

Application Of Machine Learning In Data Processing

M.Sc Thesis

by

Ravi Shankar



**Discipline of Physics, Indian Institute of Technology
Indore,
Khandwa Road, Simrol, Indore - 453552, India**

Application Of Machine Learning In Data Processing

A THESIS

*Submitted in partial fulfillment of the
requirements for the award of the degree*

of

MASTERS OF SCIENCE

by

Ravi Shankar



Discipline of Physics, Indian Institute of Technology

Indore,

Khandwa Road, Simrol, Indore - 453552, India

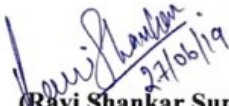


INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **APPLICATION OF MACHINE LEARNING IN DATA PROCESSING** in the partial fulfillment of the requirements for the award of the degree of **MASTER OF SCIENCE** and submitted in the **DISCIPLINE OF PHYSICS, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from **JULY 2018** to **JUNE 2019** under the supervision of **Dr. Ankhi Roy**, Associate Professor, Indian Institute of Technology Indore.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

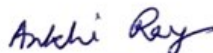

(Ravi Shankar Suman)

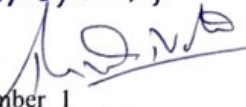
This is to certify that the above statement made by the candidate is correct to the best of my/our knowledge.


M.Sc. Thesis Supervisor

(Dr. Ankhi Roy)

Ravi Shankar Suman has successfully given his M.Sc. Oral Examination held on 21/06/2019


M.Sc. Thesis Supervisor
Date: 28/06/2019


PSPC Member 1
Date: 01/7/19


Convener, DPGC
Date: 01/07/19


PSPC Member 2
Date: 21/7/2019

Dedicated
to
My Family

Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisor Dr. Ankhi Roy for the continuous support of my M.Sc study and research, for her patience, motivation, enthusiasm, and immense knowledge.

Besides my advisor, I would like to thank the rest of my thesis PSPC members: Dr. Manavendra Mahato and Dr. Manoneeta Chakraborty for their encouragement and support.

I thank my fellow labmates, Mr. Sudhir Rode, Mr. Sumit Kundu, Mr. Ravinder Singh, Mr. Hridey Chetri, and Mr. Prasoon Chakraborty, for being together during my research work and for all the fun we have had in the last one year.

I would also like to thank: Mr. Piyush Kalra, Mr. Pavish, Miss. Swati Malhotra, Mr. Ashish Bisht, Mr. Deepak, Mr. Rahul Lamba for encouragement and emotional support.

In particular, I am grateful to Mr. Naveen Maindola, who has helped me a lot in every way.

I would like to thank my family: My father, mother, brother, and especially my sister who gave me a reason to live and supporting me spiritually throughout my life.

Last but not least, I would like to thank Miss. Nidhi Kashyap to stay with me always in every situation.

Thanks for all your encouragement!

Abstract

The main objective of this thesis is to present the Machine learning(ML) techniques and their algorithms briefly; Chapter 1 outlines the introduction about ML and its branches and the main reason why statistical agencies should start exploring the use of machine learning techniques. The goal of machine learning is to program computers to use example data or past experience to solve a given problem. Chapter 2 outlines the analysis of the most popular algorithm that is Support Vector Machines, which has been used to solve classification task under supervised learning. Chapter 3 describes the Result obtained from high energy and Diabetes dataset using the SVM algorithm, LibSvm is used to generate the results and plots. The watershed algorithm is used for image processing and finding the clusters in the diffraction pattern of a protein crystal. Last section covers the application of machine learning in the field of science, medical science, social media and many more.

Contents

Title	1
Acknowledgements	v
Abstract	vii
List of abbreviations	xii
1 Introduction	1
1.1 Supervised machine learning	2
1.2 Unsupervised machine learning	4
2 Theory And Tools	5
2.1 Support Vector Machines	5
2.2 Hypothesis formulation	6
2.3 Regression	7
2.4 Classification	8
2.5 Regularization	10
2.6 V-Fold Cross Validation	10
2.7 LibSvm	12
2.8 Unsupervised learning and image processing	12
2.9 Working with DBSCAN	13
2.10 Diffraction Pattern of a Protien Crystal	14
3 Analysis And Results	15
3.1 Prerequisites	15
3.2 High energy dataset:	16
3.3 Diabetes dataset:	17
3.4 Diffraction pattern of protien crystal	18
4 Application Of Machline Learning	21
Appendices	25

A HIGH ENERGY DATASET	27
B DIABETES DATASET	31

List of Figures

1.1	Brances of machine learning[2]	2
2.1	svm mode : hypothesis is a line in 2D and plane in 3D.[4]	6
2.2	cost function[$J(\theta)$] vs parameters(θ)	8
2.3	Five fold cross validation process.[6]	11
2.4	Different steps for image processing	12
2.5	shows the steps of forming cluster.[9]	13
2.6	Given image of diffraction pattern of protien crystal.[10]	14
3.1	cross validation result for best \mathbf{C} and γ	17
3.2	Parameter search using k-nearest neighbour for best eps value	19
3.3	clusters generated using DBSCAN	19

List of abbreviations

SVM

SupportVectorMachine

Chapter 1

Introduction

Machine learning is a branch of artificial intelligence that allows computer systems to learn directly from examples, data, and experience. Nowadays many people interact with systems based on machine learning every day, for example, recommended systems, such as used in social media, streaming websites and online shopping websites where our computers automatically show the item of our interest, email filtration where it automatically filters the spam emails from the working directory and many more. As this field develops further, machine learning shows promise of supporting potentially transformative advances in a range of areas, and the social and economic opportunities which follow are significant. In healthcare, machine learning is creating systems that can help doctors give more accurate or effective diagnoses for certain conditions such as diabetes diagnosis, cancer treatment, and many other severe diseases. In transport, it is supporting the development of autonomous vehicles such as Tesla- the self-driving car, and helping to make existing transport networks more efficient. For public services, it has the potential to target support more effectively to those in need, or to tailor services to users. And in science, machine learning is helping to make sense of a large amount of data available to researchers today, offering new insights into biology, physics, medicine, the social sciences, and more.

Basically there are two major branches of machine learning:

- a) Supervised machine learning b) Unsupervised machine learning

Apart from the above two, there are Reinforcement and Semi-supervised learning, which are least used by the users and industries. Machine learning is about getting machines to learn from past data and then make decisions on similar data. It is about using several predictive algorithms to forecast the behavior of data so that calculated decisions can be taken. Machine learning algorithms are built on statistical features. Its growing popularity is primarily due to an increase in data availability and advancements in technology.[1]

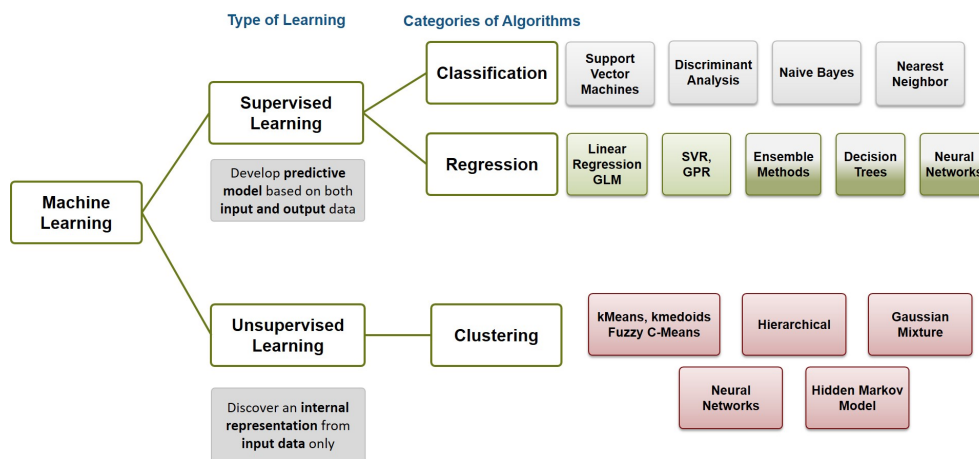


Figure 1.1: Branches of machine learning[2]

Supervised machine learning

It is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples. In this, each example is a pair consisting of an input object and the desired output value. Supervised learning is a type of Machine Learning that enables the model to predict outcomes after they are trained based on past data. Training data for supervised learning includes a set of examples with paired input subjects and the desired output (which is also referred to as the supervisory signal). In supervised learning for image processing, for example, an AI system might be provided with labeled pictures of vehicles in categories such as

cars and trucks. After a sufficient amount of observation, the system should be able to distinguish between and categorize unlabeled images, at which time training can be said to be complete.

