**IORD Journal of Science & Technology**
**E-ISSN: 2348-0831**
**Volume 3, Issue 3 (MAR-APRIL 2017 PP98 -103**
**IMPACT FACTOR 1.719**
**www.iord.in**

# "Implementation of KNN using Hybrid K-Nearest Neighbour Method "

**Miss. Yamini Laxane [1]**

**[1](PG Student,VITNagpur)**

**Abstract Abstract**

***Today data mining is used in many applications areas medical, scientific research, banking and many more. From last decade, Internet has given rise to many privacy issues. To solve these issues many theoretical and practical solutions to the classification problem have been proposed using different security models. However, cloud computing allow users to outsource their data to cloud. User prefers to encrypt the data before storing it on cloud, but performing any classification on encrypted data is main issue. Today's privacy-preserving classification techniques are not useful for encrypted data, so here we uses k-NN classifier over encrypted data in the cloud. The proposed technique protects the security of data, privacy of user's input query, and hides the access patterns. Our aim is to develop a secure k-NN classifier on encrypted data using the semi-honest model. Also, efficiency of K nearest neighbor classification is analyzed using real world data set under different parameters conditions. The proposed protocol protects the confidentiality of data, privacy of user's input query, and hides the data access patterns. To the best of our knowledge, our work is the first to develop a secure k-NN classifier over encrypted data under the semi-honest model. Also, we empirically analyze the efficiency of our proposed protocol using a real-world dataset under different parameter settings.***

***Keywords— (Searching Time,Existing System,Proposed System,Implementation, KNN,Hybrid KNN)***

## 1. INTRODUCTION

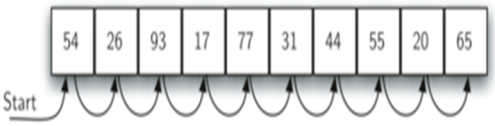
Application will contain large number of database of employees, which I secured on cloud, But if an authorized user (agent) wants to access the database, we will provide one user interface where he will be able to search the employee database with some searching parameters, and ate the end of result we will use KNN algorithm to show related results those related results will be according to the KNN algorithm. KNN algorithm is one of the simplest classification algorithm. Even with such simplicity, it can give highly competitive results. KNN algorithm can also be used for regression problems. The only difference from the discussed methodology will be using averages of nearest neighbors rather than voting from nearest neighbors. KNN can be coded in a single line on R. I am yet to explore how can we use KNN algorithm on SAS.

## 2. Existing System

The existing system contain searching the data from the database using different queries, not any particular algorithm. The data sorting is done using only query and not any rules or protocols like algorithm.

When data items are stored in a collection such as a list, we say that they have a linear or sequential relationship. Each data item is stored in a position relative to the others. In Python lists, these relative positions are the index values of the individual items. Since these index values are ordered, it is possible for us to visit them in sequence. This process gives rise to our first searching technique, the **sequential search.**

Starting at the first item in the list, we simply move from item to item, following the underlying sequential ordering until we either find what we are looking for or run out of items. If we run out of items, we have discovered that the item we were searching for was not present.




The Python implementation for this algorithm is shown in CodeLens 1. The function needs the list and the item we are looking for and returns a boolean value as to whether it is present. The boolean variable foundis initialized to False and is assigned the value True if we discover the item in the list.

**Searching Time**

```
defsequentialSearch(alist, item):

    pos = 0

    found = False


    whilepos<len(alist) and not found:

    ifalist[pos] == item:

    found = True
```

```
        else:

            pos = pos+1


        return found


testlist = [1, 2, 32, 8, 17, 19, 42, 13, 0]

        print(sequentialSearch(testlist, 3))

        print(sequentialSearch(testlist, 13))
```

