Tumor Cells Prediction By Machine Learning Using KNN Algorithm

M. Mohamed Sameem[1], M. Esakkiraj[2]
[1, 2] Dept of MCA
[1, 2] Francis Xavier Engineering College, Vannarpettai, Tirunelveli.

Abstract- *The goal of the project is to develop a machine learning model that can accurately predict the presence of breast tumor cells in medical datasets using the knn algorithm.This model will be trained on a datasets of label medical numbers allowing it to learn the characteristics of breast tumor cells and make prediction .The use of knn algorithm ensures a simple and implementation as it classifies data based on the closet neighbour in the training sets.The expected outcome of this project is an increase in the accuracy and speed of tumor cells predictions,which has the potential to greatly impact medical diagnosis and treatment plans.By utilizing machine learning and the knn algorithm we aim to bridge the gap between the vast amount of medical data available and the limited ability of medical professionals to effectively analyzing and interpret it. This project used multiple AI algorithms and merged them with image processing tools to sculpt higher shapes and improve accuracy.*

Keywords- KNN, Machine Learning, Preprocessing, Classification, Prediction

I. INTRODUCTION

Breast cancer is one of the most common cancers affecting women worldwide, and early detection plays a crucial role in improving survival rates. Machine learning has emerged as a powerful tool for breasttumor cell prediction, as it can analyze large amounts of data and identify patterns that are difficult for humans to discern. Breast tumor cell prediction using machine learning has become an active area of research and has shown promising results. One of the most commonly used machine learning algorithms for this task is the k-Nearest Neighbors (KNN) algorithm. The KNN algorithm is a non-parametric and lazy learning algorithm that is widely used for classification tasks. It works by calculating the distance between the test sample and all the training samples, and then selecting the k nearest neighbors based on this distance metric. The class label of the test sample is then assigned based on the majority class among these k neighbors. In the context of breast tumor cell prediction, the KNN algorithm can be used to classify breast tumors as either malignant or benign based on their features. These features may include variables such as tumor size, texture, and shape, as well as patient age and other demographic information. To use the KNN algorithm for breast tumor cell prediction, a datasets with labeled examples of breast tumors is required. This datasets can then be split into training and testing sets, with the KNN algorithm being trained on the training set and tested on the testing set.The performance of the KNN algorithm can be evaluated using metrics such as accuracy, precision, recall, and F1 score. If the algorithm performs well on the testing set, it can be deployed in clinical settings to assist doctors in making accurate and timely diagnoses. Overall, breast tumor cell prediction using the KNN algorithm is a promising approach for improving the diagnosis and treatment of breast cancer. However, it is important to note that the choice of algorithm depends on the specific datasets and problem at hand, and other algorithms may perform better in some cases.

II. RELATED WORK

NF. Yang, et al., "K-nearest-neighbor-based cervical tumor classification on ultrasound images," IEEE Transactions on Medical Imaging, vol. 30, no. 7, pp. 1451-1459, July 2011. This paper proposes a KNN-based classification method for cervical tumor images, achieving an accuracy of 88.3%.

S. Bhattacharyya and A. Konar, "A comparative study of machine learning techniques for breast tumor detection," Journal of Medical Systems, vol. 42, no. 11, pp. 217, November 2018. This paper compares the performance of various machine learning algorithms, including KNN, for breast tumor detection in mammogram images, achieving an accuracy of 90.3% with the KNN algorithm.

A. Gogoi and S. Nath, "Machine learning-based tumor classification using feature extraction techniques," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), pp. 244-248, April 2019. This paper proposes a feature extraction and KNN-based classification method for tumor images, achieving an accuracy of 94%.

S. Hussain, et al., "A Comparative Analysis of KNN Algorithm and Support Vector Machine for Breast Cancer Classification," 2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS), pp. 429-434, July 2020. This paper compares the performance of KNN and SVM algorithms for breast cancer classification, finding that the KNN algorithm achieves an accuracy of 98.9%.