Supervised learning models have some advantages over the unsupervised approach, but they also have limitations. The systems are more likely to make judgments that humans can relate to, for example, because humans have provided the basis for decisions. However, in the case of a retrieval-based method, supervised learning systems have trouble dealing with new information. If a system with categories for cars and trucks is presented with a bicycle, for example, it would have to be incorrectly lumped in one category or the other. If the AI system was generative, however, it may not know what the bicycle is but would be able to recognize it as belonging to a separate category. The majority of practical machine learning uses supervised learning.

Supervised learning is where you have input variables (x) and an output variable (Y), and you use an algorithm to learn the mapping function from the input to the output. $Y = f(X)$

The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data.

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process. We know the correct answers; the algorithm iteratively makes predictions on the training data and is corrected by the teacher. It is learning stops when the algorithm achieves an acceptable level of performance. There are many algorithms that can be used for supervised learning, e.g., Support vector machine, neural networks, decision tree, k-nearest neighbors, etc.

Unsupervised machine learning

It is one of the two major types of machine learning. It refers to when systems can make sense out of the data only using input features. These algorithms identify patterns, anomalies, and similarities in the data. It is used for forming the clusters in the datasets. Clustering refers to a set of techniques and algorithms used to find clusters (subgroups) in a dataset and involves partitioning the data into groups of similar observations. The concept of 'similar observations' is a bit relative and subjective, but it essentially means that the data points in a given group are more similar to each other than they are to data points in a different group. There are many types of clustering algorithms and models, which all use their technique of dividing the data into a certain number of groups of similar data. Due to the significant difference in these approaches, the results can be primarily affected, and therefore one must understand these different algorithms to some extent to choose the most appropriate method to use. K-means, hierarchical clustering and DBSCAN are most widely used unsupervised clustering techniques. We will be discussing clustering techniques in further chapters.

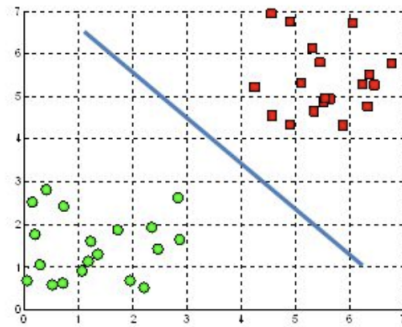
Chapter 2

Theory And Tools

Support Vector Machines

The Support Vector Machine algorithm originally carried out by two Mathematician Vladimir N.Vapnik and Alexey Ya. Chervonkis in the year of 1963. Later on, it was modified to work with nonlinear classifies by the use of kernel function or often called kernel trick. It is one of the most optimized and robust supervised machine learning algorithm used for discriminative classifier formally defined by a separating hyperplane. In other words, Given a set of points of two types in \mathbf{N} dimensional place SVM generates a $(\mathbf{N}-1)$ dimensional hyperplane to separate those points into two groups. In two dimensional space, this hyperplane is a line dividing a plane into two parts wherein each class lay in either side. 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 3. Support Vector Machine algorithm is meant to handle two types of problems, i.e., classification as well as regression.[3]

A hyperplane in \mathbb{R}^2 is a line



A hyperplane in \mathbb{R}^3 is a plane

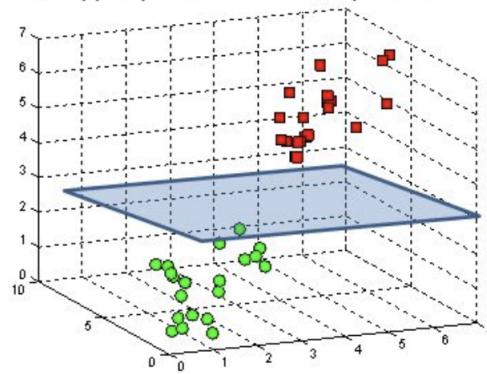


Figure 2.1: svm mode : hypothesis is a line in 2D and plane in 3D.[4]

Hypothesis formulation

Training data-sets(data used for the train a model) has been provided in the pair of input and output for supervised machine learning and further categorize in regression in case output is continuous value and classification when output is discrete.

$$\text{Training set: } (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}) \dots (x^{(m)}, y^{(m)}) \quad (2.1)$$

x : input features

y : output target

m : no. of training examples

These datasets are being used to build a model which learns a function that is being used in predicting unknown datasets. So we define a generalized function, which is to be determined during the training process described as Hypothesis function.

Hypothesis :

$$h_{\theta}(x^{(i)}) = \theta_0 x_0^{(i)} + \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)} + \dots + \theta_n x_n^{(i)} \quad (\text{for } x_0 = 1) \quad (2.2)$$

$$h_{\theta}(x^{(i)}) = \theta^T X^{(i)} \quad (2.3)$$

Where,

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \\ \vdots \\ \vdots \\ \theta_n \end{bmatrix} \in R^{n+1} \quad \text{and} \quad X = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ \vdots \\ \vdots \\ x_n \end{bmatrix} \in R^{n+1} \quad (2.4)$$

Where X and θ are features and parameters respectively, Hypothesis function will be a line in 2D feature space, the plane in 3D and hyperplane in case of more than 3D.

Regression

When training datasets have continuous output, ie.

$$y^{(i)} \in R \quad (2.5)$$

For best fit hyperplane, we compute cost function, also called least square error defined as:

$$\text{cost} : J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - (y^{(i)}))^2 \quad (2.6)$$

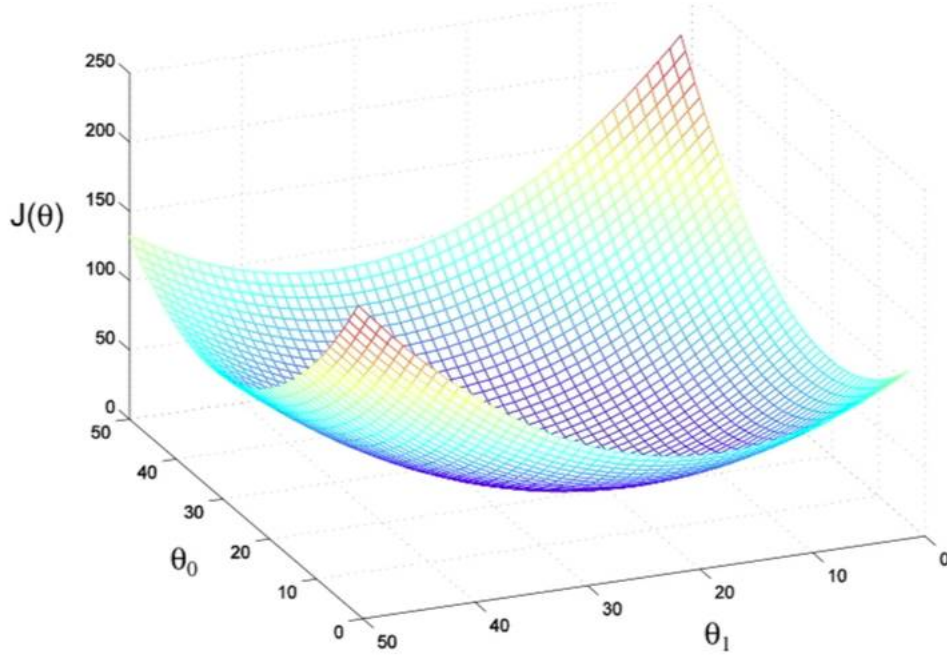


Figure 2.2: cost function $J(\theta)$ vs parameters (θ)

Where m is no training examples.