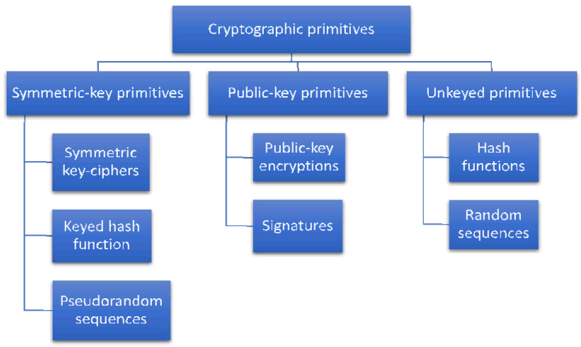
### 3. Problem Definition

Suppose Alice owns a database D of n records t1; . . . ; tn and m þ 1 attributes. Let ti;j denote the jth attribute value of record ti. Initially, Alice encrypts her database attributewise, that is, she computes Epkðti;jÞ, for 1 i n and 1 j m þ 1, where column ðm þ 1Þ contains the class labels. We assume that the underlying encryption scheme is semantically secure [4]. Let the encrypted database be denoted by D0. We assume that Alice outsources D0 as well as the future classification process to the cloud. Let Bob be an authorized user who wants to classify his input record q ¼ hq1; . . . ; qmi by applying the k-NN classification method based on D0. We refer to such a process as privacy-preserving k-NN (PPkNN) classification over encrypted data in the cloud. Formally, we define the PPkNN protocol as: PPkNNðD0; qÞ ! cq; where cq denotes the class label for q after applying k-NN classification method on D0 and q

Suppose Alice owns a database D of n records t1; . . . ; tn and m þ 1 attributes. Let ti;j denote the jth attribute value of record ti. Initially, Alice encrypts her database attributewise, that is, she computes Epkðti;jÞ, for 1 i n and 1 j m þ 1, where column ðm þ 1Þ contains the class labels. We assume that the underlying encryption scheme is semantically secure [4]. Let the encrypted database be denoted by D0. We assume that Alice outsources D0 as well as the future classification process to the cloud. Let Bob be an authorized user who wants to classify his input record q ¼ hq1; . . . ; qmi by applying the k-NN classification method based on D0. We refer to such a process as privacy-preserving k-NN (PPkNN) classification over encrypted data in the cloud. Formally, we define the PPkNN protocol as: PPkNNðD0; qÞ ! cq; where cq denotes the class label for q after applying k-NN classification method on D0 and q.

#### 3.1 PRIVACY-PRESERVING PRIMITIVES

Here we present a set of generic sub-protocols that will be used in constructing our proposed k-NN protocol in Section 5. All of the below protocols are considered under two-party semi-honest setting. In particular, we assume the existence of two semi-honest parties P1 and P2 such that the Paillier's secret key sk is known only to P2 whereas pk is public.




### 4. Methodology:

**PHP:**

PHP is an HTML-embedded scripting language. Much of its syntax is borrowed from C, Java and Perl with a couple of unique PHP-specific features thrown in. The goal of the language is to allow web developers to write dynamically generated pages quickly. PHP stands for *PHP: Hypertext Preprocessor*. This confuses many people because the first word of the acronym is the acronym. This type of acronym is called a recursive acronym.PHP code may be embedded into HTML or HTML5 markup, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as a de facto standard. Since 2014 work has gone on to create a formal PHP specification.[9]

**HTML**

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a webserver or from local storage and render them into multimedia web pages.

HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects, such as interactive forms, may be embedded into the rendered page. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. Tags such as <img /> and <input /> introduce content into the page directly. Others such as ... surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.HTML can embed programs written in a scripting language such as JavaScript which affect the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, has encouraged the use of CSS over explicit presentational HTML since 1997

**Javascript**

JavaScript (/ˈdʒɑːvə skrɪpt/), often abbreviated as JS, is a high-level, dynamic, weakly typed, object-based, multi-paradigm, and interpreted programming language. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Webcontent production. It is used to make webpages interactive and provide online programs, including video games. The majority of websites employ it, and all modern web browsers support it without the need for plug-ins by means of a built-in JavaScript engine. Each of the many JavaScript engines represent a different implementation of JavaScript, all based on the ECMAScript specification, with some engines not supporting the spec fully, and with many engines supporting additional features beyond ECMA.As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has an API for working with text, arrays, dates, regular expressions, and basic manipulation of the DOM, but does not include any I/O, such as networking, storage, or graphics facilities, relying for these upon the host environment in which it is embedded.Initially only implemented client-side in web browsers, JavaScript engines are now embedded in many other types of host software, including server-side in web servers and databases, and in non-web programs such as word processors and PDF software, and in runtime environments that make JavaScript available for writing mobile and desktop applications, including desktop widgets.Although there are strong outward similarities between JavaScript and Java, including language name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design; JavaScript was influenced by programming languages such as Selfand Scheme.mperative and structured[edit]

