**IMPLEMENTATION OF KNN ALGORITHM USING MACHINE LEARNING FOR HEALTH CARE APPLICATIONS**

**A PROJECT REPORT**

***Submitted to***

*in the partial fulfillment of the requirements for*

*the award of the degree of*

in the Department of

**Electronics and Communication Engineering [ECE]**

by

**CH. SANATH KUMAR- 16311A04C8**

**G. BHARATH KUMAR - 16311A04D5**

Under the Guidance of

**MRS. N. ANUPAMA RANI**

Assistant Professor

Department of ECE

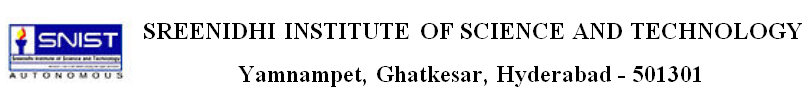


**DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING**

**SREENIDHI INSTITUTE OF SCIENCE & TECHNOLOGY**

**(AUTONOMOUS)**

**Yamnampet, Ghatkesar, R.R District, Hyderabad – 501301**

****

**Date:**

**CERTIFICATE**

This is to certify that the Project Work entitled “**IMPLEMENTATION OF KNN ALGORITHM USING MACHINE LEARNING FOR HEALTH CARE APPLICATIONS**”,

being submitted by

**CH. SANATH KUMAR (ROLL NO. 16311A04C8) &**

**G. BHARATH KUMAR (ROLL NO. 16311A04D5)**

in fulfillment of B.Tech 4th year 1st semester in Department of **Electronics and Communication Engineering [ECE], Sreenidhi Institute of Science and Technology,** an autonomous institute under Jawaharlal Nehru Technological University, Telangana, is a record of bonafide work carried out by them under our guidance and supervision.

**PROJECT GUIDE PROJECT COORDINATOR**

**Mrs. N. ANUPAMA RANI Dr. D. AJITHA**

**Assistant Professor Associate Professor**

**DEPARTMENT OF ECE DEPARTMENT OF ECE**

**Dr. S.P.V SUBBARAO**

**Head & Professor**

**DEPARTMENT OF ECE**

**ACKNOWLEDGEMENT**

We would like to thank my supervisor **Mrs. N. ANUPAMA RANI,** professor, Department of ECE for giving us constant guidance, support and motivation throughout the period this course work was carried out. Their readiness for consultation at all times, their educative comments and assistance even with practical things have been invaluable. We are thankful that she had gave the freedom to do the work with our ideas.

We express my sincere gratitude to **DR. S.P.V. SUBBA RAO**, Head of Department, ECE for helping us in carrying out this project giving support throughout the period of our study in SNIST.

We are also thankful to our principle **DR. CH. SIVA REDDY**, giving us this guidance and support, motivation throughout the period of our B.Tech course work carried out.

We convey our special thanks to honorable Executive Director **DR. P.N. REDDY SNIST** for his continuous support in maintaining pleasant and friendly atmosphere in campus for studies and course work.

We are also thankful to all teaching and non-teaching staff of our department who has rendered their co-operation in completion of this project report.

We also thank our parents and friends and well-wishers who aided us in completion of the project report.

**CH. SANATH KUMAR (ROLL NO. 16311A04C8)**

**G. BHARATH KUMAR (ROLL NO. 16311A04D5)**

**DECLARATION**

We hereby declare that the work described in this report, entitled “**IMPLEMENTATION OF KNN ALGORITHM USING MACHINE LEARNING FOR HEALTH CARE APPLICATIONS**”, which is being submitted by us in fulfillment of B.Tech 4th year, 1st semester in Department of **Electronics and Communication Engineering [ECE], Sreenidhi Institute Of Science & Technology** affiliated to Jawaharlal Nehru Technological University Hyderabad, Kukatpally, Hyderabad (Telangana) -500 085 is the result of investigations carried out by us under the Guidance of **Mrs. ANUPAMA RANI,** Assistantprofessor, Department of ECE, Sreenidhi Institute Of Science And Technology, Hyderabad. The work is original and has not been submitted for any Degree/Diploma of this or any other university.

Place: Hyderabad

Date:

**Signature**

**CH. SANATH KUMAR (ROLL NO. 16311A04C8)**

**G. BHARATH KUMAR (ROLL NO. 16311A04D5)**

**CONTENTS**

**TITLE Page Number**

List of Figures 7

Abstract 8

**Chapter-1 Introduction** 9-12

1.1 Objective 13

* 1. Problem definition 13

**Chapter-2** **LITERATURE SURVEY** 14

**Chapter-3 Methodology**

3.1 Classification algorithms15 3.2 KNN algorithm 15

3.3 Flow Chart 16-17

**Chapter-4** **Software Tools**

4.1 Software Description 18

4.2 Language Description 18

**Chapter-5** **Experiment analysis and results**

5.1 Data Set Description 19-20

5.2 Output Graphs 21-27

5.3 Advantages 28

5.4 Disadvantages 28

5.5 Applications 29

**Chapter-6 Conclusion and Future Scope** 30

**Chapter-7 References** 31

**APPENDIX**

1. **Source Code** 32-33
2. **Sample Data Set** 34-36

**LIST OF FIGURES**

|  |  |
| --- | --- |
| **Figure no.** | **Name** |
| 1 | k nearest neighbor clustering |
| 2 | Clustering |
| 3 | KNN data recognition model for patients |
| 4 | Sample data set |
| 5 | Health care center based on region |
| 6 | Health care center based on Age |
| 7 | Health care center based on level of treatment |
| 8 | Health care center based on charges |
| 9 | Health care center based on income |
| 10 | Health care center based on disease and problems |
| 11 | Health care center based on address |
| 12 | Health care center based on gender |
| 13 | Total number of hospitals in region |
| 14 | Health care center based on speciality |

