these algorithms. The use of these algorithms is trivial and since these are well and field tested, you can safely use them in your AI applications. Most of these libraries are free to use even for commercial purposes.

#### WEEK 2

##### Supervised Learning Algorithms:

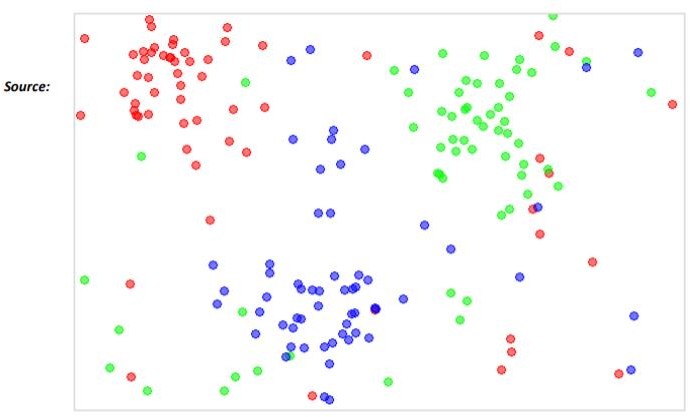
Some of the widely used algorithms of supervised learning are as shown below:

* k-Nearest Neighbours
* Decision Trees
* Naive Bayes
* Logistic Regression
* Support Vector Machines
* Linear/Multi-Linear Regression

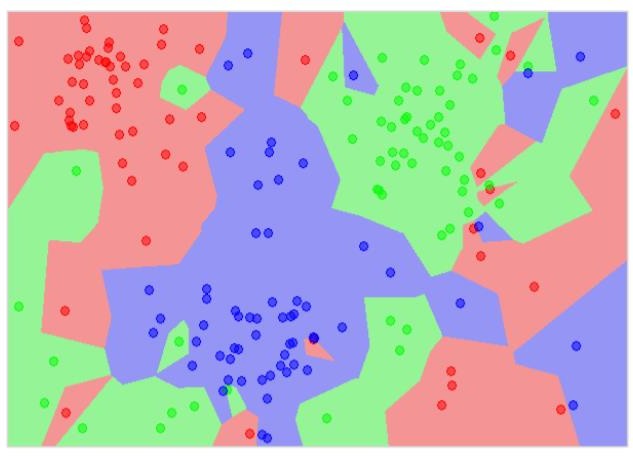
##### k-Nearest Neighbours:

The k-Nearest Neighbours, which is simply called kNN is a statistical technique that can be used for solving for classification and regression problems.

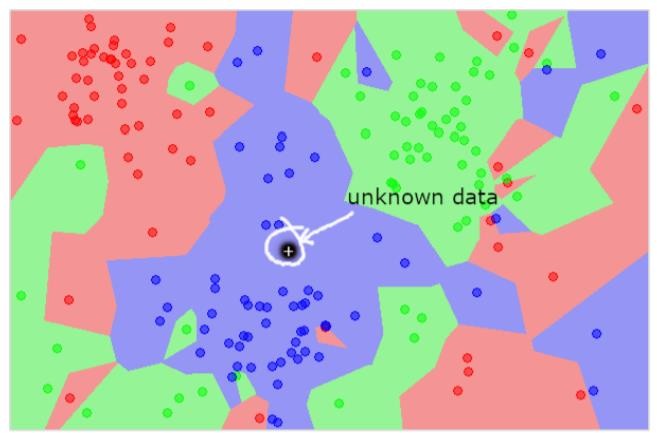
Consider the distribution of objects as shown in the image given below:



The diagram shows three types of objects, marked in red, blue and green colors. When you run the kNN classifier on the above dataset, the boundaries for each type of object will be marked as shown below:



Now, consider a new unknown object that you want to classify as red, green or blue. This is depicted in the figure below.



As you see it visually, the unknown data point belongs to a class of blue objects. Mathematically, this can be concluded by measuring the distance of this unknown point with every other point in the data set. When you do so, you will know that most of its neighbours are of blue color.