JavaScript supports much of the structured programming syntax from C (e.g., if statements, while loops, switch statements, do while loops, etc.). One partial exception is scoping: JavaScript originally had only function scoping with var. ECMAScript 2015 added keywords let and const for block scoping, meaning JavaScript now has both function and block scoping. Like C, JavaScript makes a distinction between expressions and statements. One syntactic difference from C is automatic semicolon insertion, which allows the semicolons that would normally terminate statements to be omitted.[26]DynamicTyping As with most scripting languages, JavaScript is dynamically typed; a type is associated with each value, rather than just with each expression. For example, a variable that is at one time bound to a number may later be re-bound to a string.[27] JavaScript supports various ways to test the type of an object, including duck typing.[28]

CSS

**Cascading** Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language. Although most often used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications. CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.Separation of formatting and content makes it possible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. It can also display the web page differently depending on the screen size or viewing device. Readers can also specify a different style sheet, such as a CSS file stored on their own computer, to override the one the author specified.Changes to the graphic design of a document (or hundreds of documents) can be applied quickly and easily, by editing a few lines in the CSS file they use, rather than by changing markup in the documents.The CSS specification describes a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called *cascade*, priorities (or *weights*) are calculated and assigned to rules, so that the results are predictable.

## 5. Proposed System

In the first phase of the application, we have worked on the Secured Login of the authorized user who will be able to see the result of KNN algorithm.

Login is secured with the md5 encryption in PHPMD5 is one in a series of message digest algorithms designed by Professor Ronald Rivest of MIT (Rivest, 1992). When analytic work indicated that MD5's predecessor MD4 was likely to be insecure, Rivest designed MD5 in 1991 as a secure replacement. (Hans Dobbertin did indeed later find weaknesses in MD4.)In 1993, Den Boer and Bosselaers gave an early, although limited, result of finding a "pseudo-collision" of the MD5 compression function; that is, two different initialization vectors that produce an identical digest.In 1996, Dobbertin announced a collision of the compression function of MD5 (Dobbertin, 1996). While this was not an attack on the full MD5 hash function, it was close enough for cryptographers to recommend switching to a replacement, such as SHA-1 or RIPEMD-160.The size of the hash value (128 bits) is small enough to contemplate a birthday attack. MD5CRK was a distributed project started in March 2004 with the aim of demonstrating that MD5 is practically insecure by finding a collision using a birthday attack.

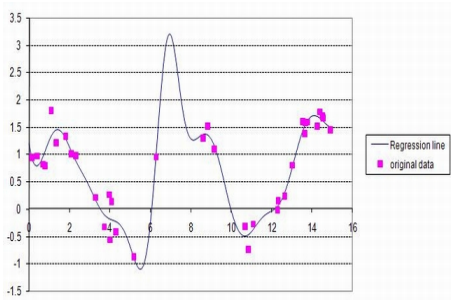
### 5.1 Hybrid KNN Algorithim

Conventional k -nearest neighbor (KNN) classification approaches have several limitations when dealing with some problems caused by the special datasets, such as the sparse problem, the imbalance problem, and the noise problem. In this paper, we first perform a brief survey on the recent progress of the KNN classification approaches. Then, the hybrid KNN (HBKNN) classification approach, which takes into account local and global information of the query sample, is designe address the problems raised from the special datasets. In following, the random subspace ensemble framework based HBKNN (RS-HBKNN) classifier is proposed to perf classification on the datasets with noisy attributes in the hi dimensional space. Finally, the nonparametric tests are propo to be adopted to compare the proposed method with other classification approaches over multiple datasets. The experiments on the real-world datasets from the Knowledge Extraction based on Evolutionary Learning dataset repository demonstrate that RS-HBKNN works well on real datasets, and outperforms most of the state-of-the-art classification approaches.

