VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Artificial Intelligence - CSE3013

Group - 11

Carcinoma Analysis and Detection Using Biopsy

Slot - A2+TA2

Rajeshkannan Sir

Advaith Chandra Srivastav

20BCE0983

(Vellore Institute of Technology, Vellore, India)

Aryan Sharma

20BCE0952

(Vellore Institute of Technology, Vellore, India)

Arsalan Ibrahim

20BCE0934

(Vellore Institute of Technology, Vellore, India)

Vipul Chaudhary

20BCE0076

(Vellore Institute of Technology, Vellore, India)

ABSTRACT:

As we know *Carcinoma has been a* common disease *for* a long time and looking at the world's situation more and *more doctors* are concentrating their time and efforts into finding the cures of new viruses in the world.

This calls for a way to automate the *analysis* and detection of carcinoma which can be done using our method of solution. Carcinoma in *Layman's* term is cancer which is a deadly disease, cancer is a disease in which some of the body's cells grow uncontrollably and spread to other parts of the body. To detect this, the doctors usually review the MRI scans and give suggestions to the patients on the next step that can be taken, which is usually **Biopsy** (if tumour is detected)

Our mechanism automatically detects if there is a tumour in the body using the MRI scan without the help of doctors. The next step is to check the biopsy report and after using our mechanism we can detect if the tumour is cancerous or non cancerous. This highly decreases the workload on the doctors and also decreases the waiting time for the patients. Thus, the next step can quickly be taken by the patient. The reviewing of a MRI scan requires a lot of knowledge about the subject which is available sparsely, i.e, something that only the specialised doctors have. The specialised doctors are few in number when compared to the number of patients of Carcinoma.

INTRODUCTION:

This Project revolves around one of the major problems in the entire world. Cancer, or, Carcinoma. Cancer is a disease in which some of the body’s cells grow uncontrollably and spread to other parts of the body. Cancer is a genetic disease—that is, it is caused by changes to genes that control the way our cells function, especially how they grow and divide.

Cancer can start almost anywhere in the human body, which is made up of trillions of cells. Normally, human cells grow and multiply (through a process called cell division) to form new cells as the body needs them. When cells grow old or become damaged, they die, and new cells take their place.

Taking into account all these details, and coming in contact with a few doctors in our vicinity, we analysed and gave a thorough research on this problem, mentioned in the abstract. Using this generation's Artificial Intelligence power, we have a very adaptive and powerful program through which we can easily know if a person has cancer or not. The only thing we require is an image of the MRI scan. Using CNN (Convolution Neural Network) and Keras, we can achieve what we aimed for.

The project also helps in differentiating between cancerous and non-cancerous *tumours* using the Biopsy report (if present). There are more than 100 types of cancer. We'll be using the concepts of Deep learning (Keras) to achieve our aim in this project. Since there are multiple types of Cancers (Level - 1, Level - 2, etc.) we can differentiate between them.

LITERATURE REVIEW:

PAPER TITLE **JOURNAL DETAILS**	**METHOD/** **ALGORITHM**	**CHALLENGES**	**OBSERVATIONS**
Sathyakumar K, Munoz M, Singh J, et al. (August 25, 2020) Automated Lung Cancer Detection Using Artificial Intelligence (AI) Deep Convolutional Neural Networks: A Narrative Literature Review. Cureus 12(8): e10017. doi:10.7759/cureus. 10017	PICO framework, a universally accepted standard within the research community that stands for the acronym PROBLEM: in this narrative review looks at lung cancer, INTERVENTION: this review also looks deeper into the artificial intelligence (AI) application comparing each research team's AI deep learning model performance, COMPARISON: this	The testing model should be tested on a much larger scale to ensure they work on large, real production data. The test dataset should not undergo image augmentation, but be tested from the original large dataset.	Minimum redundancy and maximum relevance (mRMR) feature selection method with ensemble CNN performed better than the methods described in the three other papers. It is the use of additional techniques such as image augmentation, principal component analysis (PCA), mRMR and appropriate feature selection that made this performance difference

	review also explores the deep learning ensemble convolutional neural network method (CNN) and compares with the four research groups' classic machine learning classifier performance. AI model performance OUTCOMES are measured for each research team model performance sensitivity measuring how well the algorithm recognizes the type of lung nodule correctly. The model accuracy measures the proportion of data that was classified correctly.		
D. Voth, "Using AI to detect breast cancer," in *IEEE Intelligent Systems*, vol. 20, no. 1, pp. 5-7, Jan.-Feb. 2005, doi: 10.1109/MIS.2005.14.	Use of image-enabled data mining software, based on the Soldier (source optimised lexicon digital expanded representations) query tool developed at Georgia Tech. The tool uses pixel-driven queries to find information doctors can draw on when making diagnoses.	The system lacks a large archive of encoded imagery to draw on at this time. It is envisioned that a database be built on the fly from clinical workflows at centers. Alternatively, a single institution or consortium might choose to build the historical archive.	The system analyzes the relevant imagery and associated point data to make an inference that can help the doctor make a decision in a clinical situation.