**ABSTRACT**

Clustering is defined as a discrete streamlining issue. The goal of Clustering is to locate the best one, among all parcels of the informational index, as indicated by some quality measure. In the measurable setting where we accept that the limited informational index has been inspected from some fundamental space, the objective isn't to locate the best segment of the given example, yet to rough the genuine segment of the basic space.

KNN calculation is utilized to comprehend the arrangement and relapse model issues. K-closest neighbor or K-NN calculation makes a fanciful limit to group the information. At the point when new information comes in, the calculation will attempt to anticipate that to the closest of the limit line. Along these lines, bigger k esteem implies cover bends of partition bringing about less mindboggling models. Though, littler k esteem prompts complex models.

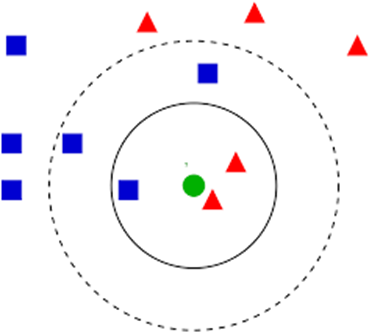
We propose a novel and efﬁcient Clustering based KNN relapse calculation which as opposed to looking for closest neighbors legitimately in the whole dataset, ﬁrst ﬁnd the bunch wherein the question point should lie. We ﬁrst Hierarchically group the information in the preprocessing step, at that point a recursive pursuit beginning from root hub of the order is performed. At last we ﬁnd the k closest neighbors of inquiry focus in that group and return the weighted mean of their reaction variable as result.

**Keywords:** Clustering, KNN algorithm, Regression.

**CHAPTER- 1**

**INTRODUCTION**

Clustering is the issue of finding "significant" bunches in given information. By a the most widely recognized way to deal with Clustering is to deﬁne a grouping quality capacity Qn, and afterward build a calculation which can limit (or expand) Qn. There exists an immense assortment of grouping quality capacities: the K-implies target work dependent on the separation of the information focuses to the bunch focuses, chart cut based target capacities, for example, proportion cut or standardized cut, or different criteria dependent on some capacity of the inside and between-group similitudes. When a specific grouping quality capacity Qn has been chosen, the target of Clustering is expressed as a discrete improvement issue. Given an informational index Xn ={X1,… Xn} and a Clustering quality capacity Qn, the perfect grouping calculation should consider every conceivable parcel of the informational collection and yield the one that limits Qn. The certain comprehension is that "as well as can be expected" be any segment out of the arrangement of every single imaginable segment of the informational collection. The useful test is then to develop a calculation which can unequivocally register this "best" grouping by taking care of an enhancement issue. We will consider this methodology the "discrete enhancement way to deal with Clustering". Presently let us take a gander at grouping from the point of view of factual learning hypothesis. Here we accept that the ﬁnite informational collection has been examined from a fundamental information space X as indicated by some likelihood measure P. A definitive objective in this setting isn't to find the most ideal segment of the informational collection Xn, however to gain proficiency with the "genuine grouping" of the hidden space X. While it isn't evident how this "genuine Clustering" ought to be deﬁned in a general setting (cf. von Luxburg and Ben-David, 2005), in a methodology dependent on quality capacities this is clear. We pick a Clustering quality capacity Q on the arrangement of allotments of the whole information space X, and deﬁne the genuine grouping f∗ to be the segment of X which limits Q. In a ﬁnite test defining, the objective is presently to inexact this genuine grouping just as conceivable. To this end, we deﬁne an observational quality capacity Qn which can be assessed dependent on the ﬁnite test just, and develop the exact grouping fn as the minimizer of Qn. In this setting, a significant property of a Clustering calculation is consistency: we require that Q(fn) combines to Q(f∗) when n→∞. This unequivocally helps to remember the standard methodology in administered classiﬁcation, the experimental hazard minimization approach. For this methodology, the most significant understanding of measurable learning hypothesis is that so as to be reliable, learning calculations need to pick their capacities from some "little" work space as it were. There are numerous ways how the size of a capacity space can be quantiﬁed. Probably the simplest ways is to utilize breaking coefﬁcients s(F ,n) (see Section 2 for subtleties). An ordinary outcome in measurable learning hypothesis is that a vital condition for consistency is Elogs(F ,n)/n→0 (cf. Hypothesis 2.3 in Vapnik, 1995, Section 12.4 of Devroye et al., 1996). That is, the "quantity of capacities" s(F ,n) in F must not develop exponentially in n, else one can't ensure for consistency.