### 5.2 Feature Application:

The application contain admin login, where he can create database of his employees, and can search them using knearest algorithm. Admin can create, update delete any numbers of employee data and search them, searching will be done using kNearest algorithm.

In a project on education and relating to how students and novice programmers learn to program, in several classes of novice first year students, an assignment is given to all the students to perform in a certain time period. Each compilation of the student is captured, this means all code and content and different file classes. This is a "snap-shot" of each program. So for the purposes of this tutorial, I will use a sample data of number of compiler errors; time in performing the exercise, and the number of compiles. Each of these students have a final grade for the course. This will be used to determine how well they have done on a particular assignment.

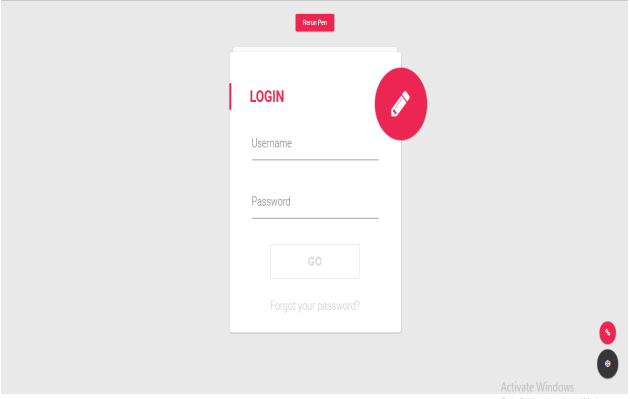


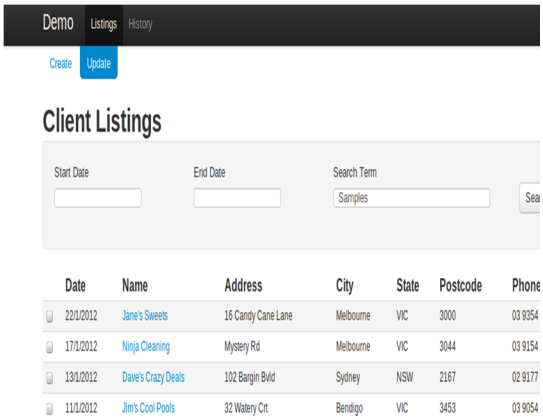

The following is a sample data which is being used as the training data. Again supervised learning requires you to know what the "category" of each training data is and to what category they belong to. In terms of this project, my categories are the grades of the course from A, A-, B+, B, B-, C, F.

The algorithm on how to compute the K-nearest neighbors is as follows:

1. Determine the parameter K = number of nearest neighbors beforehand. This value is all up to you. Calculate the distance between the query-instance and all the training samples. You can use any distance algorithm. Sort the distances for all the training samples and determine the nearest neighbor based on the K-th minimum distance. Since this is supervised learning, get all the Categories of your training data for the sorted value which fall under K.