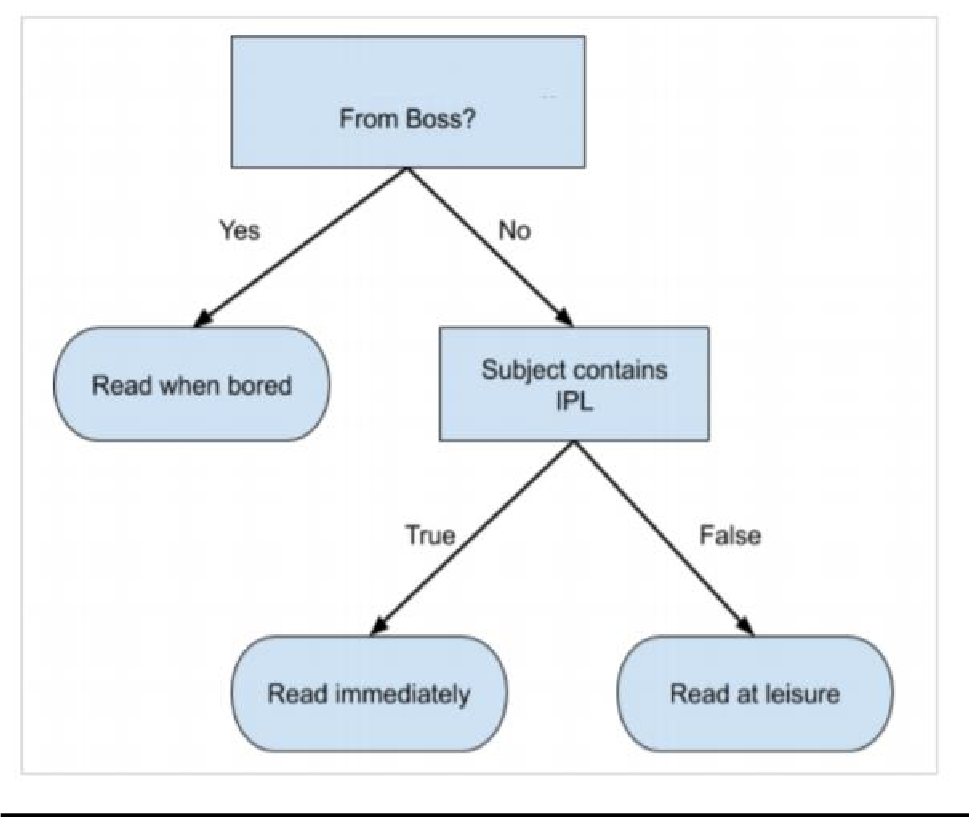
The average distance to red and green objects would be definitely more than the average distance to blue objects. Thus, this unknown object can be classified as belonging to blue class. The kNN algorithm can also be used for regression problems. The kNN algorithm is available as ready-to-use in most of the ML libraries like Sci-Kit Learn.

##### Decision Trees

A tree has many analogies in real life, and turns out that it has influenced a wide area of **machine learning**, covering both **classification and regression**. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal, its also widely used in machine learning

You would write a code to classify your input data based on this flowchart. The flowchart is self-explanatory and trivial. In this scenario, you are trying to classify an incoming email to decide when to read it.

In reality, the decision trees can be large and complex. There are several algorithms available to create and traverse these trees. As a Machine Learning enthusiast, you need to understand and master these techniques of creating and traversing decision trees.



Machine Learning with Python and IBM Watson Studio

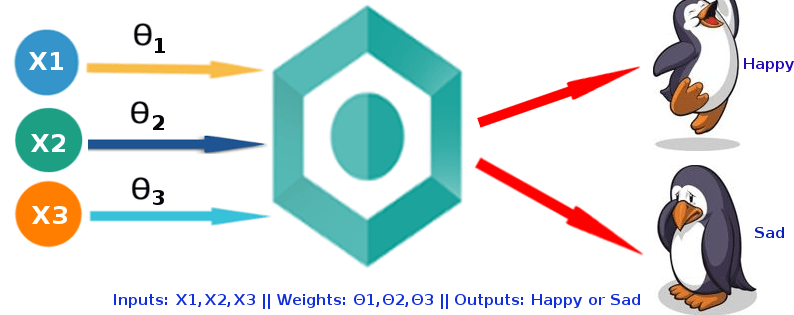
##### Naïve Bayes:

Naive Bayes is used for creating classifiers. Suppose you want to sort out (classify) fruits of different kinds from a fruit basket. You may use features such as color, size and shape of a fruit, for example, any fruit that is red in color, is round in shape and is about 10 cm in diameter may be considered as Apple.