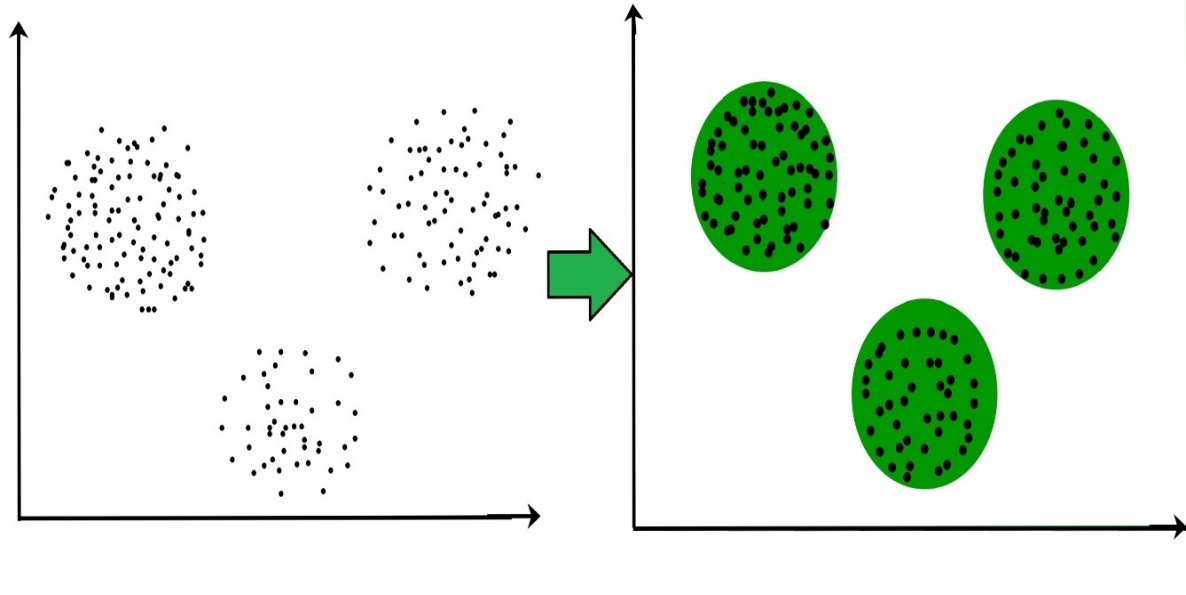
**Fig 1. k nearest neighbor clustering**

By and by, for most grouping target capacities and numerous informational collections the discrete streamlining approach can't be performed superbly as the comparing enhancement issue is NP hard. Rather, individuals resort to heuristics and acknowledge problematic arrangements. One methodology is to utilize neighborhood enhancement strategies conceivably finishing off with nearby minima as it were. This is the thing that occurs in the Kmeans calculation: despite the fact that the K-implies issue for ﬁxed K and ﬁxed measurement isn't NP hard, it is still unreasonably hard for being understood all inclusive practically speaking. Another methodology is to develop an unwinding of the first issue which can be unraveled efﬁciently (ghostly grouping is a model for this). For such heuristics, by and large one can't ensure how close the heuristic arrangement is to the ﬁnite test ideal. This circumstance is obviously unsuitable: as a rule, we neither have ensures on the ﬁnite test conduct of the calculation, nor on its factual consistency in the point of confinement. The accompanying elective methodology looks substantially more encouraging. Rather than endeavoring to tackle the discrete streamlining issue over the arrangement all things considered, and afterward depending on relaxations because of the hardness of this issue, we reverse the situation. Straightforwardly from the start, we just consider competitor segments in some confined class Fn containing just polynomial numerous capacities. At that point the discrete enhancement issue of limiting Qn over Fn isn't NP hard—officially it very well may be tackled in polynomially numerous means by attempting all competitors in Fn. From a hypothetical perspective this methodology has the bit of leeway that the subsequent grouping calculation has the capability of being reliable. Furthermore, this methodology additionally has points of interest by and by: as opposed to managing uncontrolled relaxations of the first issue, we confine the capacity class to some little subset Fn of "sensible" segments. Inside this subset, we at that point have unlimited oversight over the arrangement of the advancement issue and can ﬁnd the worldwide ideal. Put another way, one can likewise decipher this methodology as some controlled method to rough an answer of the NP hard enhancement issue on the ﬁnite test, with the positive reaction of complying with the standards of measurable learning hypothesis.

**CLUSTERING**

It is fundamentally an unaided learning technique. An unaided learning strategy is a technique where we draw references from datasets comprising of info information without marked reactions. For the most part, it is utilized as a procedure to discover significant structure, illustrative hidden procedures, generative highlights, and groupings innate in a lot of models.

Bunching is the errand of isolating the populace or information focuses into various gatherings to such an extent that information focuses in similar gatherings are progressively like other information focuses in a similar gathering and not at all like the information focuses in different gatherings. It is essentially an assortment of articles based on comparability and disparity between them.

For ex–The information focuses in the diagram beneath bunched together can be arranged into one single gathering. We can recognize the groups, and we can distinguish that there are 3 bunches in the underneath picture.