Gradient descent is being used to minimize the cost function for the value of θ and given as:

$$\theta_j := \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j} \quad (2.7)$$

$$\theta_j := \theta_j - \frac{\alpha}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (2.8)$$

here α is the learning rate(steps towards the local minima of cost function),

note: if α is too large gradient descent will fail to find the local minima, and if it is too small it will take huge time to converge at local minima, so choosing the value of α should be taken care.

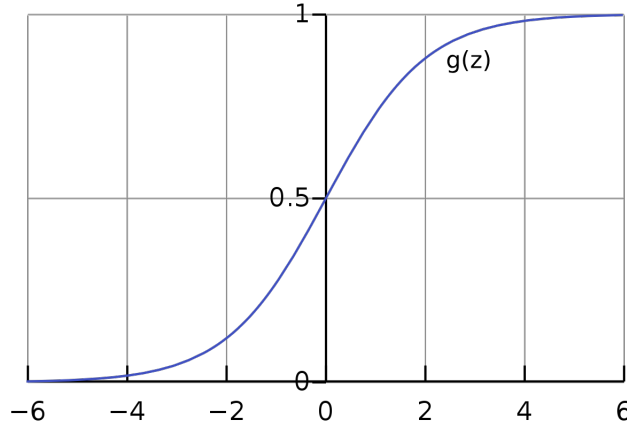
The plot shows the local minima of the cost function for the values of the parameter θ .

Classification

The output of training datasets has discrete values, ie.

$$y^{(i)} \in [0, 1] \quad (2.9)$$

Here 0 and 1 belong to the class label, eg. 0 labeled as electron and one labeled as pion when working of particle physics data; Log function is being used to come up with a hypothesis for hyperplane.



here $z = \theta^T X$ (x-axis)

$$g(\theta^T X^{(i)}) = \frac{1}{1 + e^{-\theta^T X^{(i)}}} = h_\theta(x^{(i)}) \quad (2.10)$$

Here using log function we want our model following rules for classification:

if $\theta^T X > 0$; we want our model to predict $y = 1$

if $\theta^T X < 0$; we want our model to predict $y = 0$

Cost function defined for the classification is given as:

$$J(\theta) = \frac{1}{m} \left(\sum_{i=1}^m y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \right) \quad (2.11)$$

note: There could be other cost function then this; this is derived from statistics using the principle of maximum likelihood estimation, as it has a nice property that is a convex function.

Again gradient descent is being used for minimization, ie :

$$\frac{\partial J(\theta)}{\partial \theta} = \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (2.12)$$

$$\text{so, } \theta_j := \theta_j - \frac{\alpha}{m} \sum_{i=1}^m (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (j = 0, 1, 2 \dots n) \quad (2.13)$$

looks similar to linear regression only change is the hypothesis function is modified with a log function, ie.

$$h_\theta(x^{(i)}) = \frac{1}{1 + e^{-\theta^T X^{(i)}}} \quad (2.14)$$

Regularization

When we best fit our hyperplane model on training dataset by computing cost function and built a model, most of the times model does not perform well on the testing datasets called **overfitting** of data, so to avoid overfitting we introduce another parameter called trade-off parameter denoted by C which controls θ .

so modified cost function with regularization parameter given as:

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m y^{(i)} \log x^{(i)} + (1 - y^{(i)}) (\log(1 - h_{\theta}(x^{(i)}))) \right] + \frac{\lambda}{2m} \sum_{j=1}^m \theta_j^2 \quad (2.15)$$

here $\lambda = \frac{1}{C}$

Gradient descent with regularization term:

$$\theta_j := \theta_j \left(1 - \alpha \frac{\lambda}{m} \right) - \alpha \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} \quad (2.16)$$

usually this term $\theta_j \left(1 - \alpha \frac{\lambda}{m} \right) < 1$; means regularization term is shrinking the value of θ .

V-Fold Cross Validation

There are two parameters: C and γ . It is not known beforehand which C and γ is best for a given problem; consequently, some model selection (parameter search) must be done. The goal is to identify good (C, γ) so that the classifier can accurately predict unknown data (i.e., testing data). Note that it may not be useful to achieve high training accuracy. A common strategy is to separate the data set into two parts, of which one is considered unknown. The prediction accuracy obtained from the “unknown” set more precisely reflects the performance on classifying an independent data set. An improved version of this procedure is known as cross-validation. In v-fold cross-validation, we first divide the training set into v subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining $(v-1)$. Thus, each instance of the whole training set is

predicted once, so the cross-validation accuracy is the percentage of data which are correctly classified. The cross-validation procedure can prevent the overfitting problem. It is very time-consuming and computationally costly so setting v should be taken care.

note : As γ is a kernel parameter which only comes into play when we need to map our data into higher dimensions, here we are only dealing with linear function.

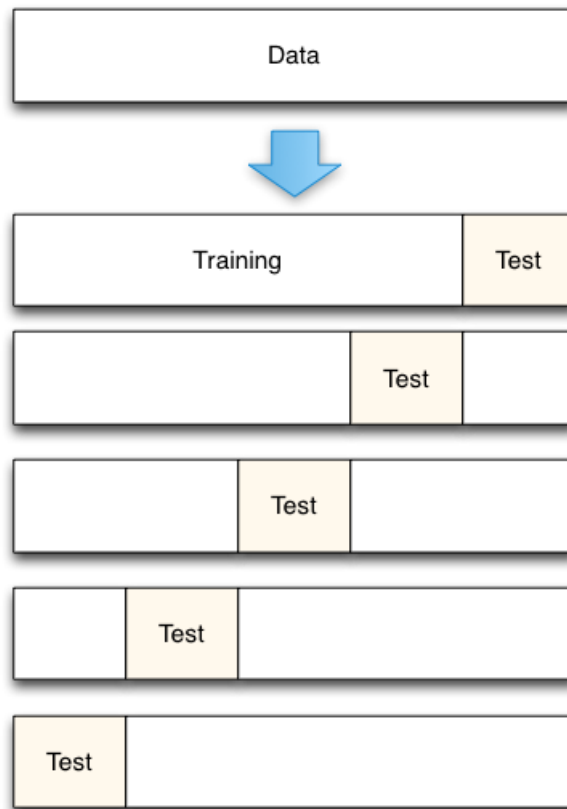


Figure 2.3: Five fold cross validation process.[6]

This method splits the data set into V equal partitions (“folds”), then use one fold as the testing set and the union of the other folds as the training set. Then the model is tested for accuracy. The process will follow the above steps K times, using the different fold as the testing set each time. The average testing accuracy of the process is testing accuracy.[5]

LibSvm

LIBSVM is an integrated software for support vector machines; it is available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>. It solves all kind of optimization problem with minimization of the cost function to find out the best regularization parameter for SVM, it computes cross validation search for values of C and γ , and these parameters are used to train a model further.[7]