So, to train the model, you would use these features and test the probability that a given feature matches the desired constraints. The probabilities of different features are then combined to arrive at a probability that a given fruit is an Apple. Naive Bayes generally requires a small number of training data for classification.

Also, it is a simple, yet effective and commonly-used, machine learning classifier. It is a probabilistic classifier that makes classifications using the Maximum A Posteriori decision rule in a Bayesian setting. It can also be represented using a very simple Bayesian network. Naive Bayes classifiers have been especially popular for text classification, and are a traditional solution for problems such as spam detection.

##### Logistic Regression:



Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications. Logistic Regression is used when the dependent variable(target) is categorical.

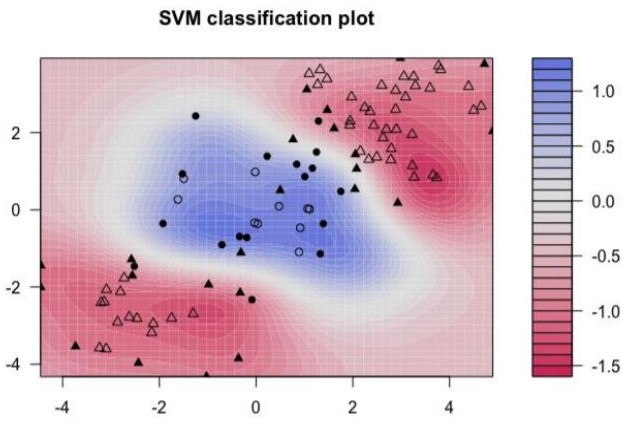
Machine Learning with Python and IBM Watson Studio

For example,

* + - * To predict whether an email is spam (1) or (0)
      * Whether the tumor is malignant (1) or not (0)

Consider a scenario where we need to classify whether an email is spam or not. If we use linear regression for this problem, there is a need for setting up a threshold based on which classification can be done. Say if the actual class is malignant, predicted continuous value 0.4 and the threshold value is 0.5, the data point will be classified as not malignant which can lead to serious consequence in real time. From this example, it can be inferred that linear regression is not suitable for classification problem. Linear regression is unbounded, and this brings logistic regression into picture. Their value strictly ranges from 0 to 1.

##### 3.3.4 Support Vector Machines:

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. The distribution of data. Here the three classes of data cannot be linearly separated. The boundary curves are non- linear. In such a case, finding the equation of the curve becomes a complex job.

The Support Vector Machines (SVM) comes handy in determining the separation boundaries in such situations.

### Introduction to IBM Watson Studio and NodeRed

The IBM® Watson™ Studio learning path demonstrates various ways of using IBM Watson Studio to predict customer churn. It ranges from a semi-automated approach using the AutoAI Experiment tool to a diagrammatic approach using SPSS Modeler Flows to a fully programmed style using Jupyter notebooks for Python.

For all of the tasks we use [IBM Watson Studio](https://dataplatform.cloud.ibm.com/docs/content/getting-started/welcome-main.html). It gives us the environment and tools to solve business problems by collaboratively working with data. You can choose the tools needed to analyze and visualize data; to cleanse and shape the data; to ingest streaming data; or to create, train, and deploy machine learning models.