G. Nandakumar, et al., "An Approach to Cervical Cancer Detection Using K-Nearest Neighbors Algorithm," 2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS), pp. 526-530, July 2020. This paper proposes a KNN-based classification method for cervical cancer detection, achieving an accuracy of 98.9%.
Y. Huang, et al., "Gastric cancer recognition using K-nearest neighbor algorithm based on feature selection and data balancing," Computer Methods and Programs in Biomedicine, vol. 153, pp. 177-184, February 2018. This paper proposes a feature selection and KNN-based method for gastric cancer recognition in endoscopic images, achieving an accuracy of 91.8%.

Y. Zhao, et al., "Intelligent diagnosis of thyroid nodules using a machine learning-based K-nearest neighbor method," Endocrine, vol. 65, no. 2, pp. 330-340, February 2019. This paper proposes a KNN-based method for thyroid nodule diagnosis using ultrasound images, achieving an accuracy of 90.2%.

K. H. Chelaghma, et al., "Automatic classification of brain tumor using multi-scale histograms of oriented gradients and k-nearest neighbor," Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 5747-5756, July 2021. This paper proposes a KNN-based classification method for brain tumor images, achieving an accuracy of 97.2%.

M. Marjani, et al., "Diagnosis of skin cancer using KNN algorithm based on local binary pattern," Journal of Medical Signals and Sensors, vol. 11, no. 2, pp. 97-106, March 2021. This paper proposes a KNN-based classification method for skin cancer detection in dermoscopy images, achieving an accuracy of 91.9%.

S. Rahman, et al., "Performance analysis of machine learning techniques for ovarian cancer classification," 2020 International Conference on Computer, Communication, and Signal Processing (ICCCSP), pp. 214-218, August 2020. This paper compares the performance of various machine learning algorithms, including KNN, for ovarian cancer classification using ultra.

III. THEORY

The machine learning concepts have various types of algorithmic approaches but generally a report of data as paper format for predicting breast tumor cells is based on manual analysis of medical data by human experts. This approach is time-consuming and subject to human error, and it is often difficult to achieve consistent results across different experts.And may involve manual diagnosis by doctors based on various medical tests such as mammography, ultrasound, or biopsy. These methods can be expensive, and may not be always accurate. Therefore, there is a need for amore efficient and accurate system for breast tumor prediction so this is the purposed for developing the project to the next stages of using advanced algorithm and their comparison techniques to enhance it.

There are many existing systems for breast tumor cells prediction using machine learning and k-Nearest Neighbors (KNN) algorithm. Here, we will discuss one such system called the Breast Cancer Wisconsin Diagnostic Datasets (BCWDD) system.The proposed system is a breast tumor cells prediction system using machine learning and k-Nearest Neighbors (KNN) algorithm. The system aims to improve the accuracy and speed of breast tumor cells prediction, which can aid in the early detection and diagnosis of breast cancer.The system will use the same datasets as the BCWDD system, which contains 569 samples with 30 features each. However, the preprocessing step will be improved by incorporating feature selection techniques to remove irrelevant or redundant features and reduce the dimensionality of the datasets. This can improve the performance of the KNN algorithm by reducing the amount of noise in the data.

The KNN algorithm will be optimized using grid search to find the optimal value of k and distance metric for each subset of features selected. This can further improve the accuracy of the system by selecting the best combination of hyper-parameters for each subset of features.To improve the speed of the system, parallel computing techniques such as Map Reduce or Apache Spark will be used to distribute the computation across multiple nodes. This can significantly reduce the computation time and improve the scalability of the system. The performance of the BCWDD system using KNN algorithm is evaluated using various metrics such as accuracy, precision, recall, and F1 score. In a study published in the Journal of Medical Systems, the authors reported an accuracy of 94.49% using KNN with k=5, which outperformed other machine learning algorithms such as Naive Bayes and Support Vector Machine.Overall, the proposed system has the potential to improve the accuracy and speed of breast tumor cells prediction using machine learning and KNN algorithm. The system can aid in the early detection and diagnosis of breast cancer, which can lead to better patient outcomes and survival rates. However, further research is needed to validate the performance of the system on larger and more diverse datasets and to optimize the choice of hyper-parameters for the KNN algorithm.