**Fig 2. Clustering**

**Classification**

Arrangement issues mean to recognize the attributes that demonstrate the gathering to which each case has a place. This example can be utilized both to comprehend the current information and to foresee how new occasions will carry on. Information mining makes grouping models by looking at previously arranged information (cases) and inductively finding a prescient design. These current cases may originate from an authentic database, for example, individuals who have just experienced a specific restorative treatment or moved to another long separation administration. They may originate from a trial wherein an example of the whole database is tried in reality and the outcomes used to make a classifier. For instance, an example of a mailing rundown would be sent an offer, and the aftereffects of the mailing used to build up a grouping model to be applied to the whole database. In some cases a specialist orders an example of the database, and this characterization is then used to make the model which will be applied to the whole database.

* 1. **OBJECTIVE:**

The principle target of this venture is to plan a calculation to utilize a database where the information focuses are isolated into a few classes to anticipate the characterization of another example point.

This task is mostly meant to give wellbeing administrations to the general population like giving best medicinal services emergency clinic area which is situated close to their environment. The informational collection contains hardly any examples for anticipating the closest and best medical clinic dependent on the financial backing, charges gave by emergency clinic, framework and closest area. This calculation will foresee the yield in quickest and proficient manner utilizing worked in preprocessing capacities.

The goals of the investigation are:

i. To assess to what degree k-Nearest-Neighbor classifier upgrade productivity and precision among patients looking for crisis treatment.

ii. To assess the elements influencing the execution of k-Nearest-Neighbor mining method in emergency clinics.

iii. Design a vault with effectively ordered information for simple information mining.

**1.2 Problem Definition:**

The main problems faced by public is in finding best hospitals for good and efficient treatment and also finding nearest hospitals in case of any emergency. Our main objective of this project is to provide efficient information to the public about nearby hospitals based on their level of disease and charges for the treatment. People find difficulty in finding best hospitals based on their expenses. In order to provide them a solution we have designed an algorithm for finding hospitals based on their region and expense.

**CHAPTER -2**

**Literature Review:**

KNN arrangement was created from the need to perform discriminant investigation when dependable parametric evaluations of likelihood densities are obscure or hard to decide. In an unpublished US Air Force School of Aviation Medicine report in 1951, Fix and Hodges presented a non-parametric strategy for design arrangement that has since become known the k-closest neighbor rule. They acquainted a novel methodology with nonparametric grouping by depending on the 'separation' between focuses or disseminations. The essential thought is to order a person to the populace whose example contains most of 'closest neighbors. Later in 1967, a portion of the conventional properties of the k-closest neighbor rule were turned out, for example gave upper limits for the point of confinement of the danger of closest neighbor classifiers. When such conventional properties of k-closest neighbor arrangement were set up, a long queue of examination resulted including new dismissal draws near, refinements as for Bayes mistake rate, separation weighted methodologies, and delicate registering techniques. Wagner and Fritz treated assembly of the restrictive mistake rate when K =1. Devroye and Wagner created and talked about hypothetical properties, especially issues of scientific consistency, for K-closest neighbor rules. Devroye found an asymptotic headed for the lament as for the Bayes classifier. Devroye et al. gave an especially broad portrayal of solid consistency for closest neighbor techniques. Psaltis, Snapp and Venkatesh summed up the aftereffects of Cover to general measurement, and Snapp and Venkatesh further stretched out the outcomes to the instance of different classes. Bax gave probabilistic limits for the restrictive blunder rate for the situation where K= 1. Kulkarni and Posner tended to closest neighbor strategies for very broad ward information, and Holst and Irle gave formulae to the furthest reaches of the mistake rate on account of ward information. Related research incorporates that of Györfi who explored the pace of assembly to the Bayes hazard when K watches out for endlessness as T increments.

**CHAPTER- 3**

**Methodology:**

**3.1) Classification Algorithms:**

1. **Logistic calculation:** In this calculation, the probabilities depicting the potential results of a solitary preliminary are displayed utilizing a strategic capacity.
2. **Naïve bayes calculation:** Naive Bayes calculation depends on Bayes' hypothesis with the presumption of autonomy between each pair of highlights. It requires limited quantity of preparing information to appraise the parameters.
3. **K-Nearest Neighbor Algorithm:** Neighbors based grouping is a kind of languid learning as it doesn't endeavor to build a general inside model, yet just stores examples of the preparation information. This calculation is easy to execute, vigorous and compelling for huge information preparing sets.
4. **Decision tree**: a choice tree delivers an arrangement of decides that can be utilized to group the information. Choice trees can be insecure in light of the fact that little varieties in the information In this venture we utilized KNN calculation to order the information dependent on input tests.

**3.2)** **KNN Algorithm**

**Our calculation continues in two stages:**

• It ﬁrst ﬁnds the bunch in the progressive system, wherein the inquiry point has greatest probability of event.

• KNN is applied to focuses present in the group, and weighted mean of the k closest neighbors of question point in the bunch is cited as yield.

Size of the got group is less contrasted with the size of whole dataset, along these lines our calculation diminish the quest space for K-Nearest Neighbor calculation. Presently we clarify in insight concerning the grouping stage (pre-preparing step) ﬁrst and later illuminate the genuine calculation.