Unsupervised learning and image processing

Image Processing : Steps for image processing

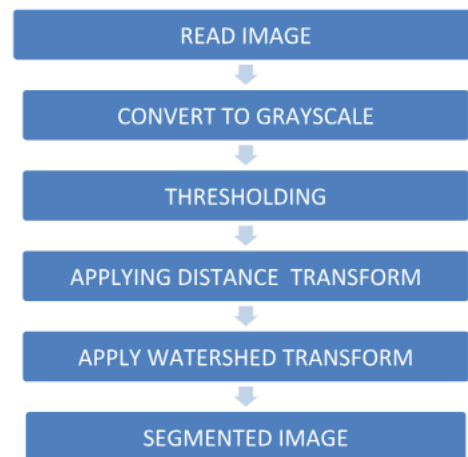


Figure 2.4: Different steps for image processing

The watershed is a classical algorithm used for segmentation in the image. Any grayscale image can be viewed as a topographic surface where high intensity denotes peaks and hills while low intensity denotes valleys. You start filling every isolated valley (local minima) with different colored water (labels). As the water rises, depending on the peaks (gradients) nearby, water from different valleys, obviously with different colors will start to merge. To avoid that, you build barriers in the locations where water merges. You continue the work of filling water and building barriers until all the peaks are under water. Then the barriers you created gives you the segmentation result.

- **DBSCAN : Density-based spatial clustering of applications with noise[8]**

Density-based spatial clustering of applications with noise (DBSCAN) is a well-known data clustering algorithm that is a unsupervised machine learning. Based on a set of points, DBSCAN groups together points that are close to each other based on a distance measurement (usually Euclidean distance) and a minimum number of points. It also marks as outliers the points that are in low-density regions.

The DBSCAN algorithm basically requires 2 parameters:

eps: specifies how close points should be to each other to be considered a part of a cluster.

minPoints: the minimum number of points to form a dense region.

- **In this algorithm, we have 3 types of data points.**

Core Point: it has more than MinPts points within eps.

Border Point: it has fewer than MinPts within eps but it is in the neighborhood of a core point.

Noise or outlier: A point which is not a core point or border point.

Working with DBSCAN

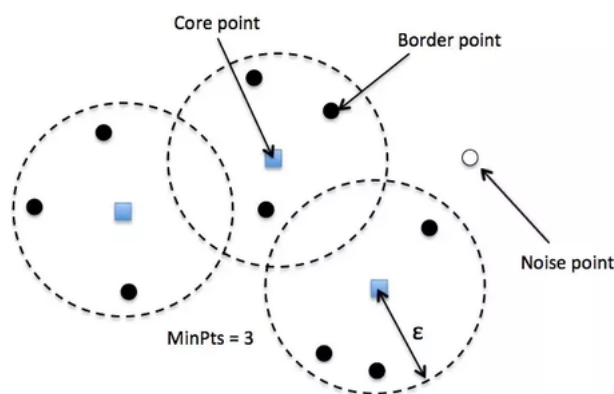


Figure 2.5: shows the steps of forming cluster.[9]

GOAL IS TO APPLY IMAGE PROCESSING AND FIND THE NO. OF CLUSTERS USING DBSCAN ALGORITHM

Diffraction Pattern of a Protein Crystal

X-ray crystallography is a technique used for determining the atomic and molecular structure of a crystal, in which the crystalline structure causes a beam of incident X-rays to diffract into many specific directions. By measuring the angles and intensities of these diffracted beams, the image we have taken is of protein crystal which has been formed in the laboratory of The Raja Ramanna Centre(RRCAT). An image has many constructive interference patterns that are bright fringes in the diffraction picture, and by the use of Unsupervised learning, we can find those bright fringes and make it as a cluster.

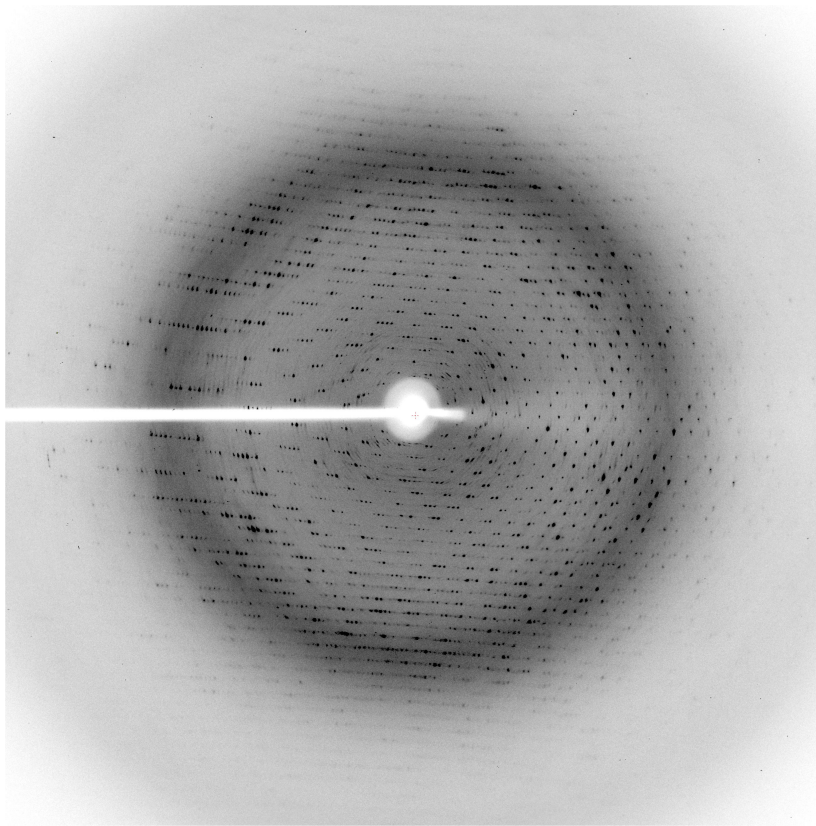


Figure 2.6: Given image of diffraction pattern of protein crystal.[10]

Image processing and DBSCAN Algorithm has been used to determine the clusters formed in the image.

Chapter 3

Analysis And Results

Prerequisites

- Python 3.+
- Anaconda (Scikit Learn, Numpy, Pandas, Matplotlib)
- Jupyter Notebook
- LibSvm
- GnuPlot
- Matlab
- Basic understanding of supervised machine learning methods : specifically classification.

The above is the essential requirement to work with the data and apply a machine learning algorithm. When encountered with a data set, first, we should analyze and “get to know” the data set. This step is necessary to familiarize with the data, to gain some understanding about the potential features, and to see if data cleaning is needed.

Cleaning:

There are several factors to consider in the data cleaning process:

- Duplicate or irrelevant observations.

- Bad labeling of data, same category occurring multiple times.
- Missing or null data points.

Feature Engineering: It is the process of transforming the gathered data into features that better represent the problem that we are trying to solve to the model, to improve its performance and accuracy.

Feature engineering creates more input features from the existing features and also combine several features to produce more intuitive features to feed to the model.

High energy dataset:

The data has been produced using Monte Carlo simulations.[**Appendix A**] When the collision occurs particles are mixed together, here in our case electron, and pion are mixed together and it is a very difficult task to separate them out, so machine learning has been used to build a classifier which learns a model and uses to separate them efficiently. The dataset contains three input features, and these features are derived by analysis to discriminate between the electron(background) and pion(signal). SVM has been used to separate the pion mixed with electron or vice-versa.

Attributes are follows:

- Momemtum(p)
- Theta(θ)
- Phi(ϕ)

Area	Data type	Accuracy without C	Accuracy with C (32768)
High energy physics	Classification	61.6%	66.2%

The above results show the SVM algorithm accurately classifies or differentiate between election and pion, which are mixed. The overall accuracy

is about 61.6%. With default parameters, but it raised to 66.2% when we tune the regularization parameter \mathbf{C} by using five-fold cross-validation method.

