DELHI TECHNOLOGICAL UNIVERSITY

DELHI-110042

MACHINE LEARNING PROJECT

(CO-327)

PREDICTION OF PHISHING WEBSITES USING MACHINE LEARNING

SUBMITTED TO:-

Ms. Indu Singh

Assisstant Professor

CSE Department DTU

SUBMITTED BY:-

Siddhant Jain

(2K16/CO/314)

Sparsh Shubham Sinha

(2K16/CO/319))

Table of Content

Declaration

We hereby certify that the work which is presented in minor project entitles **Predicting phishing websites using machine learning** is submitted to the department of of Computer Science and Engineering ,Delhi Technological University. The project is an authentic record of my own carried out from **August 2018 to October 2018** under supervision of **Ms Indu Singh, Assisstant professor CSE Department DTU.**

Siddhant Jain

(2K16/CO/314)

Sparsh Shubham Sinha
(2K16/CO/319)

Certificate

This is to certify that the minor project entitled **prediction of phishing websites using machine learning** carried out by Siddhant Jain (2K16/CO/314) and Sparsh Shubham Sinha (2K16/CO/319) in partial fulfillment for the requirements for the award of Bachelor of Technology in Computer Engineering at Delhi Technological University is an authentic work carried under my guidance and supervision .

To the best of my knowledge the matter has not been submitted to any other university for degree or diploma.

Ms. Indu Singh

(Assisstant Professor)

(Department of Computer Science and Engineering ,DTU)

Delhi-11004

Acknowledgement

“The successful completion of any task would be incomplete without accomplishing the people who made it all possible and whose constant guidance and encouragement secured us the success.”

First of all, we are grateful to the Almighty for establishing us to complete this project.

We are grateful to **Dr. Rajni Jindal, HOD** (Department of Computer Science and Engineering) Delhi Technological University (Formerly Delhi College of Engineering), New Delhi and all other faculty members of our department, for their astute guidance constant encouragement and sincere support for this project work.

We owe a debt of gratitude to our guide, **Ms. Indu Singh** (Assistant Professor COE Department) for incorporating in us the idea of a creative project, helping us in undertaking this project and also for being there whenever we needed her assistance.

I also place on record, my sense of gratitude to one and all, who directly or indirectly have lent their helping hand in this venture.

We feel proud and privileged in expressing our deep sense of gratitude to all those who have helped us in presenting this project.

Last but never the least; we thank our parents for always being with us, in every sense.

Abstract

Phishing is described as the art of emulating a website of a creditable firm intending to grab user’s private information such as usernames, passwords and social security number. Phishing websites comprise a variety of cues within its content-parts as well as browser-based security indicators. Several solutions have been proposed to tackle phishing. Nevertheless, there is no single magic bullet that can solve this threat radically. In our research we have used several Machine learning techniques such as logistic regression, Decesion tree, support vector machine and KNN to identify phishy websites based on a particular set of attributes and we have compared the performnace of these techniques in the required prediction.

Keywords- Phishing, Information Security, Support vector machines, Logistic regression , Decision tree, K nearest neighbours

List of Figures

List of Abbreviations Used

- SVM- Support vector machine
- DT- Decsion tree
- KNN- K nearest neighbours
- LR- Logistic regression
- TP- True positive
- TN- True negative
- FP- False positive
- FN- False negative

Introduction

Phishing is a fraudulent attempt, usually made through email, to steal your personal information. The best way to protect you from phishing is to learn how to recognize a phish. Phishing emails usually appear to come from a well-known organization and ask for your personal information such as credit card number, social security number, account number or password. Often times phishing attempts appear to come from sites, services and companies with which you do not even have an account. In order for Internet criminals to successfully "phish" your personal information, they must get you to go from an email to a website. Phishing emails will almost always tell you to click a link that takes you to a site where your personal information is requested. Legitimate organizations would never request this information of you via email.Cyber criminals are creating an average of around 1.4 million phishing websites every month with fake pages designed to lure the company they are spoofing and then replaced them within hours in order to ensure they're not detected. By building phishing websites with such short life-cycles, cyber criminals aim to make it hard for web crawlers to find their imposter pages, especially if there are no links to other sites. An analysis of phishing websites by researchers at webroot found that during the first half of 2017, an average of 1.4 million unique phishing websites were created every month, with the majority only online for between four and eight hours and most often pretending to be high profile technology and banking firms. According Webroot's statistics for the first half of 2017, Google was the most common company for attackers to impersonate, accounting for 35 percent of all phishing attempts. Chase, Dropbox, PayPal and Facebook made up the remaining five most popular disguises for phishing emails, while attackers also commonly claimed to be from Apple, Yahoo, Wells Fargo, Citi and Adobe.