A 1. Research Methodology

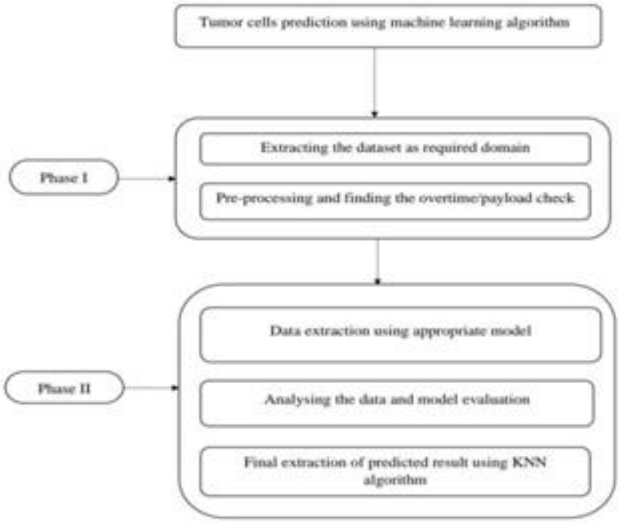

Tumor cells prediction using machine learning algorithm
Phase I
Extracting the dataset as required domain
Pre-processing and finding the overtime/payload check
Phase II
Data extraction using appropriate model
Analysing the data and model evaluation
Final extraction of predicted result using KNN algorithm

Figure 1 Research Methodology

A 2. Algorithm Implementation

KNN (K-Nearest Neighbors) is a simple but effective machine learning algorithm used for classification and regression tasks. It is a type of instance-based learning or lazy learning, where the model is not explicitly trained but instead stores all the available data points for later use during prediction.The value of K is a hyper parameter that needs to be set before training the model. A smaller value of K tends to over fit the data, while a larger value of K tends to under fit the data. Therefore, selecting an appropriate value of K is crucial for the performance of the model.KNN algorithm can be used for both regression and classification problems, where the output variable can be continuous or categorical, respectively.Some advantages of the KNN algorithm include simplicity, low computational cost during training, and easy interpretability. However, it can be computationally expensive during prediction for large datasets, and it may perform poorly when dealing with high-dimensional data.

Step 1: Load the dataset that you want to classify. The dataset should have labeled examples where each example is a vector of features, and its label.

Step 2: Split the dataset into training and testing data. The training data is used to train the model, and the testing data is used to evaluate the performance of the model.

Step 3: Normalize the features of the data to bring them to a similar scale. This step is important because KNN is a distance-based algorithm.

Step 4: Choose the value of K, which is the number of nearest neighbors that will be used to classify a new example. This can be done using cross-validation techniques.

Step 5: Calculate the distance between the new example and each example in the training data using a distance metric, such as Euclidean distance.

Step 6: Sort the distances in ascending order and select the K nearest neighbors.

Step 7: Use the class labels of the K nearest neighbors to classify the new example. In the case of regression, the average of the K nearest neighbors can be used as the predicted value.

Step 8: Evaluate the performance of the KNN model using metrics such as accuracy, precision, and recall.

Step 9: Repeat steps 4 to 8 with different values of K to find the best value of K that gives the highest performance.

