Diagnosis of Oncological Diseases based on Algorithm of Supervised Learning

Kanika Negi[1], Aakash Uppadhaya[2]

[1,2]Research Scholar

Department of Information Technology, Thakur college of Engineering and Technology

Abstract-The objective of this paper is particularly based on prediction of cancer. As cancer is being identified as disease that has different types. The early stage detection is proved to be helpful for the patient. Machine learning has played a significant role in bioinformatics science. There are various techniques in machine learning that can help to predict cancer based on various feature of tumor. Such as radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, fractal dimension and many more. We are focusing on supervised learning that can help to predict cancer. We can build various predictive models based on this technique that helps in appropriate decision making.

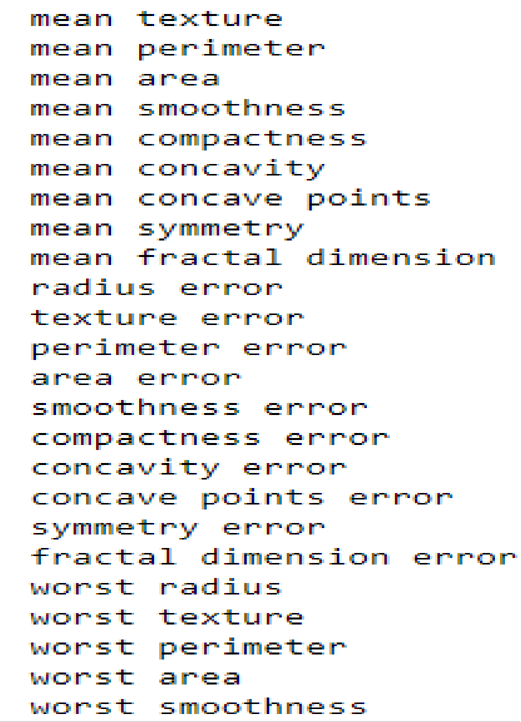
Keywords-cancer prediction, machine learning, supervised learning.

I. INTRODUCTION

From last few decades, there is an exponential upgradation of cancer research. Scientists have applied various methodology to detect cancer before they show symptoms, With the growth of new technology and modern digitalization era hence results of large amount of cancer data has been gathered and are available for medical research. The appropriate prediction is the most difficult task for the medical researchers. Machine learning is a field of computer science that enables computer to learn from data. These techniques can discover and identify patterns and relationship between them for complex dataset which they can effectively predict outcomes of a tumor type that is whether it is malignant or benign. Machine learning is sub-divided into various learning methodology such as supervised learning, unsupervised learning, reinforcement learning. supervised learning consist of regression, classification and unsupervised learning consist of cluster analysis and anomaly detection. We can implement this technique by various algorithm. In this paper we are using k-nearest neighbor for classification of cancer tumor.

II. SYSTEM MODEL

k-Nearest Neighbors (KNN) is a memory-based model defined by a set of objects known as examples (also known as instances) for which the outcome is known (i.e., the examples are labeled). Each example consists of a data case having a set of independent values labeled by a set of dependent outcomes. The independent and dependent variables can be either continuous or categorical. For continuous dependent variables, the task is regression; otherwise it is a classification. Thus, STATISTICA KNN can handle both regression and classification tasks. Here in our case we can use various features of tumor such as follow:



```
mean texture
mean perimeter
mean area
mean smoothness
mean compactness
mean concavity
mean concave points
mean symmetry
mean fractal dimension
radius error
texture error
perimeter error
area error
smoothness error
compactness error
concavity error
concave points error
symmetry error
fractal dimension error
worst radius
worst texture
worst perimeter
worst area
worst smoothness
```

Fig.1. Features.

These features act as an independent variable in our model and a feature called as target will act as dependent variable that is, if the target is set to 1 then the tumor is a malignant tumor and if target is set to 0 then tumor is benign tumor.

Given a new case of dependent values (query point), we would like to estimate the outcome based on the KNN examples. STATISTICA KNN achieves this by finding K examples that are closest in distance to the query point, hence, the name k-Nearest Neighbors. For regression problems, KNN predictions are based on averaging the outcomes of the K-nearest neighbors; for classification problems, a majority of voting is used.our problem set is based on classification problem that is 1 for malignant tumor and 0 for benign tumor.

Cross-Validation of the Model

Cross-validation is a well-established technique that can be used to obtain estimates of model parameters that are unknown. Here we discuss the applicability of this technique to estimate k.