**PREPROCESSING STEP:**

In the pre-preparing step, information is progressively bunched and mean an incentive for every one of the group is determined and put away. Every hub in this chain of importance comprise of bunches containing information point, which are recursively partitioned into three youngster groups as we descend the progression. We utilize the way that comparable occurrences have comparable estimation of reaction variable (fundamental relationship on which KNN works), while choosing the group focus. Each group hub is arranged dependent on the estimation of reaction variable, For the third bunch, mean of the other two group's middle is taken as focus. The inside is called as introductory focus. Every one of the focuses present in the bunch hub are partitioned into kid group, in view of to which kid group focus the fact of the matter is nearest. We expect to isolate each bunch hubs in a manner, that every kid hub contains a large portion of the information present in the parent hub. In any case, at each level we have added an additional kid group to make the division of focuses smooth, this bunch likewise contain directs having a place toward other group that lies at its limit with other group. This will help to appropriately arrange the inquiry occurrence that lie at the limit of groups

**3.3) FLOW CHART:**

Step 1: load the training data set

Step 2: define the value of K

Step 3: dividing data set into training and testing groups.

Step 3: preprocessing of training data values

Step 5: preprocessing of testing data values

Step 6: KNN classification

Step 4: putting the distances in shorted list

Step 5: Defining the class of the sample based on the k value

Step 6: finally predicting the class of each and every samples in the data set

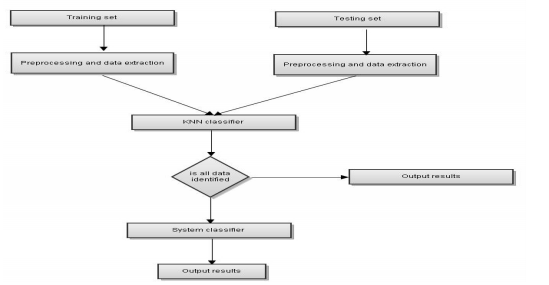


Fig 3. KNN data recognition model for patients.

The proposed k-Nearest Neighbour model is given in Figure 5. In the proposed method, the given data is preprocessed to extract all the metadata. KNN is used to find the closest neighbors of the given data with all the available training data. If a label is found then the algorithm quits, otherwise the system classifier is applied. The proposed algorithm was used to recognize the object. The results are compared to those obtained with single system classifier and KNN.

**CHAPTER-4**

**SOFTWARE TOOLS:**

**4.1) Software description:**

Spyder is an open source cross-stage coordinated improvement condition (IDE) for logical programming in the Python language. Spyder incorporates with various unmistakable bundles in the scientific Python stack, including NumPy, SciPy, Matplotlib, Pandas, IPython, SymPy and Cython, just as other open source programming. It is discharged under the MIT permit. At first made and created by Pierre Raybaut in 2009, since 2012 Spyder has been kept up and persistently improved by a group of logical Python engineers and the network.

Spyder is extensible with first-and outsider modules, incorporates support for intelligent devices for information investigation and installs Python-explicit code quality affirmation and reflection instruments, for example, Pyflakes, Pylint and Rope. It is accessible cross-stage through Anaconda, on Windows, on macOS through MacPorts, and on significant Linux dispersions, for example, Arch Linux, Debian, Fedora, Gentoo Linux, openSUSE and Ubuntu.

Spyder utilizes Qt for its GUI, and is intended to utilize both of the PyQt or PySide Python ties. QtPy, a flimsy deliberation layer created by the Spyder venture and later embraced by various different bundles, gives the adaptability to utilize either backend.

**4.2) Software language depiction:**

Python is a simple to adapt, ground-breaking programming language. It has proficient elevated level information structures and a straightforward yet compelling way to deal with object-arranged programming. Python's rich grammar and dynamic composing, together with its deciphered nature, make it a perfect language for scripting and fast application improvement in numerous regions on most stages.

The Python mediator and the broad standard library are openly accessible in source or twofold structure for every single significant stage from the Python Web webpage, and might be unreservedly disseminated. A similar site likewise contains dispersions of and pointers to many free outsider Python modules, projects and apparatuses, and extra documentation.

**CHAPTER-5**

**Experimental Analysis and Results:**

**5.1) Data set description:**

The sample data set is used in this project. It consists of eleven columns, in which one is the output column and remaining all are input samples.

Input samples:

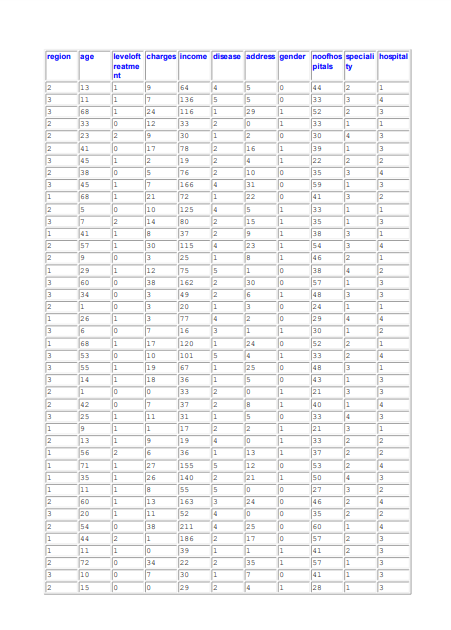
1. Region
2. Age
3. Level of treatment in the hospitals
4. Charges collected in the hospitals
5. Income of the patient
6. Disease caused to the patient
7. Address of the patient
8. Gender
9. No of hospitals in the locality

Output sample:

Economy of the hospital (healthcare industry)

**Based on the above input samples, the output (status of healthcare industry) is determined and this information is provided to the public by posting this details in the web page or by android app.**

**Data set:**



**Fig 4. sample data set**

In this project we used k=4 to identify status of the healthcare center. The parameters used to identify customer category is shown in above figure.