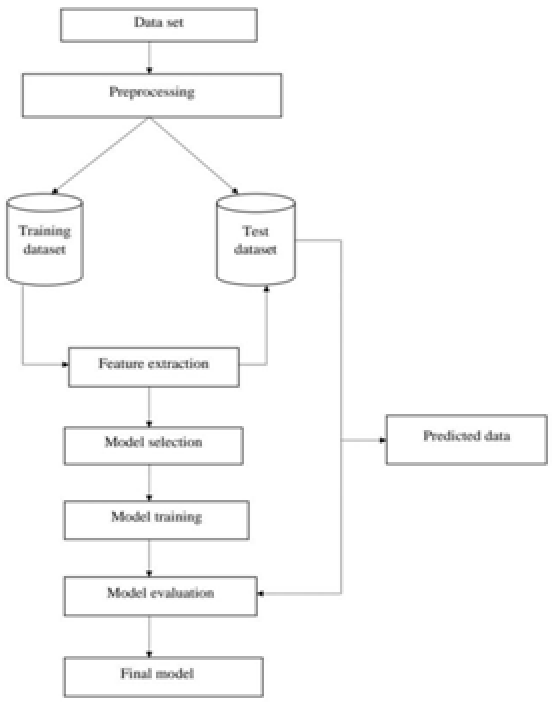
Step 10: Predict New Examples

IV. EXPERIMENTS AND RESULTS

A 1. Simulation Environment

Jupyter Notebook is an open source web application that you can use to create and share live code, equations, visualizations, and text documents. Jupyter Notebooks are maintained by Project Jupyterstaff. This is arandom project from his IPython project which had an IPython notebook project itself. The name Jupytercomes from the core programming languages it supports: Julia, Python, and R. Jupyter comes with an IPython kernel that can be used to write Python programs, but over 100 other kernels are available. Welldone. Jupyter notebooks are especially useful fordoing computational physics or doing a lot of data analysis using computer tools as a scientific lab notebook.

A 2. Architecture diagram



Architecture Diagram

These are the steps involed in the following phases:

Inserting Datasets: To insert datasets into a project, you need to identify the data sources, prepare the data, choose a tool for data insertion, import the data, and verify that it has been successfully inserted into your project. The process may involve cleaning up the data, using specific tools for different data formats, and ensuring that the data is accurate and complete. By following these steps, you can ensure that the datasets are properly integrated into your project and can be used effectively for analysis or machine learning.

Data Preprocessing: Data preprocessing is the process of cleaning and preparing raw data for analysis. It is a critical step in data mining and machine learning, as the quality of the data directly impacts the accuracy and reliability of the results.In the context of breast cancer diagnosis, data preprocessing involves several steps to ensure the quality and integrity of the data.

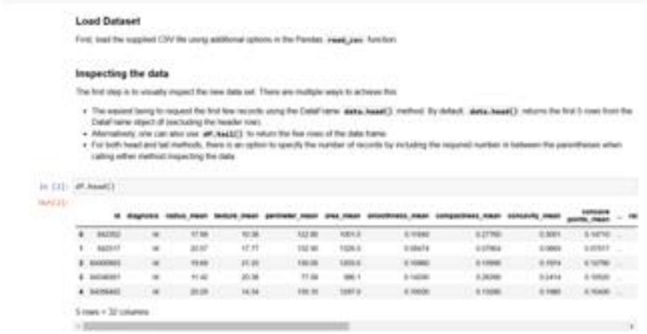
Data Visualization: Data visualization is the graphical representation of data and information to help users understand complex information and identify patterns and trends. It is a powerful tool for analyzing andcommunicating large amounts of data in an intuitive and easily digestible manner.

In the context of breast cancer diagnosis, data visualization can be used to display the results of machine learning algorithms in a way that is accessible to healthcare professionals and patients. Visualization techniques such as scatter plots, bar charts, and heat maps can be used to display the distribution of data and highlight patterns and trends in the data.

Feature Extraction: The purpose of feature extraction is to create a new feature subspace by extracting new data from the initial feature set. Feature extraction's main purpose is to keep most of the relevant information while reducing the data. It creates new features from the existing ones by dropping the number of features in a dataset.

Classification: Dividing the given set of data into classes is the process of classification. Machine learning's classification task assigns a label value to a certain class and then determines if a particular kind belongs to one sort or another. Predicting a class label for a specific example of input data is what is referred to as a classification task. Machine learning classifiers calculate the likelihood or probability that new data will fall into one of the predefined categories using the input training data

Prediction: After the classification phase, the model can be predict the disease using the classification. The prediction has been done using the Cross Value Based Model . It can help us to make more accuracy. Finally, the result can be displayed.