Phishing attack classically starts by sending an email that appears to come from an enterprise to victims asking them to update or confirm their personal information by visiting a link within the email. Although, phishers are now using several techniques in creating phishing sites, they all use a set of mutual features to create phishing websites since without those features they lose the advantage of deception. This helps us to differentiate between honest and phishy websites based on the features extracted from the visited website. Overall, two approaches are used in identifying phishing sites. The first one is based on a blacklist, in which the requested URL is compared with those in that list. The downside of this approach is that the blacklist usually cannot cover all phishing websites since, within seconds, a new fraudulent website is launched. The second approach is known as heuristic-based methods, where several features are collected from the website to categorize it as either phishy or legitimate. In contrast to the blacklist method, a heuristic-based solution can recognize freshly

created phishing websites. The accuracy of the heuristic-based methods depends on picking a set of discriminative features that could help in distinguishing the type of website. The way in which the features processed also plays an extensive role in classifying.

Decision tree, support vector machine, KNN and logistic regression are principle classification algorithms that are used for identification of target parameter based on independent attributes. A tree can be *"learned"* by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called *recursive partitioning*. The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions. The construction of decision tree classifier does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle high dimensional data. In general decision tree classifier has good accuracy. Decision tree induction is a typical inductive approach to learn knowledge on classification. K-Nearest Neighbours is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection. It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible.In addition to performing linear classification, SVMs can efficiently perform a non-linear classification, implicitly mapping their inputs into high-dimensional feature spaces.

Problem Statement

Phishing websites are fake websites generated by dishonest people to impersonate honest websites. Users may be unable to access their emails or sometimes lose money because of phishing. Predicting and stopping this attack is a critical step toward protecting online trading. The accuracy of predicting the type of the website necessarily depends on the extracted features goodness. Since most users feel safe against phishing attacks if they utilize an anti-phishing tool, this throws a great responsibility on the anti-phishing tools to be accurate in predicting phishing. In that context, we believe that developing rules of thumb to extract features from websites then utilizing them to predict the type of websites is the key to success in this event. A report published by “Gartner”, which is a research and advisory company shows that phishing attacks continue to escalate. Gartner estimates that theft through phishing attacks costs U.S. banks and credit card issuers an estimated $2.8 billion annually. The Director of Cisco’s security-technology-business-unit said, “Personalized and targeted attacks that focus on gaining access to more lucrative corporate bank accounts and valuable intellectual property are on the rise”.

Motivation

Phishing websites are fake websites generated by dishonest people to impersonate honest websites. Users may be unable to access their emails or sometimes lose money because of phishing. Predicting and stopping this attack is a critical step toward protecting online trading. The accuracy of predicting the type of the website necessarily depends on the extracted features goodness. Since most users feel safe against phishing attacks if they utilize an anti-phishing tool, this throws a great responsibility on the anti-phishing tools to be accurate in predicting phishing. In that context, we believe that developing rules of thumb to extract features from websites then utilizing them to predict the type of websites is the key to success in this event. A report published by “Gartner”, which is a research and advisory company shows that phishing attacks continue to escalate. Gartner estimates that theft through phishing attacks costs U.S. banks and credit card issuers an estimated $2.8 billion annually. The Director of Cisco’s security-technology-business-unit said, “Personalized and targeted attacks that focus on gaining access to more lucrative corporate bank accounts and valuable intellectual property are on the rise”.

Objective

We hereby have carried out four machine learning techniques namely

- Logistic regression
- Decision tree
- Support vector machine
- K nearest neighbours

and we wish to carry out a comparison of all four mathods on a dataset that predicts the website as phishing / fraudulent or legitimate. Number of attributes have been taken into consideration for doing so and we wish to compare the methods based on their accuracy and precison by deviding the dataset into test set and training set in 80:20 ratio.