The health care center is divided into 4 categories They are

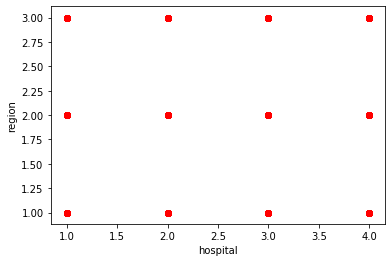
1. Cluster 1: health care centre with high charges
2. Cluster 2: government health care centre
3. Cluster 3: health care centre with medium charges
4. Cluster 4: health care centre with low charges

Based on above 9 parameters the patient may prefer into any one the above category.

**5.2) OUTPUT GRAPHS:**

**Region:**

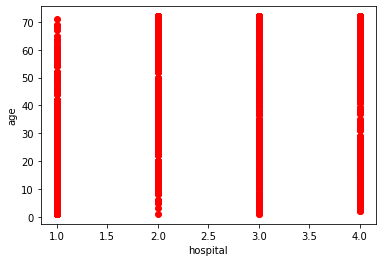
The below graph predicts health care centers based on region. Where every region has all types of health care centers. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 5. Health care centers based on region**

**Age:**

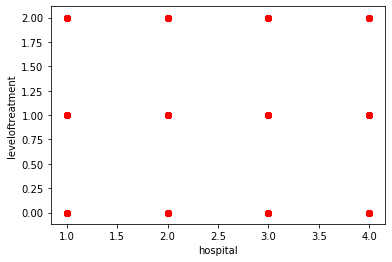
The below graph predicts health care centers based on Age. Where Age includes all patients including children. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 6. Health care center based on Age.**

**Level of Treatment:**

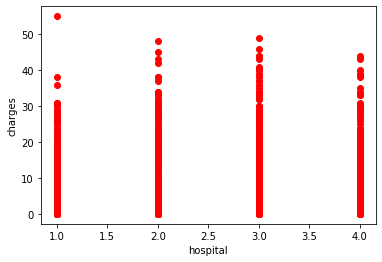
The below graph predicts health care centers based on Level of treatment. Where level of treatment indicates the equipment, standard of the health care center and other facilities available in the health care center. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 7. Health care center based on level of treatment**

**Charges:**

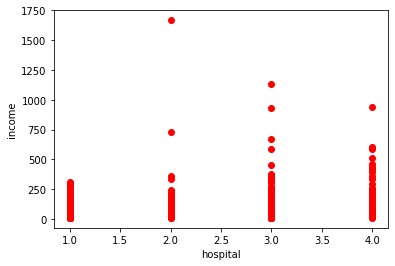
The below graph predicts health care centers based on charges. Where charges include the money required for any operation or treatment. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital including charges offered by the health care center. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 8. Health care center based on charges**

**Income:**

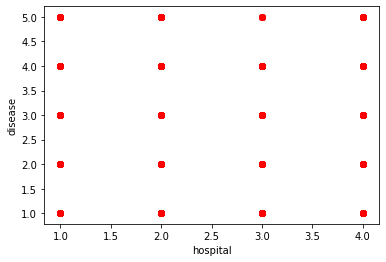
The below graph predicts health care centers based on annual income of the patient. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital including charges offered by the health care center. The app or web page shows health care center based on few fixed annual income levels. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 9. Health care center based on Income.**

**Disease:**

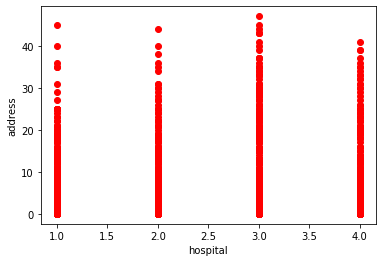
The below graph predicts health care centers based on type of disease or problem. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital including charges offered by the health care center. The app or web page shows health care center based on different specialists available in the health care center. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 10. Health care center based on disease or problem**

**Address:**

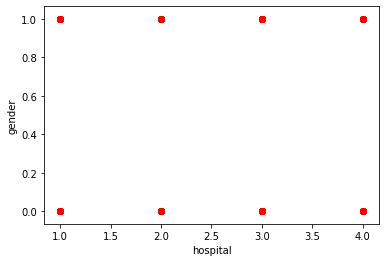
The below graph predicts health care centers based on address of the patient. Where address include pin code. But based on other input samples the health care center is shown to the patient in the app or web page