The general idea of this method is to divide the data sample into a number of v folds (randomly drawn, disjointed sub-samples or segments). For a fixed value of k, we apply the KNN model to make predictions on the vth segment (i.e., use the v-1 segments as the examples) and evaluate the error. The most common choice for classification is most conveniently defined as the accuracy (the percentage of correctly classified cases). This process is then successively applied to all possible choices of v. At the end of the v folds (cycles), the computed errors are averaged to yield a measure of the stability of the model (how well the model predicts query points). The above steps are then repeated for various k and the value achieving the lowest error (or the highest classification accuracy) is then selected as the optimal value for k (optimal in a cross-validation sense). Note that cross-validation is computationally expensive and you should be prepared to let the algorithm run for some time especially when the size of the examples sample is large. Alternatively, you can specify k. This may be a reasonable course of action should you have an idea of which value k may take (i.e., from previous KNN analyses you may have conducted on similar data) .

Distance Metric

As mentioned before, given a query point, KNN makes predictions based on the outcome of the K neighbors closest to that point. Therefore, to make predictions with KNN, we need to define a metric for measuring the distance between the query point and cases from the examples sample. One of the most popular choices to measure this distance is known as Euclidean. Other measures include Euclidean squared, City-block, and Chebyshev:

$$D(x,p) = \left\{ \begin{array}{ll} \sqrt{(x-p)^2} & \text{Euclidean} \\ (x-p)^2 & \text{Euclidean squared} \\ \text{Abs}(x-p) & \text{Cityblock} \\ \text{Max}(|x-p|) & \text{Chebyshev} \end{array} \right\}$$

eve

where x and p are the query point and a case from the examples sample, respectively.

One of the other method is Logistic regression 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.

III. PREVIOUS WORK

It was in 1940s when the first manually operated computer system, ENIAC, was invented. At that time the word "computer" was being used as a name for a human with intensive numerical computation capabilities, so, ENIAC was called a numerical computing machine. In the 1950s, we see the first computer game program claiming to be able to beat the checkers world champion. This program helped checkers players a lot in improving their skills! Around the same time, Frank Rosenblatt invented the Perceptron which was a very, very simple classifier but when it was combined in large numbers, in a network. Then we see several years of stagnation of the neural network field due to its difficulties in solving certain problems. Thanks to statistics, machine learning became very famous in 1990s. The intersection of computer science and statistics gave birth to probabilistic approaches in AI. This shifted the field further toward data-driven approaches. Having large-scale data available, scientists started to build intelligent systems that were able to analyze and learn from large amounts of data. As a highlight, IBM's Deep Blue system beat the world champion of chess, the grand master Garry Kasparov.

IV. PROPOSED METHODOLOGY

In supervised learning, each example is a pair consisting of an input object (typically a vector) and the desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can used for mapping new examples.

Basic flow of supervised learning:

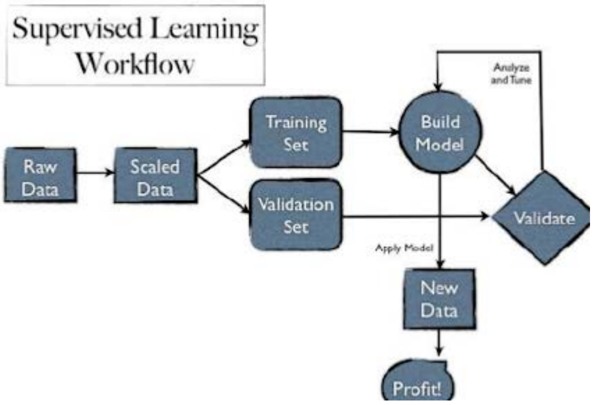


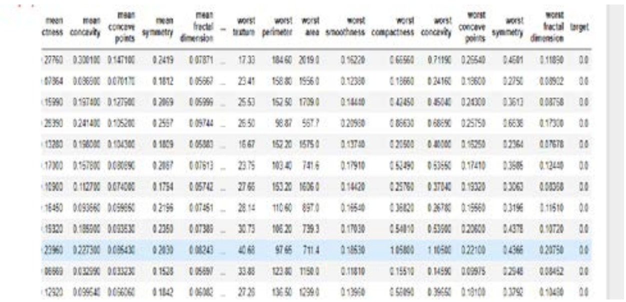
Fig.2. workflow

V. SIMULATION/EXPERIMENTAL RESULTS

k-Nearest Neighbor Predictions

In classification problems, KNN predictions are based on voting scheme in which the winner is used to label the query