- Accuracy of seperating electron: 76.0%
- Accuracy of seperating pion : 56.4%

If we take the average of both it will return the overall accuracy with 66.2%

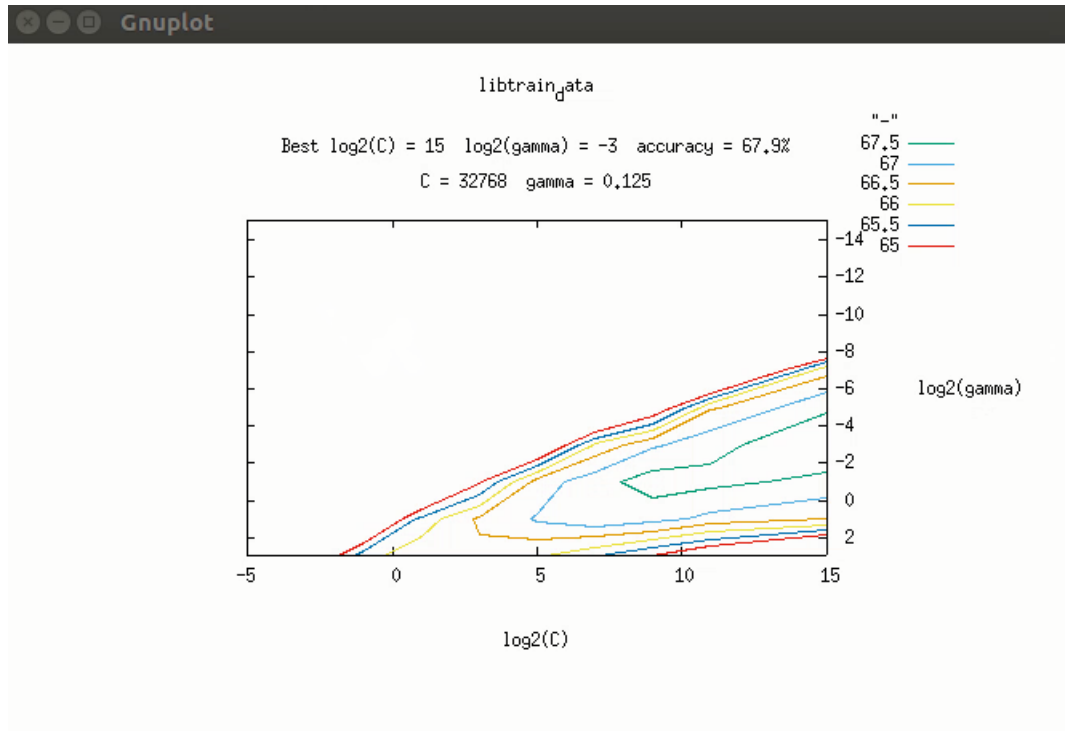


Figure 3.1: cross validation result for best \mathbf{C} and γ

Fig. 3.1 shows the cross validation result for best \mathbf{C} and γ , colors in the plot shows the every time accuracy changes while testing model on cross validation datasets.

Diabetes dataset:

Diabetes is a disease which occurs when the blood glucose level goes high, which eventually leads to other health problems such as heart diseases, kidney disease, etc. Diabetes is caused mainly due to the consumption of

highly processed food, inadequate consumption habits, etc. According to the World Health Organisation, the number of people with diabetes has been increased over the years. It is very costly for a common people to get the diagnosis, so machine learning can be used to build a system where it tells about the possibility of being a diabetes patient by inputting some symptoms which as called attributes of features in machine learning term. There are seven major diabetes attributes. [Appendix B]

Attributes are follows:

- Glucose
- Blood Pressure
- Skin Thickness
- Insulin
- BMI - Body Mass Index
- Diabetes Pedigree Function
- Age

Area	Data type	Accuracy withoutC	Accuracy with C(32768)
Diabetes	Classification	75.3%	77.4%

In the above result obtained from the SVM algorithm shows the accuracy of classification between people who have diabetes and the people are free from diabetes, here we can see with the tuning parameter C the accuracy is varying 2.1%.

Diffraction pattern of protien crystal

Clusters has been generated using image segmentation and no. of clusters has been found using DBSCAN algorithm.

- DMDBSCAN has been computed to determine the optimum value of eps parameter.

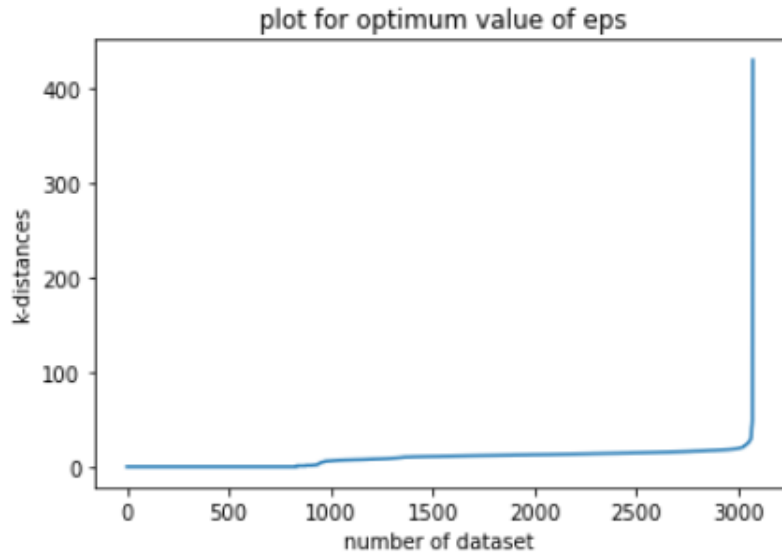


Figure 3.2: Parameter search using k-nearest neighbour for best eps value

- generated for optimum eps value using K-distance
- MinPts: 3 and optimum eps : 22 (point that have the greatest slope)

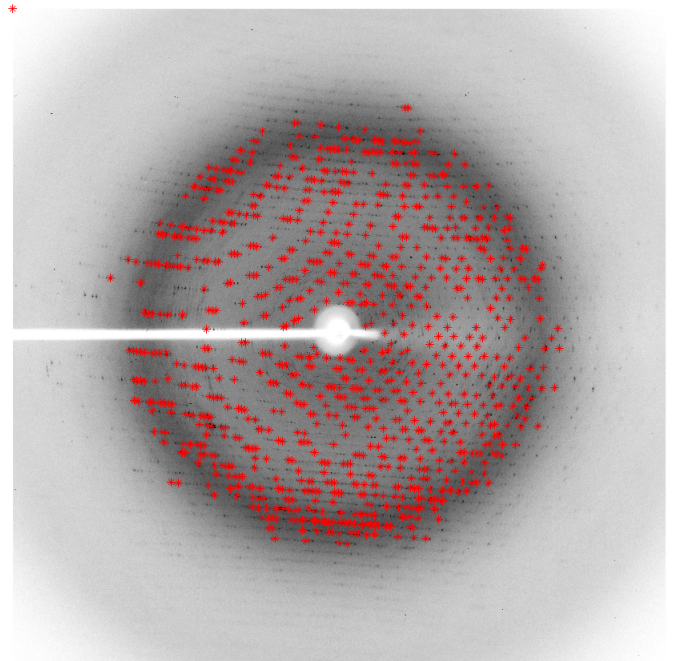


Figure 3.3: clusters generated using DBSCAN

Fig.3.3 shows the determined clusters in the image using DBSCAN algorithm with the eps value 22 and MinPts 3.

FUTURE WORK

- The result we obtained from DBSCAN Algorithm can be used further for supervised learning to come up with the trained model, which can perform for detecting the better diffraction pattern from the millions of random images.