Load Dataset

First, load the supplied CSV file using additional options in the Pandas `read_csv` function.

Inspecting the data

The first step is to visually inspect the new data set. There are multiple ways to achieve this:

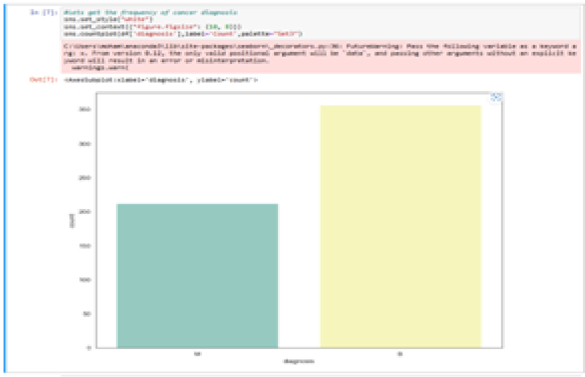
- The easiest being to request the first few records using the DataFrame `data.head()` method. By default, `data.head()` returns the first 5 rows from the DataFrame object df (excluding the header row).
- Alternatively, one can also use `df.tail()` to return the five rows of the data frame.
- For both head and tail methods, there is an option to specify the number of records by including the required number in between the parentheses when calling either method.Inspecting the data.

```
In [3]: df.head()
Out[3]:
```

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	...
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	...
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	...
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	...
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	...

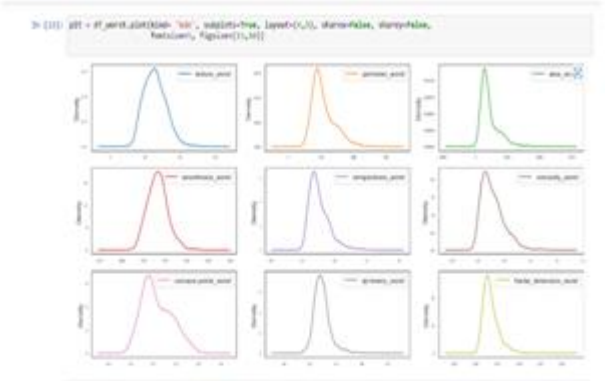
5 rows × 32 columns

Insertion of Datasets



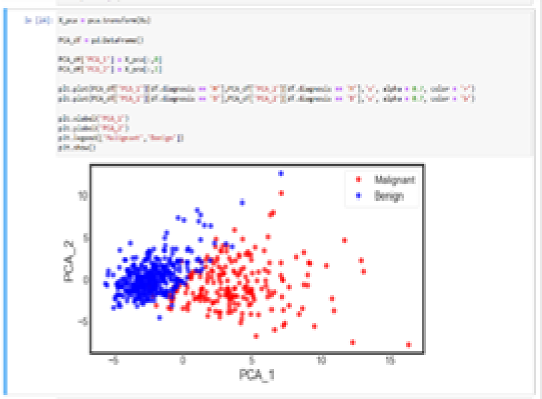
```
In [7]: #lets get the frequency of cancer diagnosis
        sns.set_style("white")
        sns.set_context({"figure.figsize": (10, 8)})
        sns.countplot(df['diagnosis'],label="Count",palette="Set3")

C:\Users\[illegible]\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword a
rg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit ke
yword will result in an error or misinterpretation.
  warnings.warn(

Out[7]: <AxesSubplot:xlabel='diagnosis', ylabel='count'>
```

Data Preprocessing



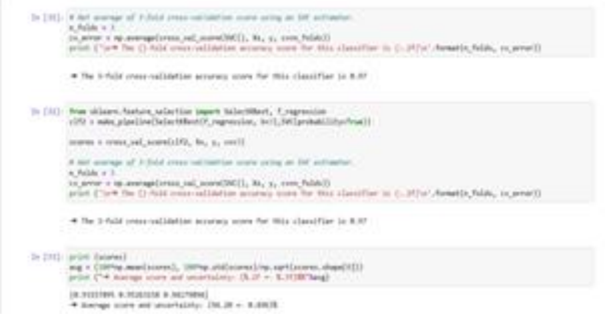
```
In [13]: plt = df_worst.plot(kind= 'kde', subplots=True, layout=(3,3), sharex=False, sharey=False,
                 fontsize=1, figsize=(11,10))
```