##### Uses of IBM Watson Studio:

* Create [projects](https://dataplatform.cloud.ibm.com/docs/content/wsj/manage-data/manage-projects.html?audience=wdp) to organize the resources (such as data connections, data assets, collaborators, and notebooks) to achieve an analytics goal.
* Access data from [connections](https://dataplatform.cloud.ibm.com/docs/content/wsj/getting-started/assets.html#connection) to your cloud or on-premises data sources.
* [Upload files](https://dataplatform.cloud.ibm.com/docs/content/wsj/manage-data/add-data-project.html) to the project’s object storage.
* Create and maintain [data catalogs](https://dataplatform.cloud.ibm.com/docs/content/wsj/catalog/manage-catalog.html) to discover, index, and share data.
* [Refine data](https://dataplatform.cloud.ibm.com/docs/content/wsj/refinery/refining_data.html) by cleansing and shaping the data to prepare it for analysis.
* Perform [data science](https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/data-science.html) tasks by creating Jupyter Notebooks for Python or Scala to run code that processes data and then view the results inline. Alternatively, you can use RStudio for R.
* Ingest and [analyze streams](https://dataplatform.cloud.ibm.com/docs/content/wsj/streaming-pipelines/overview-streaming-pipelines.html) data with the Streams Designer tool.
* Create, test, and deploy [machine learning](https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/ml-overview.html) and **deep learning** models.
* [Classify images](https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/visual-recognition-overview.html) by training deep learning models to recognize image content.
* Create and share [dashboards](https://dataplatform.cloud.ibm.com/docs/content/wsj/getting-started/assets.html#dashboard) of data visualizations without coding.

Technically, IBM Watson Studio is based on a variety of open source technology and IBM products as shown in the following figure. In the context of data science, IBM Watson Studio can be viewed as an integrated, multi-role collaboration platform that supports the developer, data engineer, business analyst, and the data scientist in the process of solving a data science problem. For the developer role, other components of the IBM Cloud platform might be relevant as well in building applications that use machine learning services.

# CHAPTER 4: MINI-PROJECT

### WEEK 3 & 4:

1. **Mini-Project and Hackathon**

**Title:** Predicting Power Output Based on Weather Conditions in Wind Turbines

##### ABOUT THE PROJECT:

##### windmills

Wind energy is eco-friendly, it does not emit harmful gases into the atmosphere like CO2, CO and many more.

Carbon-free technologies like renewable energy help combat climate change, but many of them have not reached their full potential.

Consider wind power: over the past decade, wind farms have become an important source of carbon-free electricity as the cost of turbines has plummeted and adoption has surged.

However, the variable nature of wind itself makes it an unpredictable energy source—less useful than one that can reliably deliver power at a set time.

Wind power output is based on the attributes such as wind energy, wind speed, wind direction and date/time.

Based on these predictions, our model recommends how to make optimal hourly delivery commitments to the power grid a full day in advance. This is important, because energy sources that can be scheduled (i.e. can deliver a set amount of electricity at a set time) are often more valuable to the grid.

Multiple linear regression model to build our project. Multiple linear regression is such a technique and the most commonly used short term prediction method, which will predict power output based on weather conditions. This method gives less prediction error.

The main objective of our project is to predict the wind power output generated based on the attributes given in the datasets which include wind speed, wind direction and constants.

##### PROBLEM STATEMENT:

**Fossil fuels have been used extensively all over the world to satisfy energy demands. However, their availability is limited and their negative impact on the environment undeniable. Fossil fuels increases global warming and emit harmful gases in the atmosphere.**

**The Solution:** Our goal is to predict power output based on weather conditions factors like wind speed, theoretical power constants, temperature, direction. The alternative source of energy i.e wind energy produces power considering all the factors thereby reducing pollution, global warming and many more harmful things on environment.

##### DATA COLLECTION:

### Context

In Wind Turbines, The System’s measure save data's like wind speed, wind direction, generated power etc. for 10 minutes intervals. This file was taken from a wind turbine’s system that is working and generating power in Turkey.

### Content

The data's in the file are:

* Date/Time (for 10 minutes intervals)
* LV ActivePower (kW): The power generated by the turbine for that moment
* Wind Speed (m/s): The wind speed at the hub height of the turbine (the wind speed that turbine use for electricity generation)
* Theoretical\_Power\_Curve (KWh): The theoretical power values that the turbine generates with that wind speed which is given by the turbine manufacturer
* Wind Direction (°): The wind direction at the hub height of the turbine (wind turbines turn to this direction automatically)