Chapter 4

Application Of Machine Learning

Machine Learning represents the significant steps towards the computers to learn like humans, that gives the whole new vision to the world, there is much application almost in every field, some are discussed below:

- **Alerts (Maps):** Google Maps is probably the app we use whenever we go out and require assistance in directions and traffic. Maps suggested: "Despite the Heavy Traffic, you are on the fastest route." It takes the past data and makes the model with unsupervised machine learning and shows the prediction to the users.
- **Social Media (Facebook):** One of the most common applications of Machine Learning is Automatic Friend Tagging Suggestions on Facebook or any other social media platform. Facebook uses face detection and Image recognition to automatically find the face of the person which matches its Database and hence suggests us to tag that person based on DeepFace.
- **Transportation and Commuting (Uber):** It uses Machine Learning algorithm which uses historic Trip Data to make a more accurate ETA prediction. With the implementation of Machine Learning, they saw a 26 percentage accuracy in Delivery and Pickup.

- **Products Recommendations:** Suppose you check an item on Amazon, but you do not buy it then and there. But the next day, you're watching videos on YouTube, and suddenly you see an ad for the same item. You switch to Facebook, there also you see the same advertisement.
- **Virtual Personal Assistants:** As the name suggests, Virtual Personal Assistants assist in finding useful information, when asked via text or voice. Few of the crucial Applications of Machine Learning here are:
 - a) Speech Recognition
 - b) Speech to Text Conversion
 - c) Natural Language Processing
 - d) Text to Speech Conversion
 All you need to do is ask the question from, Siri,/google assistant/Alexa or any personal assistant and that uses machine learning to give the response to your questions. Recently personal assistants are being used in Chatbots which are being implemented in various food ordering apps, online training websites and also in Commuting apps.
- **Self Driving Cars:** Well, here is one of the coolest application of Machine Learning. It's here, and people are already using it. Machine Learning plays a very important role in Self Driving Cars. All might have heard about Tesla. The leader in this business and their current Artificial Intelligence is driven by hardware manufacturer NVIDIA, which is based on Unsupervised Learning Algorithm.
- **Dynamic Pricing:** Setting the right price for a good or service is an old problem in economic theory. There are a vast amount of pricing strategies that depend on the objective sought. Be it a movie ticket, a plane ticket or cab fares, everything is dynamically priced. In recent years, Machine Learning has enabled pricing solutions to track buying trends and determine more competitive product prices.
- **Online Video Streaming (Netflix):** With over millions of subscribers, there is no doubt that Netflix is best of the online streaming

world. The Netflix uses Machine Learning algorithm constantly gathers massive amounts of data about users' activities like:

- a) When the user pause, rewind or fast forward
- b) What day user watch content (TV Shows on Weekdays and Movies on Weekends)
- c) The Date and Time you watch
- d) When you pause and leave content
- e) The ratings Given (about 4 million per day), Searches (about 3 million per day)
- f) Browsing and Scrolling Behavior and almost everything.

- **Fraud Detection:** Fraud Detection is one of the most necessary Applications of Machine Learning. The number of transactions has increased due to a plethora of payment channels – credit/debit cards, smartphones, numerous wallets, UPI, and much more. At the same time, the number of criminals has become adept at finding loopholes. Whenever a customer carries out a transaction – the Machine Learning model thoroughly x-rays their profile, searching for suspicious patterns. In Machine Learning, problems like fraud detection are usually framed as classification problems under supervised ML.

Above are some examples in the field of machine learning apart from this, machine learning has been used almost in every domain, it is emerging in the field of medical science for medical diagnosis and in robotics where robots interact with people and communicate with them as a normal human. ML is creating a new era, and our future technology is going to be heavily driven on ML and deep learning algorithms.

Appendices

Appendix A

HIGH ENERGY DATASET

0.0 0:0.272608 1:0.831977 2:0.79052 0.0 0:0.346416 1:0.775786 2:1.58391
0.0 0:0.110355 1:0.640251 2:0.502849 0.0 0:0.202378 1:0.743214 2:-0.141936
0.0 0:0.15568 1:0.395943 2:1.63518 0.0 0:0.228272 1:0.308686 2:0.605667
0.0 0:0.036578 1:1.0704 2:0.738827 0.0 0:0.618229 1:0.297597 2:2.48569 0.0
0:0.153334 1:1.56683 2:1.38492 0.0 0:0.361 1:0.486136 2:-1.60712 0.0 0:0.290789
1:0.388376 2:2.47519 0.0 0:0.0117805 1:0.474411 2:-2.66628 0.0 0:0.0852504
1:0.75261 2:2.92471 0.0 0:0.149547 1:0.864272 2:2.09128 0.0 0:0.202147 1:0.452645
2:-1.8651 0.0 0:0.170423 1:0.788671 2:-0.207581 0.0 0:0.391568 1:0.626014
2:2.12114 0.0 0:0.177411 1:1.34938 2:-3.13917 0.0 0:0.271523 1:0.977798
2:1.3271 0.0 0:0.181116 1:0.256566 2:-1.77128 0.0 0:0.0670026 1:0.86279
2:1.52989 0.0 0:0.0109131 1:0.695863 2:-0.867871 0.0 0:0.164147 1:1.02855
2:0.00770141 0.0 0:0.0698997 1:1.34167 2:2.00982 0.0 0:0.218155 1:1.30241
2:3.07677 0.0 0:0.0994148 1:0.78337 2:2.15712 0.0 0:0.0768956 1:0.412286
2:1.23999 0.0 0:0.0668097 1:0.972083 2:0.542441 0.0 0:0.228172 1:0.90676
2:-1.50444 0.0 0:0.0454396 1:1.32677 2:2.4407 0.0 0:0.0341962 1:1.99323 2:-
2.47493 0.0 0:0.045172 1:0.181795 2:-3.04899 0.0 0:0.10119 1:2.27294 2:2.06146
0.0 0:0.213654 1:1.36755 2:0.748784 0.0 0:0.264569 1:1.11187 2:1.85366 0.0
0:0.124339 1:2.72638 2:0.242539 0.0 0:0.341833 1:0.534212 2:-2.40202 0.0
0:0.279955 1:0.999668 2:0.241558 0.0 0:0.108082 1:1.24624 2:-1.94868 0.0
0:0.364853 1:0.646351 2:-2.93386 0.0 0:0.0134632 1:1.2137 2:1.94707 0.0
0:0.0541091 1:0.441056 2:1.99216 0.0 0:0.0900097 1:0.655477 2:-0.232207
0.0 0:0.238732 1:1.09776 2:-2.23711 0.0 0:0.174955 1:0.351757 2:2.27869 0.0
0:0.486466 1:0.207339 2:0.904746 0.0 0:0.0785854 1:0.311131 2:-0.252006