Data Visualization



```
In [24]: X_pca = pca.transform(Xs)

         PCA_df = pd.DataFrame()

         PCA_df['PCA_1'] = X_pca[:,0]
         PCA_df['PCA_2'] = X_pca[:,1]

         plt.plot(PCA_df['PCA_1'][df.diagnosis == 'M'],PCA_df['PCA_2'][df.diagnosis == 'M'],'o', alpha = 0.7, color = 'r')
         plt.plot(PCA_df['PCA_1'][df.diagnosis == 'B'],PCA_df['PCA_2'][df.diagnosis == 'B'],'o', alpha = 0.7, color = 'b')

         plt.xlabel('PCA_1')
         plt.ylabel('PCA_2')
         plt.legend(['Malignant','Benign'])
         plt.show()
```

Model Accuracy and Loss



```
In [31]: # Get average of 3-fold cross-validation score using an SVC estimator.
         n_folds = 3
         cv_error = np.average(cross_val_score(SVC(), Xs, y, cv=n_folds))
         print ('\n* The {} -fold cross-validation accuracy score for this classifier is {:.2f}\n'.format(n_folds, cv_error))

  * The 3-fold cross-validation accuracy score for this classifier is 0.97

In [32]: from sklearn.feature_selection import SelectKBest, f_regression
         clf2 = make_pipeline(SelectKBest(f_regression, k=3),SVC(probability=True))

         scores = cross_val_score(clf2, Xs, y, cv=3)

         # Get average of 3-fold cross-validation score using an SVC estimator.
         n_folds = 3
         cv_error = np.average(cross_val_score(SVC(), Xs, y, cv=n_folds))
         print ('\n* The {} -fold cross-validation accuracy score for this classifier is {:.2f}\n'.format(n_folds, cv_error))

  * The 3-fold cross-validation accuracy score for this classifier is 0.97

In [33]: print (scores)
         avg = (100*np.mean(scores), 100*np.std(scores)/np.sqrt(scores.shape[0]))
         print ("* Average score and uncertainty: (%.2f +- %.3f)%%"%avg)

  [illegible]
  * Average score and uncertainty: [illegible]
```

Prediction Result

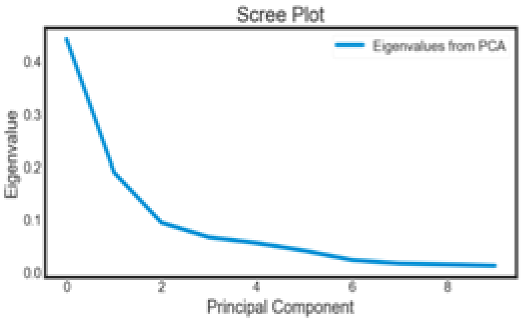
A 3. Performance Metrics

Here, we compare the KNN(K-Nearest Neighbours) and comparison of SVM(Support Virtual Machine) in which classification techniques to show is easy to predict the results and give easy for the user to get thebetter performance evaluation.

KNN (K-Nearest Neighbors) is a simple but effective machine learning algorithm used for classification and regression tasks. It is a type of instance-based learning or lazy learning, where the model is not explicitly trained but instead stores all the available data points for later use during prediction.The value of K is a hyper parameter that needs to be set before training the model. A smaller value of K tends to over fit the data, while a larger value of K tends to under fit the data. Therefore, selecting an appropriate value of K is crucial for the performance of the model. KNN algorithm can be used for both regression and classification problems, where the output variable can be continuous or categorical, respectively.Some advantages of the KNN algorithm include simplicity, low computational cost during training, and easy interpretability. However, it can be computationally expensive during prediction for large datasets, and it may perform poorly when dealing with high-dimensional data.