Machine Learning with Python and IBM Watson Studio

**Figures and Tables**

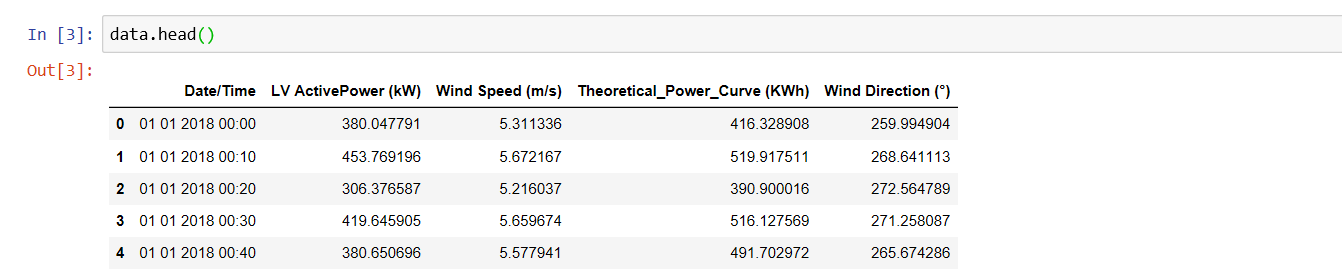


Fig: Sample Dataset

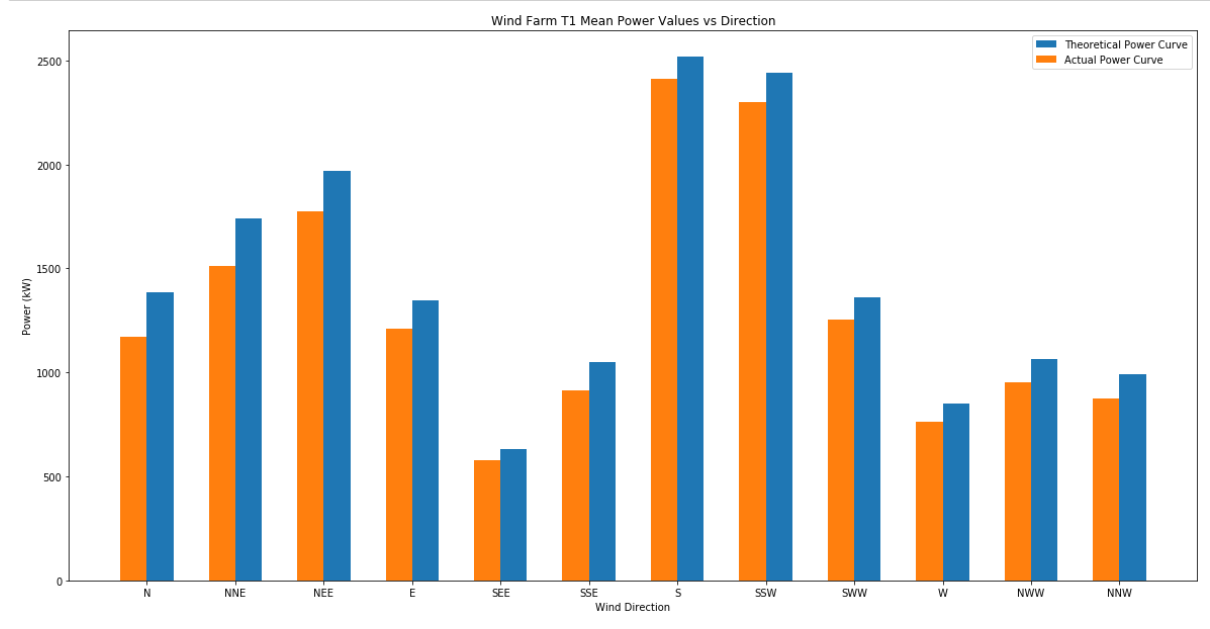


Fig: Wind Direction vs Power Curve

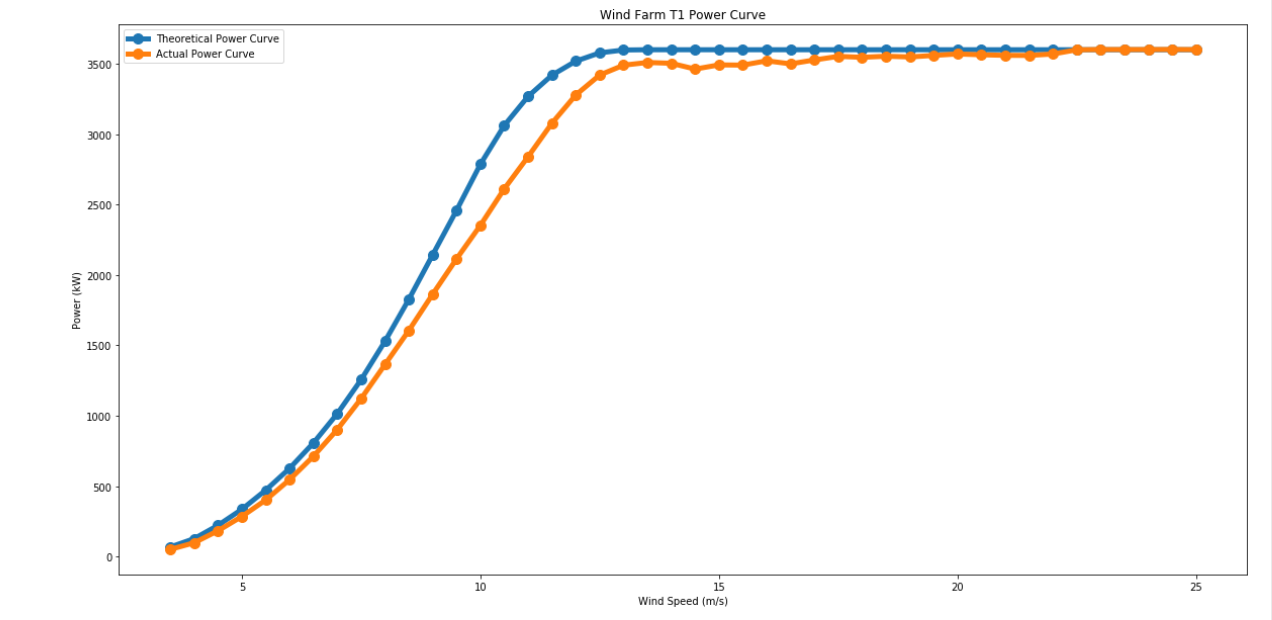


Fig: Wind Speed vs Power Curve

##### DATA MODELLING:

The data model that we have used in our project is “Multiple Linear Regression” model.

Multiple Linear Regression attempts to model the relationship between two or more features and a response by fitting a linear equation to observed data.

Multiple Linear Regression is a simple and common way to analyse linear regression. The model is often used for predictive analysis since it defines the relationship between two or more variables.

The steps to perform multiple linear regression is almost similar to that of simple linear Regression.

The Difference lies in the evaluation. We can use it to find out which factor has the highest impact on the predicted output and now different variable relate to each other.

*Here:****Y=b0+b1\*x1+b2\*x2+b3\*x3+……bn\*xn*** *Y = Dependent variable and x1, x2, x3, …… xn = multiple independent variables*

The dataset collected is tested using all the regression models and the one which gives maximum accuracy is selected. For our dataset multiple linear regression gives 90% accuracy so this model is selected.