Literature Overview

Although quite a lot of of anti-phishing solutions are offered nowadays, most of them are not able to make a high accurate decision thus the false-positive decisions raised intensely. In this section, we review current anti-phishing methodologies and the features they employ in developing anti-phishing solutions.

One approach employed in , is based on experimentally contrasting association classification algorithms, i.e. Clasification Based Association (CBA) ,and Multi-class Classification based on Association classification with other traditional clasification algorithms (C4.5, PART,...etc.). The authors have gathered 27 different features from various websites and then categorize them into six criteria as shown in Table 1. To evaluate the selected features, the authors conducted experiments using the following data-mining techniques, MCAR , CBA , C4.5 , PRISM , PART and JRip. The results showed a significant relation between “Domain-Identity” and “URL” features. There was insignificant impact of the “Page Style” on “Social Human Factor” related features on the accuracy. Later in 2010 , the authors used the 27 features to build a model based on fuzzy-logic. Although, this is a promising solution it lacks to clarify how the features were extracted from the website, specifically features related to human-factors. Moreover, the rules were established based on human experience, which is one of the problems we aim to resolve in this article. Furthermore, the website was classified into five different classes i.e. (Very Legitimate, Legitimate, Suspicious, Phishy and Very Phishy), but the authors did not clarify what is the fine line that differentiates between these classes.

Another method proposed in , suggested a new way to detect phishing webbehaviorsapturing abnormal behave websitesmonstrated by these websites. The authors have selected six structural-features those are: Abnormal URL, Abnormal DNS record, Abnormal Anchors, Server-Form-Handler, Abnormal cookie, and Abnormal SSL-certificate. Once these features and the website identity are known, Support-Vector-Machine classifier “Vapnik’s” is used to determine whether the website is phishy or not. The classification accuracy of this method was 84%, which is relatively considered low. However, this method snubs important features that can play key role in determining the legitimacy of a website. This explains the low detection-rate.Nevertheless, this approach does not depend on any previous knowledge of the user or experience in computer

security. A method proposed in (16), suggested utilizing CANTINA (Carnegie Mellon Anti-phishing and Network Analysis Tool) which is content-based technique to detect phishing websites, using the term-frequency- inverse-document-frequency (TF-IDF) information-retrieval measures . TF-IDF produces weights thatvvassess the term importance to a document, by counting its frequency. CANTINA works as follow:

1. Calculate the TF-IDF for a given webpage.

2. Take the five highest TF-IDF terms and find the lexical-signature.

3. The lexical-signature is fed into a search engine.

If the N tops searching results having the current webpage, it is considered a legitimate webpage. Otherwise, it is a phishing webpage. N was set to 30 in the experiments. If the search engine returns zero result, thus the website is labelled as phishy, this point was the main disadvantage of using such technique since this would increase the false-positive rate. To overcome this weakness, the authors combined TF-IDF with some other features; those are Age of Domain, Known-Images, Suspicious-URL, IP-Address, Dots in URL, Forms. Another approach that utilizes CANTINA with an additional attribute and uses different machine-learning algorithms was proposed in (1). The authors have used 100 phishing websites and 100 legitimate ones in the experiments which is considered limited. The authors have performed three experiments; the first one evaluated a reduced CANTINA features set i.e. (dots in URL, IP-address, suspicious-URL and suspicious- link), and the second experiment involved testing whether the new feature i.e. (domain top-page similarity) is significant enough to play a key role in detecting website type. The third experiment evaluated the results after adding the new suggest feature to the reduced CANTINA features. The result showed that the new feature plays a key role in detecting the website type. The best accurate algorithm was Neural Network with error-rate equals to 7.50%, and Naïve Bayes (NB) gave the worst result with 22.5% error-rate.

In , the authors compared a number of learning methods including Support-Vector-Machine, rule-based techniques, decision-trees, and Bayephphishingechniques in detecting phishy emails. A random forest was implemented in PILFER (Phishing Identification by Learning on Features of Email Received). PILFER detected 96% of the phishing emails correctly with a false-positive rate of 0.1%. Ten email’s features displayed are used in the

experiments those are IP address URLs, Age of Domain, Non-matching URLs, “Here” Link, HTML emails, Number of Links, Number of Domains, Number of Dots, Containing Javascript, Spam-filter Output.