## 6. Result Analysis






**References**

1. P. Mell and T. Grance, "The NIST definition of cloud computing (draft)," NIST Special Publication, vol. 800, p. 145, 2011
2. S. De Capitani di Vimercati, S. Foresti, and P. Samarati "Managing and accessing data in the cloud: Privacy risks an approaches," in Proc. 7th Int. Conf. Risk Security Internet Syst., 2012 pp. 1–9 1272 IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 27, NO. 5, MAY 2015
3. P. Williams, R. Sion, and B. Carbunar, "Building castles out o mud: Practical access pattern privacy and correctness o untrusted storage," in Proc. 15th ACM Conf. Comput. Commun Security, 2008, pp. 139–148
4. P. Paillier, "Public key cryptosystems based on composite degree residuosity classes," in Proc. 17th Int. Conf. Theory Appl. Cryptographi Techn., 1999, pp. 223–238.
5. B. K. Samanthula, Y. Elmehdwi, and W. Jiang, "k-nearest neighbor classification over semantically secure encrypted relational data, eprint arXiv:1403.5001, 2014.
6. C. Gentry, "Fully homomorphic encryption using ideal lattices, in Proc. 41st Annu. ACM Sympos. Theory Comput., 2009, pp. 169 178.
7. C. Gentry and S. Halevi, "Implementing gentry's fully-homomorphi encryption scheme," in Proc. 30th Annu. Int. Conf. Theor Appl. Cryptographic Techn.: Adv. Cryptol., 2011, pp. 129–148.
8. A. Shamir, "How to share a secret," Commun. ACM, vol. 22 pp. 612–613, 1979.
9. D. Bogdanov, S. Laur, and J. Willemson, "Sharemind: A framewo for fast privacy preserving computations," in Proc. 13th Eur Symp. Res. Comput. Security: Comput. Security, 2008, pp. 192–206
10. R. Agrawal and R. Srikant, "Privacy-preserving data mining, ACM Sigmod Rec., vol. 29, pp. 439–450, 2000.
11. Y. Lindell and B. Pinkas, "Privacy preserving data mining," Proc. 20th Annu. Int. Cryptol. Conf. Adv. Cryptol., 2000, pp. 36–54.
12. P. Zhang, Y. Tong, S. Tang, and D. Yang, "Privacy preserve Naive Bayes classification," in Proc. 1st Int. Conf. Adv. Data Minin Appl., 2005, pp. 744–752.
13. A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privac preserving mining of association rules," Inf. Syst., vol. 29, no. pp. 343–364, 2004.
14. R. J. Bayardo and R. Agrawal, "Data privacy through optimal kanonymization, in Proc. IEEE 21st Int. Conf. Data Eng., 2005 pp. 217–228.
15. H. Hu, J. Xu, C. Ren, and B. Choi, "Processing private queries ove untrusted data cloud through privacy homomorphism," in Proc IEEE 27th Int. Conf. Data Eng., 2011, pp. 601–612.
16. M. Kantarcioglu and C. Clifton, "Privately computing a distribut k-nn classifier," in Proc. 8th Eur. Conf. Principles Practi Knowl. Discovery Databases, 2004, pp. 279–290.
17. L. Xiong, S. Chitti, and L. Liu, "K nearest neighbor classification across multiple private databases," in Proc. 15th ACM Int. Con Inform. Knowl. Manage., 2006, pp. 840–841.
18. Y. Qi and M. J. Atallah, "Efficient privacy-preserving k-neare neighbor search," in Proc. IEEE 28th Int. Conf. Distrib. Compu Syst., 2008, pp. 311–319.
19. R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order preserving encryption for numeric data," in Proc. ACM SIGMOD Int. Con Manage. Data, 2004, pp. 563–574
20. H. Hacig€um€u¸s, B. Iyer, C. Li, and S. Mehrotra, "Executing sql ove encrypted data in the database-service-provider model," in Proc ACM SIGMOD Int. Conf. Manage. Data, 2002, pp. 216–227.
21. B. Hore, S. Mehrotra, M. Canim, and M. Kantarcioglu, "Secur multidimensional range queries over outsourced data," VLDB J.,vol. 21, no. 3, pp. 333–358, 2012.\
22. W. K. Wong, D. W.-l. Cheung, B. Kao, and N. Mamoulis, "Secur kNN computation on encrypted databases," in Proc. ACM SIGMOInt. Conf. Manage. Data, 2009, pp. 139–152
23. X. Xiao, F. Li, and B. Yao, "Secure nearest neighbor revisited," I Proc. IEEE Int. Conf. Data Eng., 2013, pp. 733–744.
24. Y. Elmehdwi, B. K. Samanthula, and W. Jiang, "Secure k-neares neighbor query over encrypted data in outsourced environments, in Proc. IEEE 30th Int. Conf. Data Eng., 2014, pp. 664–675.
25. A. C. Yao, "Protocols for secure computations," in Proc. 23r Annu. Symp. Found. Comput. Sci., 1982, pp. 160–164.
26. O. Goldreich, S. Micali, and A. Wigderson, "How to play an mental game—A completeness theorem for protocols with hones majority," in Proc. 19th Annu. ACM Symp. Theory Comput., 1987

**IORD Journal of Science & Technology**
**E-ISSN: 2348-0831**
**Volume 3, Issue 3 (MAR-APRIL 2017 PP98 -103**
**IMPACT FACTOR 1.719**
**www.iord.in**