0.0 0:0.169658 1:0.911413 2:0.203139 0.0 0:0.181792 1:1.10748 2:0.012218
 0.0 0:0.270757 1:0.99019 2:0.598941 0.0 0:0.0226541 1:0.56577 2:2.76532
 0.0 0:0.328382 1:0.671881 2:1.17792 0.0 0:0.0876404 1:1.75524 2:2.83341
 0.0 0:0.351384 1:0.524423 2:2.76271 0.0 0:0.181599 1:1.69687 2:1.15789 0.0
 0:0.137496 1:2.12936 2:2.38349 0.0 0:0.385204 1:0.774917 2:0.521696 0.0
 0:0.502751 1:0.388646 2:-0.907742 0.0 0:0.123497 1:2.34274 2:1.74031 0.0
 0:0.303327 1:0.614512 2:-1.08142 0.0 0:0.179289 1:0.926747 2:-1.83638 0.0
 0:0.707347 1:0.332387 2:2.71065 0.0 0:0.145808 1:1.20671 2:0.342628 0.0
 0:0.484407 1:0.498034 2:2.91083 0.0 0:0.114456 1:1.26103 2:1.61244 0.0 0:0.185784
 1:1.32867 2:-2.21229 0.0 0:0.00825064 1:0.716023 2:0.553769 0.0 0:0.033701
 1:1.01943 2:1.43463 0.0 0:0.104296 1:0.918941 2:1.58762 0.0 0:0.17537 1:0.890835
 2:2.57232 0.0 0:0.199644 1:1.21737 2:-0.669532 0.0 0:0.294293 1:0.518226
 2:1.90054 0.0 0:0.398506 1:1.03964 2:2.49162 0.0 0:0.0261787 1:1.76793 2:2.04437
 0.0 0:0.0132962 1:2.03407 2:-2.81165 0.0 0:0.253086 1:0.925469 2:0.719094
 0.0 0:0.192837 1:1.49075 2:-2.83863 0.0 0:0.392171 1:0.343489 2:1.45801 0.0
 0:0.0990438 1:0.726624 2:-0.732185 0.0 0:0.153944 1:1.19905 2:2.64509 0.0
 0:0.0859674 1:0.855216 2:2.19928 0.0 0:0.420938 1:0.634636 2:1.03649 0.0
 0:0.359378 1:0.586257 2:-2.70893 0.0 0:0.183798 1:1.22895 2:-0.111901 0.0
 0:0.628674 1:0.0731876 2:-1.47781 0.0 0:0.0440576 1:1.34922 2:-0.870507
 0.0 0:0.220518 1:1.01087 2:1.64972 0.0 0:0.45279 1:0.366745 2:3.0798 1.0
 0:0.111105 1:0.772622 2:-1.84195 1.0 0:0.178059 1:0.456078 2:-1.78065 1.0
 0:0.158217 1:0.599536 2:2.5098 1.0 0:0.471684 1:0.416691 2:0.0189362 1.0
 0:0.20505 1:0.654725 2:1.18726 1.0 0:0.256538 1:1.01677 2:-0.868615 1.0
 0:0.090169 1:1.47137 2:-3.06882 1.0 0:0.29856 1:0.801356 2:1.78454 1.0 0:0.0861348
 1:0.332509 2:-0.335897 1.0 0:0.388927 1:0.683727 2:-0.852895 1.0 0:0.175661
 1:1.19887 2:-1.85264 1.0 0:0.490127 1:0.194722 2:-2.03324 1.0 0:0.221016
 1:0.388609 2:1.36543 1.0 0:0.450484 1:0.186699 2:1.45537 1.0 0:0.315354
 1:0.869171 2:-2.25295 1.0 0:0.31061 1:0.246198 2:-3.00349 1.0 0:0.191433
 1:0.828033 2:-2.88176 1.0 0:0.019551 1:1.77674 2:2.96868 1.0 0:0.368612
 1:0.450271 2:2.85228 1.0 0:0.441835 1:0.0269582 2:-0.641167 1.0 0:0.0888598
 1:1.56994 2:1.39985 1.0 0:0.190441 1:1.15279 2:-0.939196 1.0 0:0.380065
 1:0.429569 2:1.79556 1.0 0:0.180667 1:0.784744 2:-2.19361 1.0 0:0.422309

1:0.428403 2:2.01923 1.0 0:0.263777 1:0.712367 2:-0.339322 1.0 0:0.248797
1:1.15827 2:-1.10159 1.0 0:0.310598 1:0.237382 2:-0.11711 1.0 0:0.104802
1:0.713938 2:1.70672 1.0 0:0.212006 1:0.215716 2:-1.47617 1.0 0:0.330594
1:0.651643 2:2.86566 1.0 0:0.101474 1:1.08475 2:-1.26391 1.0 0:0.230752
1:0.592582 2:-3.14001 1.0 0:0.14381 1:1.23874 2:-0.523473 1.0 0:0.322605
1:0.530304 2:-1.90288 1.0 0:0.216315 1:0.811248 2:2.66667 1.0 0:0.0661208
1:1.65209 2:-0.980718 1.0 0:0.152827 1:0.673142 2:0.468902 1.0 0:0.280438
1:0.564844 2:-2.39903 1.0 0:0.180797 1:1.08593 2:-0.0820307 1.0 0:0.281886
1:0.626464 2:1.87846 1.0 0:0.113257 1:0.832065 2:-0.48837 1.0 0:0.263143
1:0.683731 2:-2.99294 1.0 0:0.0761769 1:1.79373 2:0.071201 1.0 0:0.205862
1:0.832188 2:-0.560689 1.0 0:0.276446 1:0.8096 2:-2.06938 1.0 0:0.18855 1:0.437455
2:0.876051 1.0 0:0.344592 1:0.0292238 2:-0.0969882 1.0 0:0.266881 1:0.925037
2:-2.9437 1.0 0:0.403987 1:0.715127 2:-1.31384 1.0 0:0.221181 1:0.680304
2:0.799612 1.0 0:0.383996 1:0.205601 2:2.56561 1.0 0:0.149563 1:0.811356
2:-0.956625 1.0 0:0.127282 1:0.833877 2:-1.49505 1.0 0:0.326588 1:0.488323
2:1.98746 1.0 0:0.265295 1:0.233009 2:-1.10152 1.0 0:0.258416 1:0.892214
2:0.738039 1.0 0:0.423735 1:0.125807 2:-1.53636 1.0 0:0.159543 1:0.535811
2:-2.20728 1.0 0:0.373958 1:0.555676 2:-2.48047 1.0 0:0.255645 1:0.785077
2:0.547217 1.0 0:0.461517 1:0.304729 2:0.0219486 1.0 0:0.0724595 1:1.37632
2:-2.18814 1.0 0:0.232945 1:1.13534 2:-3.11527 1.0 0:0.245371 1:0.62144 2:-
0.329368 1.0 0:0.251378 1:0.602875 2:1.43154 1.0 0:0.335482 1:0.641922
2:0.86878 1.0 0:0.262205 1:0.714121 2:2.14915 1.0 0:0.229869 1:0.875207 2:-
0.0947139 1.0 0:0.427228 1:0.357578 2:0.714324 1.0 0:0.0722426 1:0.972888
2:2.55215 1.0 0:0.395865 1:0.598898 2:1.73929upto
15000 instances.