Our Proposed Approach

We will factor in the 32 attributes used to describe the address of a website to determine if the link leads to a phishing website.

List of attributes:

A. ADDRESS BAR BASED FEATURES

a. **IP Address**: If IP address is used as an alternative of a domain name in the URL e.g. 125.98.3.123 or it can be transformed to hexadecimal representation e.g. http://0x58.0xCC.0xCA.0x62, the user can almost be sure someone is trying to steal his personal information.

b. **Long URL**: Long URLs commonly used to hide the doubtful part in the address bar. Scientifically, there is no reliable length distinguishes phishing URLs from legitimate ones. As in (21), the proposed length of legitimate URLs is 75. However, the authors did not justify the reason behind their value. To ensure accuracy of our study, we calculated the length of URLs of the legitimate and phishing websites in our dataset and produced an average URL length. The results showed that if the length of the URL is less than or equal 54 characters then the URL classified as "Legitimate". On the other hand, if the URL length is greater than 74 characters then the website is "Phishy".

c. **Using @ Symbol**: Using "@" symbol in the URL leads the browser to ignore everything preceding the "@" symbol since the real address often follows the "@" symbol. . After reviewing our dataset, we were able to find 90 URL shaving "@" symbol, which constitute only 3.6%.

d. **Prefix or Suffix Separated by "-"** : Dashes are rarely used in legitimate domain-names. Phishers resort to add suffixes or prefixes separated by "-" to the domain names so that users feel they are dealing with a legitimate webpage. 661 URLs having "-"symbol were found in our dataset which constitutes 26.4%.

e. **Sub-Domain and Multi Sub-Domains:** Assume that we have the following linkhttp://www.hud.ac.uk/students/portal.com. A domain-name always includes the top-level domain (TLD), which in our example is "uk." The "ac" part is shorthand for academic, ".ac.uk" is called the second-level domain (SLD), and "hud" is the actual name of the domain. We note that the legitimate URL link has two dots in the URL since we can ignore typing "www.". If the number of dots is equal to three then the URL is classified as "Suspicious" since it has one sub-domain. However, if the dots are greater than three it is classified as "Phishy" since it will have multiple sub-domains. Our dataset contains 1109 URLs having three or more dots in domain part, which constitute 44.4%.

f. **HTTPS "Hyper Text Transfer Protocol with Secure Sockets Layer" and SSL "Secure Sockets Layer"**: Legitimate websites utilize secure domain-names every time sensitive information is transferred. The existence of https is important in giving the impression of website legitimacy, but it is not enough, since in 2005 more than 450 phishing URLs using https recognized by Netcraft Toolbar Community. Therefore, we further checked the certificate assigned with https including the extent of strust of certificate issuer unlike some previous researches, which consider fake https as a valid without

checking the certificate of the authority provider. Certificate authorities that are consistently listed among the top names for trustinclude GeoTrust, GoDaddy, Network Solutions, Thawte, and VeriSign. By reviewing our dataset, we find that the minimum certificate age for the URLs supporting HTTPs protocol was 2 years. In our dataset, we find 2321 URLs does not support https or use a fake https, which constitute 92.8%.

B. ABNORMAL BASED FEATURES

a. **Request URL**: For legitimate websites, most of the objects within the webpage are linked to the same domain .For example ,if the URL typed in the address bar was http://www.hud.ac.uk/students/portal.com we extract the keyword <src=> from the webpage source code and check whether the domain in the URL is different from that in <src>. If the result is true, the website is classified as "Phishy". To develop a rule for this feature, we calculated the ratio of URLs in source code that have different domain than the domain typed in the address bar. By reviewing our dataset, we find that the legitimate websites have in the worst case 22% of its objects loaded from different domains, whereas for phishing websites the ratio was in best case was 61%. Thus, we assumed that if the ratio is less than 22% then the website is considered "Legitimate" else if the ratio is between 22% and 61% then the website considered "Suspicious". Otherwise, the website is considered "Phishy". In this feature, we computed the feature existence rate not the number of feature existence; since the number of request URL in the website varies. The dataset contains 2500 URLs having this feature, which constitute 100 %.