**Fig 11. Health care center based on address**

**Gender:**

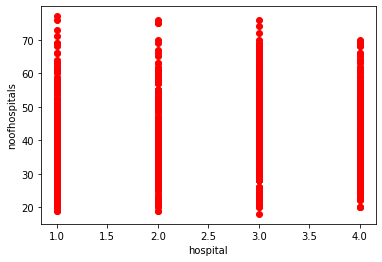
The below graph predicts health care centers based on Gender. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital or hospitals for females including charges offered by the health care center. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 12. Health care center based on Gender**

**Number of hospitals:**

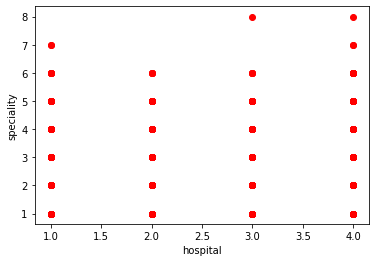
The below graph predicts number of health care centers in a region. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital or hospitals for females including charges offered by the health care center. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 13. Total number of hospitals in a region**

**Speciality:**

The below graph predicts health care centers based on requirement of the patient. Based on the input sample the patient gets the details of the health care center it may be children’s hospital or general hospital or hospitals for females including charges offered by the health care center. But based on other input samples the health care center is shown to the patient in the app or web page.



**Fig 14. Health care center based on its speciality**

**OUTPUT:**

Train set: (900, 11) (900,)

Test set: (100, 11) (100,)

[1 1 1 2 4 3 3 4 2 4 1 4 2 1 3 1 3 4 3 4 3 3 3 1 1 3 1 1 1 2 1 1 4 4 3 2 2

1 2 3 1 4 2 1 2 1 4 1 4 4 2 1 2 3 2 3 3 3 4 1 1 4 2 3 3 3 1 2 2 1 1 3 4 2

2 1 1 2 2 3 3 3 1 4 1 2 3 4 1 1 2 2 1 1 1 1 2 2 3 1]

The output contains a training set and test set. 900 inputs are given as training set and remaining 100 are considered under testing set. Under training set all

**5.3) Advantages:**

• It is anything but difficult to actualize. There are just two parameters required to execute K Nearest Algorithm for example the estimation of K.

• Effective If The Training Data Is Large

• Robust To Noisy Training Data

• Simplicity, viability, instinct and focused order execution in numerous spaces

**5.4) Disadvantages:**

• Distance based learning isn't clear which kind of separation to utilize and which credit to use to create the best outcomes.

• Computation cost is very high since we have to process separation of each inquiry example to all preparation tests.

**5.5) Applications:**

KNN as an information mining procedure has a wide assortment of utilizations in arrangement just as relapse. A portion of the uses of this strategy are referenced beneath:

**Content mining:** The KNN calculation is one of the most prominent calculations for content arrangement or content mining. Probably the latest chips away at this theme are for example. Various quantities of closest neighbors are utilized for various classes in this methodology, as opposed to a fixed number over all classes. Along these lines, the main parameter that should be picked by the client when utilizing KNN, the K esteem, turns out to be less reasonable and henceforth it shouldn't be deliberately picked as in the standard calculation. To be sure, the likelihood that an obscure example has a place with a class is figured by utilizing just some top Kn closest neighbors for that class. The Kn esteem is gotten from K as per the size of the comparing class in the preparation set. This adjusted KNN was effective and less reasonable to the K esteems when applied to content mining issues.

**Horticulture:** In general,KNN is applied not exactly other information mining systems in agribusiness related fields. It has been applied, for example, for recreating every day precipitations and other climate factors. Another intriguing application is the assessment of woods inventories and for evaluating backwoods factors. In these applications, satellite symbolism is utilized, with the point of mapping the land spread and land use with scarcely any discrete classes. Different utilizations of the k-NN strategy in agribusiness incorporate atmosphere guaging and evaluating soil water parameters.

**Account:** Data mining as a procedure of finding valuable examples and connections has its own specialty in monetary demonstrating. Like other computational strategies pretty much every information mining strategy and method has been utilized in monetary demonstrating. An inadequate rundown incorporates an assortment of straight and nonlinear models multi-layer neural systems, k-implies and various leveled bunching, k-closest neighbors, choice tree investigation, relapse (calculated relapse, general numerous relapse), ARIMA, head segment examination, and Bayesian learning. Securities exchange estimating is one of the most center money related assignments of KNN. Securities exchange determining incorporates revealing business sector patterns, arranging speculation methodologies, distinguishing the best time to buy the stocks, and what stocks to buy. Some of different utilizations of KNN in fund are referenced underneath:

1. **Forecasting financial exchange:** Predict the cost of a stock, based on organization execution measures and monetary information.
2. Currency swapping scale
3. Bank liquidations
4. Understanding and overseeing budgetary hazard
5. Trading prospects
6. Credit rating
7. Loan the executives
8. Bank client profiling

i) Money laundering examinations

**CHAPTER 6**

**Conclusion and Future Scope:**

We can reason that it is conceivable to improve the order exactness of KNN calculation, utilizing the channel strategies for diminishing the dimensionality of the information. To demonstrate this, we actualized and observationally tried channel strategies for lessening the dimensionality of the information. Test results show that the techniques applied viably add to the identification and end of immaterial, repetitive information and clamor in the information. As a rule the channel strategies select pertinent credits and add to the more prominent grouping exactness. In further research it is intriguing to apply different procedures to tackle the issue of dimensionality decrease of information, for example, wrapper strategies and extraction of qualities and investigate and think about the impacts of their usage on KNN calculation. These procedures could likewise improve the presentation of KNN calculation. Class Based Weighted K-Nearest Neighbor: Rather than just taking a normal of the coefﬁcient estimation of k/m closest neighbor for every one of the class, on the off chance that we utilize some heuristic capacity to ascertain the normal (like dependent on the separation normal), we may perform better for an assortment of datasets.