Nehmat Houssami, Georgia Kirkpatrick-Jones, Naomi Noguchi & Christoph I. Lee (2019) Artificial Intelligence (AI) for the early detection of breast cancer: a scoping review to assess AI’s potential in breast screening practice, Expert Review of Medical Devices, 16:5, 351-362, DOI: 10.1080/17434440.2019.1610387	A scoping review was performed to assess and summarize, in a structured manner, the evidence on the use of AI in breast cancer detection. Given the heterogeneity of research in this field, conventional data synthesis using standard systematic reviews or meta-analysis would not be appropriate. Scoping reviews allow evidence mapping and synthesis from a variety of studies and sources to address broad research questions and identify evidence gaps.	In the context of breast cancer, ongoing research using AI for early detection includes a global effort attempting to develop advanced machine learning algorithms for interpreting screening mammograms to potentially improve breast cancer screening by reducing false-positives. The potential application of AI in breast cancer diagnostics extends to imaging modalities and also pathology interpretation.	Scoping review of studies of AI for breast cancer detection showed predominantly retrospective studies based on relatively small and highly selected image datasets. Although the reviewed studies used novel techniques, reported encouraging results for AI model accuracy. The methodologic issues highlighted in their work can help inform future studies and improve the translation of AI systems into breast cancer screening practice.
José Luis Raya-Povedano , Sara Romero-Martín, Esperanza Elías-Cabot, Albert Gubern-Mérida, Alejandro Rodríguez-Ruiz, Marina Álvarez-Benito (May 4, 2021) AI-based Strategies to Reduce Workload in Breast	Consecutive screening-paired and independently read DM and DBT images An AI system computed a cancer risk score for DM and DBT examinations independently. Each original setting was compared with a simulated autonomous AI	Limitations include that the study was only performed with data from a single site and single mammography and AI vendor.It is a retrospective study where the AI scenarios are simulated, it is not possible to know the impact on radiologists’ performance in the	In conclusion, the study shows a strategy with an AI system where screening workload could be safely reduced up to 70% for both digital mammography –and digital breast tomosynthesis –based programs, as well as allow the transition from DM- to DBT-based

Cancer Screening with Mammography and Tomosynthesis: A Retrospective Evaluation, https://doi.org/10.1148/radiol.2021203555	triaging strategy in terms of workload, sensitivity, and recall rate. The McNemar test with Bonferroni correction was used for statistical analysis. The hypothesis was that in the AI-based strategy, workload could be significantly reduced, with noninferior sensitivity and recall rate Noninferiority was concluded if the sensitivity or the recall rate was superior.	setting where they might read only the 30% most suspicious screening examinations.	screening without an increase in workload. Given the increasing lack of expert breast radiologists as well as the increased workload associated with the introduction of DBT, new strategies potentially using AI could be necessary to maintain the cost-efficiency of screening programs. Further prospective studies are needed to validate their findings.
Kohlberger, T., Liu, Y., Moran, M., Chen, P. C., Brown, T., Hipp, J. D., Mermel, C. H., & Stumpe, M. C. (2019). Whole-Slide Image Focus Quality: Automatic Assessment and Impact on AI Cancer Detection. *Journal of pathology informatics*, *10*, 39. https://doi.org/10.4103/jpi.jpi_11_19	A convolution Neural network was developed (ConvFocus) to exhaustively localise and quantify the severity of OOF regions on digitised slides. ConvFocus was developed using their refined semi-synthetic OOF data generation process and evaluated using seven slides spanning three different tissue and	Digital pathology enables remote access or consults and powerful image analysis algorithms. However, the slide digitization process can create artefacts such as out-of-focus (OOF). OOF is often only detected on careful review, potentially causing rescanning, and workflow delays. Although scan time operator screening for whole-slide	When compared to pathologist-graded focus quality, ConvFocus achieved Spearman rank coefficients of 0.81 and 0.94 on two scanners and reproduced the expected OOF patterns from z-stack scanning. the impact of OOF on the accuracy of a state-of-the-art metastatic breast cancer detector was also evaluated