b. **URL of Anchor:** An anchor is an element defined by the <a> tag. This feature is treated exactly as "Request URL". By reviewing our dataset, we find that the legitimate websites have in the worst case 31% of its anchor-tag connected to a different domain, whereas for phishing websites we find that the ratio was 67% in best case. Thus, we assumed that if the ratio is less than 31%then the website is considered "Legitimate" else if the ratio is between 31% and 67% then the website considered "Suspicious". Otherwise, the website is considered "Phishy". By reviewing our dataset, we find 581 URLs having this feature, which constitute 23.2%.

c. **Server Form Handler (SFH):** SFH that contains empty string or "about:blank" are considered doubtful since an action should be taken upon submitted information. In addition, if the domain-name in SFH-s is different from the domain of the webpage this gives an indication that the webpage is suspicious because the submitted information is rarely handled by external domains. In our dataset, we find 101 URLs having SFHs, which constitutes only 4.0%

d. **Abnormal URL:** This feature can be extracted from WHOIS database. For a legitimate website, identity is typically part of its URL. 412 URLs having this feature were founded in our dataset, which constitutes 16.4%.

C. HTML and Javascript Based Features

a. **Redirect Page**: Open redirects found on websites are liable to be exploited by phishers to create a link to their site. In our dataset, we find that the maximum number of redirect pages in the phishing websites was three, whereas this feature is rarely used in legitimate websites since we find only 21 legitimate

website having this feature and it is used for one time only in those websites. Thus if the redirection number is less than 2 then we will assign “Legitimate”, else if the redirection number is greater than or equal 2 and less than 4 then we will assign "Suspecious", otherwise we will assign “Phishy”. 249 URLs having redirect-page were encountered in our phishing dataset, which constitute 10%.

b. **Using onMouseOver to Hide the Link:** Phishers may use JavaScript to display a fake URL in the status bar to the users. To extract this feature we must explore the webpage source code particularly the “onmouseover” event and check if it make any changes on the status bar. 496 URLs having this feature were founded in our dataset, which constitutes 20%
c. **Disabling Right Click**: Phishers use JavaScript to disable the right-click function, so that users cannot view and save the source code. This feature is treated exactly as “Using onMouseOver to hide the Link”. However, for this feature, we will search for event “event. Button==2” in the source code and check if right click is disabled. We find this feature 40% times in our dataset, which constitutes1.6%.
d. **Using Pop-up Window:** It is unusual to find a legitimate website asking users to submit their credentials through a popup window. 227 URLs were founded in our dataset in which users credential submitted through a popup window, which constitutes 9.1%.

D. Domain Based Features

a. **Age of Domain:** This feature can be extracted from WHOIS database. In our dataset, we find that some domains host several phishy URL in several time slots. The blacklist may succeed in protecting the users if it works on the domain level not on the URL level i.e. add the domain-name to the blacklist not the URL address. However, find that 78% of phishing domains were in fact hacked domains, which already serve a legitimate website. Thus, blacklisting those domains will in-turn adds the legitimate websites to blacklist as well. Even though the phishing website has moved from the domain, legitimate websites may be left on blacklists for a long time; causing the reputation of the legitimate website or organization to be harmed. Some blacklists such as “Google’s Blacklist” need on average seven hours to be updated. By reviewing our dataset, we find that the minimum age of the legitimate domain was 6months. For this feature, if the domain created less than six months, it is classified as “Phishy”; otherwise, the website is considered “Legitimate”. In our dataset, 2392 URLs created less than 6 months, which constitute 95.6%.
b. **DNS Record:** For phishing sites, either the claimed identity in not recognized by the WHOIS database orfoundedcord of the hostname is not founded .If the DNS record is empty or not found then the website is classified as “Phishy”, otherwise it is classified as “Legitimate”. 160 URLs were found in our dataset where the DNS record is not found, and that constitute 6.4%.
c. **Website Traffic:** This feature measures the popularity of the website by determining the number of visitors and the number of pages they visit. However, since phishing websites lives for a short period-of-time thus they may not be recognized by the Alexa database. By reviewing our dataset, we find that in worst-case legitimate websites ranked among the top 150,000. Therefore, if the domain has no traffic or not being recognized by the Alexa

database it is classified as “Phishy” otherwise if the website ranked among the top 150,000 it is classified as “Legitimate” else it is classified as “Suspicious”. This feature constitutes 89.2% of the dataset since it appears 2231 times.