##### 4.4.1 MODEL ACCURACY:

##### 

* 1. **OUTPUT:**

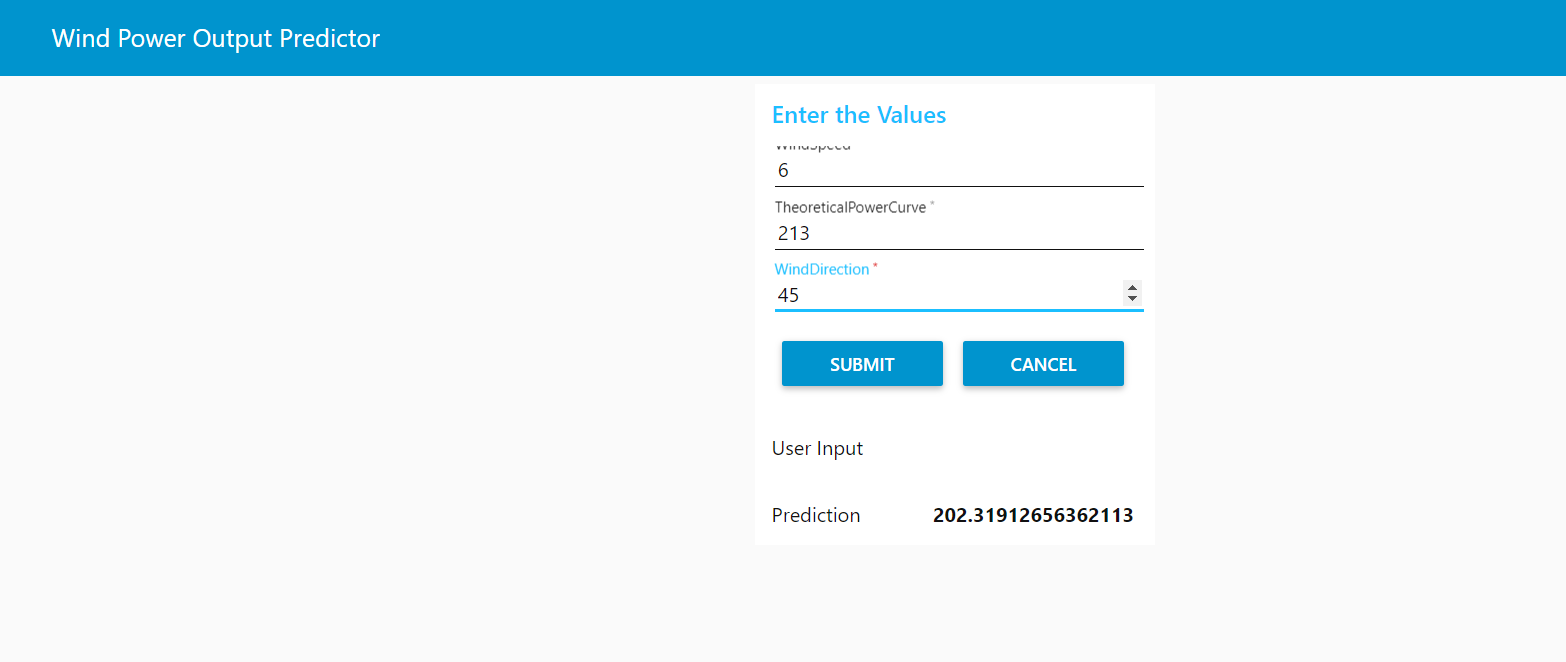
****

Fig: Front End Using NODERED

**CHAPTER 5: REFLECTION**

1. **REFLECTION**

* The First two weeks of the Internship consisted of various Intensive Hands- on Training on the Machine Learning. Basic Crash Course on the Python programming language required for the Data Modelling was also the part of this session.

Creating a Data Model using various Regression and Classification Models, Designing the Front-end UI using NodeRed and managing all these data over the cloud using IBM Watson Studio was the highlight of this training.

* Third Week consisted of the Implementation of the various Algorithms over the IBM Cloud and Developing a Mini-Project using these tools was done.
* In the Last week of the Internship, the Project was presented to the audience along with our Project team members.

A Developer Challenge was also conducted at the end of the Internship involving a task to be completed using the learnt skills to tackle the real- world issue using Machine Learning Techniques.

### Roles and Responsibilities:

* + - Working on programming-based projects.
    - Understanding of fundamental design principles behind an application.
    - Understanding project requirements, and develop the application modules to meet the requirements as provided in the design documents
    - Maintaining code quality.

### Skills Developed:

* + - Machine Learning Fundamentals.
    - M.L Algorithms and its implementation.
    - Python Programming for Data Science.
    - Teamwork, Friendliness, Professional Ethics.
    - Software Engineering Principles
    - Self-Management.
    - Time-Management
    - Communication Skills,

# CHAPTER 6: CONCLUSION

### CONCLUSION

This internship has been an excellent and rewarding experience. It enabled me to give new insight and more comprehensive understanding about the real information. I acquired knowledge concerning my technical skills, software installation, project development and other tasks provided me to develop and improve my skills. I grew more independent in work and also in everyday life.

This internship has helped in reducing the level of difficulty while developing machine learning projects. Other than the technical perspective the soft skills that I learnt also will surely help me in my future professional career like working in teams, meeting client deadlines effectively and other work ethics.