	three different stain types, each of which were digitized using two different whole-slide scanner models ConvFocus's predictions were compared with pathologist-annotated focus quality grades across 514 distinct regions representing 37,700 35 µm × 35 µm image patches, and 21 digitized “z-stack” WSIs that contain known OOF patterns.	OOF is feasible, manual screening for OOF affecting only parts of a slide is impractical.	and saw a consistent decrease in performance with increasing OOF.
Alexander J. T. Wanders , Willem Mees, Petra A.M. Bun, Natasja Janssen, Alejandro Rodríguez-Ruiz, Mehmet Ufuk Dalmış, Nico Karssemeijer, Carla H. van Gils, Ioannis Sechopoulos, Ritse M. Mann, Cornelis Jan van Rooden (February 8, 2022) https://doi.org/10.1148/radiol.210832	This retrospective nested case-control study performed with screening examinations included women who developed IC and women with normal follow-up findings (from January 2011 to January 2015). An AI cancer detection system analyzed all studies yielding a score of 1–10, representing increasing likelihood of malignancy. BD	The model cannot assess the performance of the risk model in the detection of stage IV cancers or its long-term effect to reduce morbidity and mortality within the screening program. The model was trained and tested using screening examinations from one vendor. It has a retrospective character. it does not allow estimates of the probability of	A total of 2222 women with IC and 4661 women in the control group were included (mean age, 61 years; age range, 49–76 years). AUC of the NN model was 0.79 (95% CI: 0.77,0.81), which was higher than AUC of the AI cancer detection system or BD alone (AUC, 0.73 [95% CI: 0.71, 0.76] and

	was automatically computed using publicly available software. An NN model was trained by combining the AI score and BD using 10-fold cross-validation. Bootstrap analysis was used to calculate the area under the receiver operating characteristic curve (AUC), sensitivity at 90% specificity, and 95% CIs of the AI, BD, and NN models.	breast cancer in a particular patient.	0.69 [95% CI: 0.67, 0.71], respectively; *P* < .001 for both). At 90% specificity, the NN model had a sensitivity of 50.9% (339 of 666 women; 95% CI: 45.2, 56.3) for prediction of IC, which was higher than that of the AI system (37.5%; 250 of 666 women; 95% CI: 33.0, 43.7; *P* < .001) or BD percentage alone (22.4%; 149 of 666 women; 95% CI: 17.9, 28.5; *P* < .001).
Merali, Z., Wang, J.Z., Badhiwala, J.H. *et al.* A deep learning model for detection of cervical spinal cord compression in MRI scans. *Sci Rep* 11, 10473 (2021). https://doi.org/10.1038/s41598-021-89848-3	Pre-processing Unlike many other recent deep learning approaches which use the whole of the image, they only focus on a limited area of it to extract key features. Removing unnecessary Parts the negative results reduce drastically. Similar distributions	Although the proposed approach's outstanding results compared to the other recently published models, the algorithm has still limitations when encountering tumour volume of more than one-third of the whole of the brain. This is because of an increase in the size of the tumour's	In this paper, They developed a new brain tumour segmentation architecture that benefits from the characterization of the four MRI modalities. They demonstrated that working only on a part of the brain image near the tumour tissue allows a CNN model to reach performance close to human observers. This leads to reducing

	To improve the final segmentation accuracy, they used four brain modalities, namely T1, FLAIR, T1C, and T226,27. To enforce the MRI data more uniform and remove the effect of the anisotropic (especially for the FLAIR modality), they conducted the Z-Score normalisation for the used modalities. By applying this approach to a medical brain image, the output image has zero mean and unit variance24. They implemented this step by subtracting the mean and dividing by the standard deviation in only the brain region (not the background). This step was implemented independently for each brain volume of every patient.	expected area which leads to a decrease in the feature extraction performance.	the computational time and capability to make predictions fast for classifying the clinical image.