The dataset in first split into a training set and a testing set using the imported scklearn library .The following 4 algorithms will be implemented on the dataset:

1. Support Vector Machine
2. Decision Tree
3. Linear regression
4. K Nearest Neighbour

Support Vector Machine

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for either classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well (look at the below snapshot).

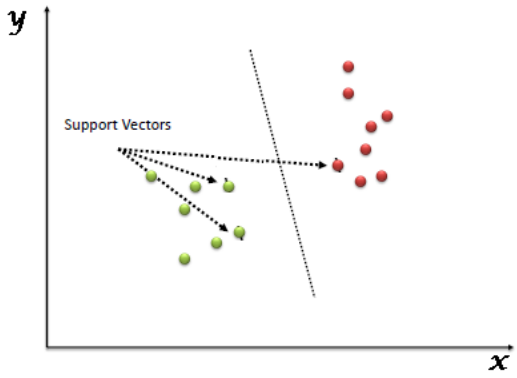

Fig 1

Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

Identify the right hyper-plane (Scenario-1): Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

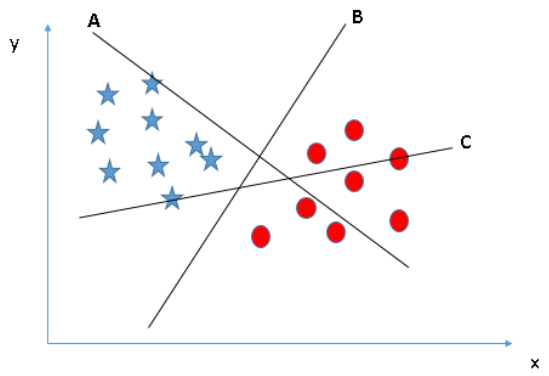

Fig 2

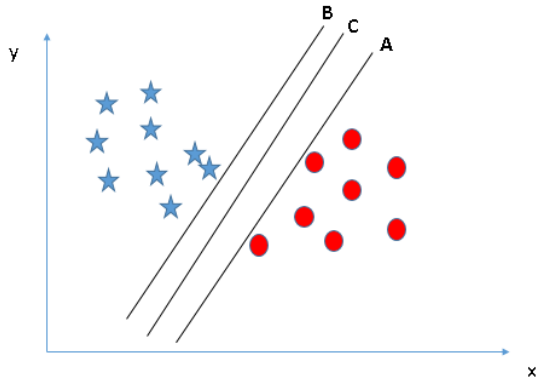

Fig 3

Identify the right hyper-plane (Scenario-2): Here, we have three hyper-planes (A, B and C) and all are segregating the classes well.

Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**.

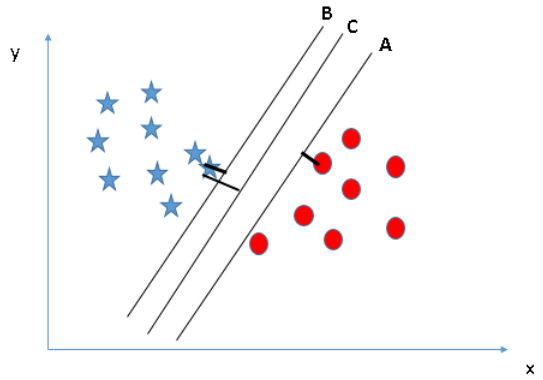

Fig 4

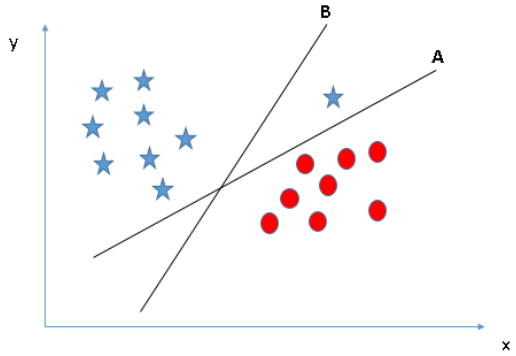

Fig 5

Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

Identify the right hyper-plane (Scenario-3): Use the rules as discussed in previous section to identify the right hyper-plane

In SVM, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, SVM has a technique called the kernel trick. These are functions which takes low dimensional input space and transform it to a higher dimensional space i.e. it converts not separable problem to separable problem, these functions are called kernels. It is

mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then find out the process to separate the data based on the labels or outputs you've defined.