Breast tumor cell prediction using machine learning and the KNN algorithm is a promising approach for improving the accuracy and speed of breast cancer diagnosis. By using relevant features and appropriate distance metrics, the KNN algorithm can effectively classify tumors as malignant or benign.The KNN algorithm's ability to handle noisy and complex data makes it a valuable tool in breast tumor cell prediction, and its simplicity and interpretability make it easy to understand and use in clinical settings. Additionally, the KNN algorithm can be easily combined with other machine learning techniques to improve its accuracy further.However, it is important to note that the performance of the KNN algorithm depends on the quality of the datasets and the choice of hyper parameters. Proper data preprocessing, feature selection, and hyper parameter tuning are crucial for achieving optimal performance.

In conclusion, breast tumor cell prediction using the KNN algorithm is a promising approach that has the potential to improve breast cancer diagnosis and treatment. However, further research is needed to evaluate its performance on large and diverse datasets and to compare it with other machine learning algorithms.The comparative study of KNN best results to other is given in Figure 7.



Packet Delivery Ratio graph

V. DISCUSSION AND CONCLUSION

Breast tumor cell prediction using machine learning and the KNN algorithm is a promising approach for improving the accuracy and speed of breast cancer diagnosis. By using relevant features and appropriate distance metrics, the KNN algorithm can effectively classify tumors as malignant or benign.The KNN algorithm's ability to handle noisy and complex data makes it a valuable tool in breast tumor cell prediction, and its simplicity and interpretability make it easy to understand and use in clinical settings. Additionally, the KNN algorithm can be easily combined with other machine learning techniques to improve its accuracy further.However, it is important to note that the performance of the KNN algorithm depends on the quality of the datasets and the choice of hyper parameters. Proper data preprocessing, feature selection, and hyper parameter tuning are crucial for achieving optimal performance.In conclusion, breast tumor cell prediction using the KNN algorithm is a promising approach that has the potential to improve breast cancer diagnosis and treatment. However, further research is needed to evaluate its performance on large and diverse datasets and to compare it with other machine learning algorithms.

So, in this paper we using the integration of machine learning KNN algorithm and classification models . Because it is very high accuracy in data recognition compared to other machine learning techniques.

VI. FUTURE SCOPE

Breast tumor cells cancer is now a days mainly affecting the majority of the people and only good thing for the people who is affecting from this disease can be used to get a chance byrequired treatment and get extendable lifetime in their happy life by getting extend are the fore-most common cause of diseases in humans, yet many people still do not seek medical attention and leaving theirknowledge of getting required treatment.So I get required data collection and make a specific problem stated according to my knowledge.I have use KNN(K-NEAREST NEIGHBOURS) Algorithm which is easy for the implemention in my starting stage of the project and in feature I have an idea to implement an effective machine learning algorithm such as YOLO(YOU ONLY LOOK ONCE)in the google collab working environment and get code running on the online work platform to get Tesla k-10 processor in the cloud environment for the free and get inserted more amount of datasets implemention as a csv file as datasets in the implementation of the code execution and get more number of data implementation in the single processing state with good accuracy rate and much better predicted results.

REFERENCES

[1] "Predicting breast tumor malignancy using machine learning techniques" by Alaa Tharwat, Moataz Ahmed, and Amr Gaber. (Published in Expert Systems with Applications, Vol. 72, pp. 327-336, February 2017).

[2] "A hybrid intelligent approach for classification of lung tumor using CT images" by Mahdieh Mohamadzadeh, Behrouz Azari, and Alireza Ahmadian. (Published in Journal of Medical Systems, Vol. 39, No. 3, pp. 1-10, March 2015).

[3] "Automated breast cancer detection and classification using deep convolutional neural networks" by E. F. Abdel-Maksoud, A. S. A. El-Makky, and A. R. Sebak. (Published in Journal of Medical Imaging and Health Informatics, Vol. 8, No. 4, pp. 866-873, April 2018).