Appendix B

DIABETES DATASET

-1 1:-0.294118 2:0.487437 3:0.180328 4:-0.292929 5:-1 6:0.00149028 7:-0.53117
8:-0.0333333 +1 1:-0.882353 2:-0.145729 3:0.0819672 4:-0.414141 5:-1 6:-
0.207153 7:-0.766866 8:-0.666667 -1 1:-0.0588235 2:0.839196 3:0.0491803
4:-1 5:-1 6:-0.305514 7:-0.492741 8:-0.633333 +1 1:-0.882353 2:-0.105528
3:0.0819672 4:-0.535354 5:-0.777778 6:-0.162444 7:-0.923997 8:-1 -1 1:-1
2:0.376884 3:-0.344262 4:-0.292929 5:-0.602837 6:0.28465 7:0.887276 8:-0.6
+1 1:-0.411765 2:0.165829 3:0.213115 4:-1 5:-1 6:-0.23696 7:-0.894962 8:-0.7
-1 1:-0.647059 2:-0.21608 3:-0.180328 4:-0.353535 5:-0.791962 6:-0.0760059
7:-0.854825 8:-0.833333 +1 1:0.176471 2:0.155779 3:-1 4:-1 5:-1 6:0.052161
7:-0.952178 8:-0.733333 -1 1:-0.764706 2:0.979899 3:0.147541 4:-0.0909091
5:0.283688 6:-0.0909091 7:-0.931682 8:0.0666667 -1 1:-0.0588235 2:0.256281
3:0.57377 4:-1 5:-1 6:-1 7:-0.868488 8:0.1 +1 1:-0.529412 2:0.105528 3:0.508197
4:-1 5:-1 6:0.120715 7:-0.903501 8:-0.7 -1 1:0.176471 2:0.688442 3:0.213115
4:-1 5:-1 6:0.132638 7:-0.608027 8:-0.566667 +1 1:0.176471 2:0.396985 3:0.311475
4:-1 5:-1 6:-0.19225 7:0.163962 8:0.2 -1 1:-0.882353 2:0.899497 3:-0.0163934
4:-0.535354 5:1 6:-0.102832 7:-0.726729 8:0.266667 -1 1:-0.411765 2:0.668342
3:0.180328 4:-0.616162 5:-0.586288 6:-0.230999 7:-0.565329 -1 1:-0.176471
2:0.00502513 3:-1 4:-1 5:-1 6:-0.105812 7:-0.653288 8:-0.633333 -1 1:-1 2:0.18593
3:0.377049 4:-0.0505051 5:-0.456265 6:0.365127 7:-0.596072 8:-0.666667 -
1 1:-0.176471 2:0.0753769 3:0.213115 4:-1 5:-1 6:-0.117735 7:-0.849701 8:-
0.666667 +1 1:-0.882353 2:0.0351759 3:-0.508197 4:-0.232323 5:-0.803783
6:0.290611 7:-0.910333 8:-0.6 -1 1:-0.882353 2:0.155779 3:0.147541 4:-0.393939
5:-0.77305 6:0.0312965 7:-0.614859 8:-0.633333 +1 1:-0.647059 2:0.266332

3:0.442623 4:-0.171717 5:-0.444444 6:0.171386 7:-0.465414 8:-0.8 +1 1:-0.0588235
 2:-0.00502513 3:0.377049 4:-1 5:-1 6:0.0551417 7:-0.735269 8:-0.0333333 -1
 1:-0.176471 2:0.969849 3:0.47541 4:-1 5:-1 6:0.186289 7:-0.681469 8:-0.333333
 -1 1:0.0588235 2:0.19598 3:0.311475 4:-0.292929 5:-1 6:-0.135618 7:-0.842015
 8:-0.733333 -1 1:0.294118 2:0.437186 3:0.540984 4:-0.333333 5:-0.654846
 6:0.0909091 7:-0.849701 -1 1:0.176471 2:0.256281 3:0.147541 4:-0.474747
 5:-0.728132 6:-0.0730253 7:-0.891546 8:-0.333333 -1 1:-0.176471 2:0.477387
 3:0.245902 4:-1 5:-1 6:0.174367 7:-0.847139 8:-0.266667 +1 1:-0.882353 2:-
 0.0251256 3:0.0819672 4:-0.69697 5:-0.669031 6:-0.308495 7:-0.650726 8:-
 0.966667 +1 1:0.529412 2:0.457286 3:0.344262 4:-0.616162 5:-0.739953 6:-
 0.338301 7:-0.857387 8:0.2 +1 1:-0.411765 2:0.175879 3:0.508197 4:-1 5:-1
 6:0.0163934 7:-0.778822 8:-0.433333 +1 1:-0.411765 2:0.0954774 3:0.229508
 4:-0.474747 5:-1 6:0.0730254 7:-0.600342 8:0.3 -1 1:-0.647059 2:0.58794 3:0.245902
 4:-0.272727 5:-0.420804 6:-0.0581222 7:-0.33988 8:-0.766667 +1 1:-0.647059
 2:-0.115578 3:-0.0491803 4:-0.777778 5:-0.87234 6:-0.260805 7:-0.838599 8:-
 0.966667 +1 1:-0.294118 2:-0.0753769 3:0.508197 4:-1 5:-1 6:-0.406855 7:-
 0.906063 8:-0.766667 +1 1:0.176471 2:0.226131 3:0.278689 4:-0.373737 5:-1
 6:-0.177347 7:-0.629377 8:-0.2 +1 1:-0.529412 2:0.0351759 3:-0.0163934 4:-
 0.333333 5:-0.546099 6:-0.28465 7:-0.241674 8:-0.6 +1 1:0.294118 2:0.386935
 3:0.245902 4:-1 5:-1 6:-0.0104321 7:-0.707942 8:-0.533333 -1 1:0.0588235
 2:0.0251256 3:0.245902 4:-0.252525 5:-1 6:-0.019374 7:-0.498719 8:-0.166667
 -1 1:-0.764706 2:-0.0954774 3:0.114754 4:-0.151515 5:-1 6:0.138599 7:-0.637062
 8:-0.8 -1 1:-0.529412 2:0.115578 3:0.180328 4:-0.0505051 5:-0.510638 6:0.105812
 7:0.12041 8:0.166667 +1 1:-0.647059 2:0.809045 3:0.0491803 4:-0.494949 5:-
 0.834515 6:0.0134128 7:-0.835184 8:-0.833333 +1 1:-0.176471 2:0.336683
 3:0.377049 4:-1 5:-1 6:0.198212 7:-0.472246 8:-0.466667 +1 1:-0.176471 2:0.0653266
 3:0.508197 4:-0.636364 5:-1 6:-0.323398 7:-0.865927 8:-0.1 -1 1:0.0588235
 2:0.718593 3:0.803279 4:-0.515152 5:-0.432624 6:0.353204 7:-0.450897 8:0.1
 +1 1:-0.176471 2:0.59799 3:0.0491803 4:-1 5:-1 6:-0.183308 7:-0.815542 8:-
 0.366667 -1 1:-1 2:0.809045 3:0.0819672 4:-0.212121 5:-1 6:0.251863 7:0.549957
 8:-0.866667 +1 1:-0.882353 2:0.467337 3:-0.0819672 4:-1 5:-1 6:-0.114754 7:-
 0.58497 8:-0.733333 +1 1:-0.764706 2:-0.286432 3:0.147541 4:-0.454545 5:-1

6:-0.165425 7:-0.566183 8:-0.966667 -1 1:-0.176471 2:0.0351759 3:0.0819672
 4:-0.353535 5:-1 6:0.165425 7:-0.772844 8:-0.666667 +1 1:-0.176471 2:0.0552764
 3:-1 4:-1 5:-1 6:-1 7:-0.806149 8:-0.9 +1 1:-0.882353 2:0.0351759 3:0.311475
 4:-0.777778 5:-0.806147 6:-0.421759 7:-0.64731 8:-0.966667 +1 1:-0.882353
 2:0.0150754 3:-0.180328 4:-0.69697 5:-0.914894 6:-0.278688 7:-0.617421 8:-
 0.833333
upto 800 patients

Bibliography

1. Alex Smola and S.V.N. Vishwanathan "*Introduction to Machine Learning*"
2. <https://www.techsparks.co.in>
3. Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin "*A Practical Guide to Support Vector Classification*"
4. <https://quantdare.com/svm/hypothesis>
5. Libsvm package: <https://www.csie.ntu.edu.tw/~cjlin/libsvm>
Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin "*A Practical Guide to Support Vector Classification*"
6. Y.Liu, " Python machine learning by examples"
7. LIBSVM:Chih-Chung Chang and Chih-Jen Lin National Taiwan University, Taipei, Taiwan "*A Library for Support Vector Machines*"
8. Determination of Optimal Epsilon (Eps) Value on DB-SCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra
9. <https://medium.com/@elutins/dbscan>
10. Diffraction Pattern Of a protien crystal: Laboratory of The Raja Ramanna Centre(RRCAT)