Decision Tree

Decision tree is the most powerful and popular tool for classification and prediction. A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (terminal node) holds a class label. A tree can be *"learned"* by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called *recursive partitioning*. The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions. The construction of decision tree classifier does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle high dimensional data. In general decision tree classifier has good accuracy. Decision tree induction is a typical inductive approach to learn knowledge on classification.

Linear Regression

The representation is a linear equation that combines a specific set of input values (x) the solution to which is the predicted output for that set of input values (y). As such, both the input values (x) and the output value are numeric.

The linear equation assigns one scale factor to each input value or column, called a coefficient and represented by the capital Greek letter Beta (B). One additional coefficient is also added, giving the line an additional degree of freedom (e.g. moving up and down on a two-dimensional plot) and is often called the intercept or the bias coefficient.

For example, in a simple regression problem (a single x and a single y), the form of the model would be:

$y = B0 + B1*x$

In higher dimensions when we have more than one input (x), the line is called a plane or a hyper-plane. The representation therefore is the form of the equation and the specific values used for the coefficients (e.g. B0 and B1 in the above example).

It is common to talk about the complexity of a regression model like linear regression. This refers to the number of coefficients used in the model.

When a coefficient becomes zero, it effectively removes the influence of the input variable on the model and therefore from the prediction made from the model ($0 * x = 0$). This becomes relevant if you look at regularization methods that change the learning algorithm to reduce the complexity of regression models by putting pressure on the absolute size of the coefficients, driving some to zero.

K Nearest Neighbour

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

1. Load the data
2. Initialise the value of k
3. For getting the predicted class, iterate from 1 to total number of training data points
    1. Calculate the distance between test data and each row of training data. Here we will use Euclidean distance as our distance metric since it’s the most popular method. The other metrics that can be used are Chebyshev, cosine, etc.
    2. Sort the calculated distances in ascending order based on distance values
    3. Get top k rows from the sorted array
    4. Get the most frequent class of these rows
    5. Return the predicted class

RESULTS:

SNO	METHOD USED	ACCURACY	PRECISION
1	Decision Tree	96.59862	97.567890
2	K Nearest Neighbours	94.983456	94.67834
3	Logistic Regression	91.768903	91.234545
4	Support Vector Machine	91.678345	91.134567