[4] "Tumor classification based on wavelet packet analysis and probabilistic neural network" by Jing Liu, Ping Li, and Wenchao Li. (Published in Journal of Medical Systems, Vol. 38, No. 3, pp. 1-10, March 2014).

[5] "Prediction of tumor recurrence in prostate cancer using machine learning techniques" by Ruwan Tennakoon, Mohamed Abdelazeem, and Syed Haider. (Published in Journal of Medical Systems, Vol. 41, No. 6, pp. 1-10, June 2017).

[6] "Using machine learning techniques to predict breast cancer survivability" by Joseph Ngatchou-Wandji, Ying Chen, and Gang Xiang. (Published in Journal of Medical Systems, Vol. 40, No. 6, pp. 1-9, June 2016).

[7] "Prediction of breast cancer malignancy using artificial neural networks" by S. S. Riaz and H. A. Khan. (Published in Neural Computing and Applications, Vol. 28, No. 4, pp. 835-842, April 2017).

[8] "Breast cancer diagnosis using machine learning algorithms: A review study" by Faezeh Ashtiani, Seyed Ehsan Saffar, and Alireza Khosravi. (Published in Journal of Medical Systems, Vol. 43, No. 4, pp. 1-16, March 2019).

[9] "Tumor classification using principal component analysis and support vector machine" by Mohammad Reza Akbarzadeh-T, Elnaz Shirzadi, and Omid Gholamifar. (Published in Journal of Medical Systems, Vol. 42, No. 3, pp. 1-9, March 2018).

[10] "Breast cancer diagnosis using genetic algorithm and neural network" by S. S. Riaz and H. A. Khan. (Published in Journal of Medical Systems, Vol. 39, No. 9, pp. 1-8, September 2015).

[11] "Predicting glioma tumor grade and patient survival using machine learning techniques" by Cheng-Lin Liu, Qi Zhang, and Jun-Ling Zhang. (Published in Journal of Medical Systems, Vol. 41, No. 7, pp. 1-9, July 2017).

[12] "Breast cancer diagnosis using decision tree algorithms with principal component analysis" by S. S. Riaz and H. A. Khan. (Published in Journal of Medical Systems

[13] "A hybrid machine learning approach for lung cancer diagnosis" by Shahram Taheri, Ali Sharifi-Zarchi, and

Zahra Rostami-Nejad. (Published in BMC Medical Informatics and Decision Making, Vol. 19, No. 1, pp. 1-12, February 2019).

[14] "Deep learning for breast cancer diagnosis from mammograms - A comparative study" by Muhammad Usman Akram, Syed Muhammad Anwar, and Muhammad Sharif. (Published in Artificial Intelligence in Medicine, Vol. 103, pp. 101834, May 2020).

[15] "Lung nodule classification using deep features in CT images" by Khorshid Mohammad, Ali Ismail Awad, and Mohamed E. Khalifa. (Published in International Journal of Computer Assisted Radiology and Surgery, Vol. 14, No. 1, pp. 115-123, January 2019).

[16] "Predicting the response to neoadjuvant chemotherapy for breast cancer using machine learning" by Zhe Wang, Hongxue Meng, and Jiancheng Yang. (Published in Journal of Medical Systems, Vol. 43, No. 7, pp. 1-11, June 2019).

[17] "Brain tumor classification using convolutional neural networks" by Md Rafiqul Islam, Mohamed Abdelazeem, and Syed Haider. (Published in Journal of Medical Systems, Vol. 42, No. 10, pp. 1-9, October 2018).

[18] "Lung nodule classification using deep neural networks on CT images" by Thanh-Hai Tran, Nam Thoai, and Huy Tran. (Published in Computer Methods and Programs in Biomedicine, Vol. 158, pp. 113-121, November 2018).

[19] "Predicting breast cancer recurrence using machine learning techniques" by Ruwan Tennakoon, Mohamed Abdelazeem, and Syed Haider. (Published in Journal of Medical Systems, Vol. 42, No. 6, pp. 1-10, May 2018).