At present our calculation can anticipate the area of closest social insurance focuses with fulfilling exactness of forecast. In future the accompanying things can be executed in this examination:

1. KNN can be joined with some different methods, for example, Fuzzy Logic which can build the exactness of forecast.

ii. The product could be inserted with equipment and utilized as a total unit of expectation. Extra component supportive to ranchers can be executed, for example, forecast of sort of harvest that ought to be planted base on the anticipated estimation of barometrical parameters.

**CHAPTER 7**

**References:**

1. Ben-David. A framework for statistical clustering with constant time approximation algorithms for k-median and k-means clustering. Machine Learning, 66:243 – 257, 2007.
2. Buhmann. Empirical risk approximation: An induction principle for unsupervised learning. Technical report, University of Bonn, 1998.
3. Czumaj and C. Sohler. Sublinear-time approximation algorithms for clustering via random sampling. Random Struct. Algorithms, 30(1-2):226–256, 2007.
4. Devroye, L. Gy¨orﬁ, and G. Lugosi. A Probabilistic Theory of Pattern Recognition. Springer, New York, 1996.
5. Figueiredo and A. Jain. Unsupervised learning of ﬁnite mixture models. PAMI, 24(3):381–396, 2002.
6. Fraley and A. Raftery. How many clusters? Which clustering method? Answers via model-based cluster analysis. Comput. J, 41(8):578–588, 1998.
7. Fritz. Distribution-free exponential error bound for nearest neighbor pattern classiﬁcation. IEEE Trans. Inf. Th., 21(5):552 – 557, 1975.
8. Garey, D. Johnson, and H. Witsenhausen. The complexity of the generalized Lloyd - max problem (corresp.). IEEE Trans. Inf. Theory, 28(2):255–256, 1982.
9. Gr¨unwald. The Minimum Description Length Principle. MIT Press, Cambridge, MA, 2007.
10. Guattery and G. Miller. On the quality of spectral separators. SIAM Journal of Matrix Anal. Appl., 19(3):701 – 719, 1998.
11. Hartigan. Consistency of single linkage for high-density clusters. JASA, 76(374):388 – 394, 1981.
12. Hartigan. Statistical theory in clustering. Journal of Classiﬁcation, 2:63 – 76, 1985.
13. Inaba, N. Katoh, and H. Imai. Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering. In Proceedings of the 10th Annual Symposium on Computational Geometry, pages 332–339. ACM Press, Stony Brook, USA, 1994.
14. van der Vaart and J. A. Wellner. Weak Convergence and Empirical Processes. Springer, New York, 1996.
15. Vapnik. The Nature of Statistical Learning Theory. Springer Verlag, New York, 1995.
16. von Luxburg. A tutorial on spectral clustering. Statistics and Computing, 17(4):395 – 416, 2007.

**APPENDIX-A:**

**SOURCE CODE:**

**Software used: SPYDER**

**Software language: Python**

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.ticker import NullFormatter

import pandas as pd

import matplotlib.ticker as ticker

from sklearn import preprocessing

df = pd.read\_csv('teleCust1000t.csv')

df.head()

#df1=pd.read\_csv('tele.csv')

#df1.head()

df['custcat'].value\_counts()

plt.scatter(df.custcat,df.income,color='red')

plt.xlabel('cat')

plt.ylabel('income')

plt.show()

df.columns

X = df[['region', 'tenure','age', 'marital', 'address', 'income', 'ed', 'employ','retire', 'gender', 'reside']] .values #.astype(float)

y = df['custcat'].values

#X1=df1[['region', 'tenure','age', 'marital', 'address', 'income', 'ed', 'employ','retire', 'gender', 'reside']] .values

#s=df1['custcat'].values

X = preprocessing.StandardScaler().fit(X).transform(X.astype(float))

#X1 = preprocessing.StandardScaler().fit(X1).transform(X1.astype(float))

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

X\_train, X\_test, y\_train, y\_test = train\_test\_split( X, y, test\_size=0.1, random\_state=4)

print ('Train set:', X\_train.shape, y\_train.shape)

print ('Test set:', X\_test.shape, y\_test.shape)

from sklearn.neighbors import KNeighborsClassifier

k = 4

#Train Model and Predict

neigh = KNeighborsClassifier(n\_neighbors = k).fit(X\_train,y\_train)

yhat = neigh.predict(X\_test)

print(yhat)

yhat[0:5]

**APPENDIX-B**

**SAMPLE DATA SET:**