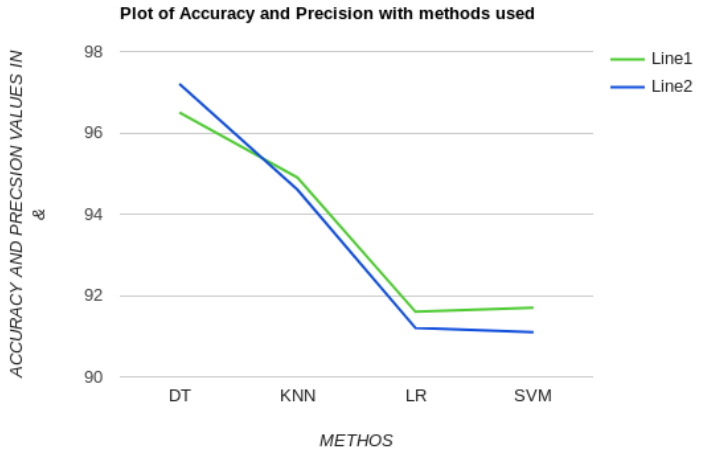

Fig 6

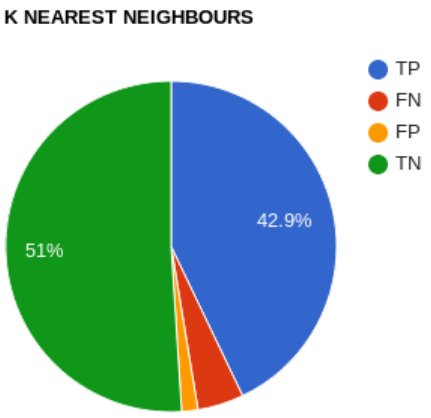

Fig 7

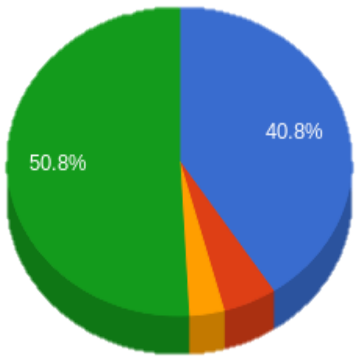

Fig 8

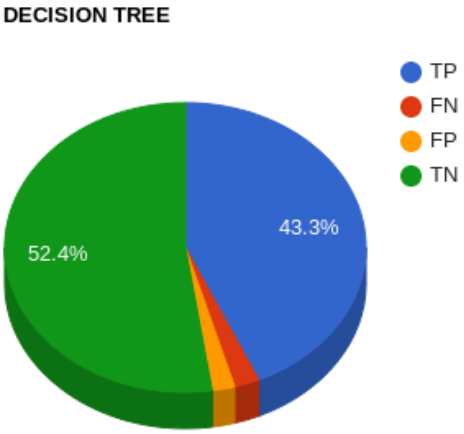

Fig 9

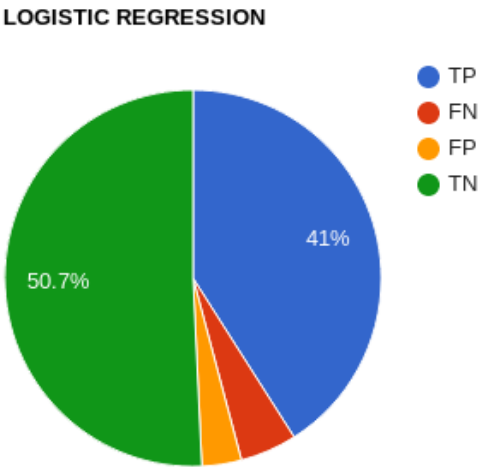

Fig 10

The following graph clearly reveals that Cart Decison tree analysis method used on the guven dataset gives best values for both accuracy and prescion and outshines other three methods of analysis ie. Logistic regression , K nearest neighbours and support vector machines. Decision tree developed using CART method gives an accuracy of around 96% and precison of 97% as compared to support vector machines that gives just 91%.

REFERENCES

1.

Kaspersky Lab, “Spam in January 2012: Love, Politics and Sport,” 2013. [Online]. Available: http://www.kaspersky.com/about/news/spam/2012/Spam_in_January_2012_Love_Politics_and_Sport. [Accessed 11 February 2013].

2.

seogod, “Black Hat SEO,” SEO Tools, 16 June 2011. [Online]. Available: http://www.seobesttools.com/black-hat-seo/. [Accessed 8 Januery 2013].

3.

R. Dhamija, J. D. Tygar and M. Hearst, “Why Phishing Works.,” in Proceedings of the SIGCHI conference on Human Factors in computing systems, Cosmopolitan Montréal, Canada, 2006.

4.

L. F. Cranor, “A framework for reasoning about the human in the loop,” in UPSEC'08 Proceedings of the 1st Conference on Usability, Psychology, and Security, Berkeley, CA, USA, 2008.

5.

D. Miyamoto, H. Hazeyama and Y. Kadobayashi, “An Evaluation of Machine Learning-based Methods for Detection of Phishing Sites,” Australian Journal of Intelligent Information Processing Systems, pp. 54-63, 2 10 2008.

6.

X. Guang, o. Jason, R. Carolyn P and C. Lorrie, “CANTINA+: A Feature-rich Machine Learning Framework for Detecting Phishing Web Sites,” ACM Transactions on Information and System Security, pp. 1-28, 09 2011.

7.

I. H. Witten and E. Frank, “Data mining: practical machine learning tools and techniques with Java implementations,” ACM, New York, NY, USA, March 2002.

8.

Y. Zhang, J. Hong and L. Cranor, “CANTINA: A Content-Based Approach to Detect Phishing Web Sites,” in Proceedings of the 16th World Wide Web Conference, Banff, Alberta, Canada, 2007.

9.

B. Widrow, M. and A. Lehr, “30 years of adaptive neural networks,” IEEE press, vol. 78, no. 6, pp. 1415-1442, 1990.

10.

I. Basheer and M. Hajmeer, “Artificial neural networks: fundamentals, computing, design, and application.,” Journal of Microbiological Methods., vol. 43, no. 1, pp. 3-31, 2000.