Let's consider following dataset



mean ctness	mean concavity	mean concave points	mean symmetry	mean fractal dimension	...	worst texture	worst perimeter	worst area	worst smoothness	worst compactness	worst concavity	worst concave points	worst symmetry	worst fractal dimension	target
27760	0.300100	0.147100	0.2419	0.07871	...	17.33	184.60	2019.0	0.16220	0.66560	0.71190	0.26540	0.4601	0.11890	0.0
07864	0.086900	0.070170	0.1812	0.05667	...	23.41	158.80	1956.0	0.12380	0.18660	0.24160	0.18600	0.2750	0.08902	0.0
15990	0.197400	0.127900	0.2069	0.05999	...	25.53	152.50	1709.0	0.14440	0.42450	0.45040	0.24300	0.3613	0.08758	0.0
28390	0.241400	0.105200	0.2597	0.09744	...	26.50	98.87	567.7	0.20980	0.86630	0.68690	0.25750	0.6638	0.17300	0.0
13280	0.198000	0.104300	0.1809	0.05883	...	16.67	152.20	1575.0	0.13740	0.20500	0.40000	0.16250	0.2364	0.07678	0.0
17000	0.157800	0.080890	0.2087	0.07613	...	23.75	103.40	741.6	0.17910	0.52490	0.53550	0.17410	0.3985	0.12440	0.0
10900	0.112700	0.074000	0.1794	0.05742	...	27.66	153.20	1606.0	0.14420	0.25760	0.37840	0.19320	0.3063	0.08368	0.0
16450	0.093660	0.059850	0.2196	0.07451	...	28.14	110.60	897.0	0.16540	0.36820	0.26780	0.15560	0.3196	0.11510	0.0
19320	0.185900	0.093530	0.2350	0.07389	...	30.73	106.20	739.3	0.17030	0.54010	0.53900	0.20600	0.4378	0.10720	0.0
23960	0.227300	0.085430	0.2030	0.08243	...	40.68	97.65	711.4	0.18530	1.05800	1.10500	0.22100	0.4366	0.20750	0.0
06669	0.032990	0.033230	0.1528	0.05697	...	33.88	123.80	1150.0	0.11810	0.15510	0.14590	0.09975	0.2948	0.08452	0.0
12920	0.099540	0.066060	0.1842	0.06082	...	27.28	136.50	1299.0	0.13960	0.56000	0.39650	0.18100	0.3792	0.10430	0.0

Fig.3. Dataset

There are also various other feature as mentions above that has taken into consideration.

We have used 30 feature that is independent variable and 1 feature as dependent with k=1 and distance metric that is used is minkowski distance the equation is as follows

$$d^{MKD}(i,j) = \sqrt[\lambda]{\sum_{k=0}^{n-1} \left| y_{i,k} - y_{j,k} \right|^{\lambda}}$$

In the equation dMKD is the Minkowski distance between the data record i and j, k the index of a variable, n the total number of variables y and λ the order of the Minkowski metric

We get accuracy of 0.91 with k=1 and accuracy of 0.93 with k=5 and keeping distance metric.

Table-1: Result-1

algorithm	**Type**	**k-factor**	**features**	**accuracy**
KNN	classification	1	30	0.91
KNN	classification	5	30	0.93

Pros:

- Simple to implement
- Flexible to feature / distance choices
- Naturally handles multi-class cases
- Can do well in practice with enough representative data

Cons:

- Large search problem to find nearest neighbors
- Storage of data
- Must know we have a meaningful distance function

Application

KNN has various application in bioinformatics science we can use them to predict DNA sequencing. It helps to predict various focus areas of cancer such ascancer susceptibility, recurrence and survival Etc.

Models to classify cancerous tumor using Logistic Regression

Logistic Regression is used in various biological application. Logistic Regression is performed when the dependent variable(target) is categorical. Here in our case Whether the tumor is malignant (1) or not (0)

Model

Output = 0 or 1

Hypothesis => Z = WX + B

hΘ(x) = sigmoid (Z)

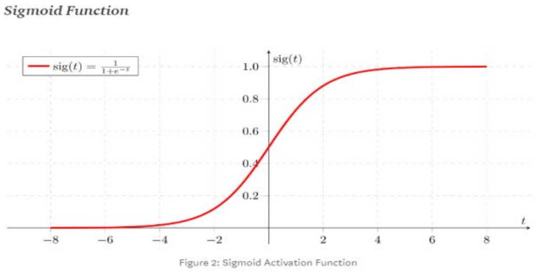


Fig.4 Sigmoid Representation.

Table-2: Result-2

Algorithm	Type	feature	accuracy
Logistic Regression	Classification	30	0.95

VI. CONCLUSION

we have detected whether particular tumor is cancerous or non- cancerous by means of supervised learning algorithm that is k-nearest neighbor and Logistic Regression. We have seen different k-parameter and the change in behavior of accuracy of algorithm while changing the k-factor. No algorithm is in general 'better' than another. It basically that any two optimization algorithms are equivalent when their performance is averaged across all possible problems. But of course in reality, you do not want to solve all possible problems but some particular practical one

VII. FUTURE SCOPES

Deep learning (also known as deep structured learning or hierarchical learning) is part of a broader family of machine learning methods based on learning data representations, as opposed to task-specific algorithms. Learning supervised, semi-supervised or unsupervised. It consists of majorly 3 parts such as CNN, RNN and standard neural network. IN which CNN is majorly used to predict disease from X-ray, MRI, C-T Scan.

REFERENCES

[1] Python Machine Learning: Introduction To Machine Learning With Python by Frank Millstein

[2] blog]A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm by Saravanan Thirumuruganathan

[3] Applied logistic regression 2nd edition by David W.Hosmer