Ranjbarzadeh, R., Bagherian Kasgari, A., Jafarzadeh Ghoushchi, S. *et al.* Brain tumor segmentation based on deep learning and an attention mechanism using MRI multi-modalities brain images. *Sci Rep* 11, 10930 (2021). https://doi.org/10.1038/s41598-021-90428-8	In their study, they proposed a simple CNN model, they extracted the augmented MRI image data of 224 × 224 input size having RGB Color channels with a batch size of 32 through their CNN model. Initially, they added a single 16 filters convolutional layer having a filter size of 3 × 3. The reason for placing a small number of filters as 16 is to detect edges, corners, and lines. And then a max-pooling layer with 2 × 2 filter was added on it to get the max summary of that image, then they increased the number of convolutional layers and the number of filters to 32, 64, and 128, having the same filter size of 3×3. This combines these small patterns as the number of filters increases and finds bigger patterns like a circle, a square, etc. And they applied max-pooling layers on top of those convolutional layers	The number of true negatives comes out to be very high which raises questions on the precision and accuracy of the design. The design also doesn't do categorical classification that is telling the type of cancer the patient has.	In this paper, a new approach was presented to classify brain tumors. First, using the image edge detection technique, they find the region of interest in MRI images and cropped them. Second, they provide an efficient methodology for brain tumour classification by proposing a simple CNN network. Their proposed system can play an effective role in the early diagnosis of dangerous disease in other clinical domains related to medical imaging, particularly lung cancer and breast cancer.

	to get the most of it. Finally, they applied a fully connected dense layer of 256 neurons along with the softmax output layer that calculates the probability score for each class and classifies the final decision labels that either the input MRI image contains cancer or does not contain cancer in Yes or No.		
Hassan Ali Khan , Wu Jue, Muhammad Mushtaq and Muhammad Umer Mushtaq *et al*. Brain tumor classification in MRI image using convolutional neural network 2020, Volume 17, Issue 5: 6203-6216. doi: 10.3934/mbe.2020328	This study proposes a method of multi-level features extraction and concatenation for early diagnosis of brain tumor. Two pre-trained deep learning models i.e. Inception-v3 and DensNet201 make this model valid. With the help of these two models, two different scenarios of brain tumor detection and its classification were evaluated. First, the features from different Inception modules were extracted from pre-trained Inception-v3 model and concatenated these features for brain tumor	One of the limitations in exisiting approaches is feature extraction from bottom layers of pre-trained models which are different from natural images to medical images. To overcome this problem, a method of multi-level features extraction is proposed which enhances the capability of the model to classify the brain tumor.	The proposed method produced 99.51% testing accuracy on testing samples and achieved the highest performance in detection of brain tumor. The ensemble method based on concatenation of dense block by using DensNet201 pretrained model outperformed as compared to the current research methods for brain tumor classification problem. Both scenarios were evaluated with the publicly available three-class brain tumor dataset.

	classification		
Noreen, Neelum & Palaniappan, Sellapan & Qayyum, Abdul & Ahmad, Iftikhar & Imran, Muhammad & Shoaib, Muhammad. (2020). A Deep Learning Model Based on Concatenation Approach for the Diagnosis of Brain Tumor. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.2978629.	This study involved retrospective analysis of prospectively collected magnetic resonance imaging (MRI) studies from patients. Patients were enrolled if they met eligibility criteria as follows: (1) Age 18 years or older; (2) imaging evidence of cervical spinal cord compression; (3) symptomatic DCM with one or more signs of myelopathy; and (4) no prior cervical spine surgery. All patients had a pre-operative MRI scan then underwent surgical decompression of the cervical spine, with or without instrumented fusion. Patients were subsequently followed for 2 years after surgery. All patients in this study had imaging evidence of cervical spinal cord compression. However, the spinal	The dataset only included patients who had a confirmed diagnosis of DCM and went on to get surgery. Patients with mild DCM or normal MRI scans were therefore underrepresented in their dataset. Many patients in the dataset did not have MRI scans that were in an appropriate format.	In recent years automated diagnostic tools have reached a substantial level of development and this progress is expected to continue. In this study they trained and tested a CNN model to detect spinal cord compression in cervical spine MRI scans. They achieved high model performance with an AUC of 0.94 on a heterogenous group of patients. They demonstrated the feasibility of training an existing CNN for a novel medical imaging classification task. Future work will need to focus on developing larger datasets to facilitate the development of a more generalized model capable of quantifying cord signal change, severity of spinal cord compression, cervical spine deformity, and nerve root compression. A more generalized

	cord compression typically affected only a portion of the cervical spine. In each MRI scan there were some spinal levels that were compressed and other spinal levels that were not compressed. Thus, they were able to obtain images of spinal cord compression and non-compressed spinal cords from this patient cohort for model training.		model may be able to improve radiology workflows and augment clinical decision-making by increasing the efficiency and objectivity of cervical spine MRI interpretation.

PROBLEM STATEMENT:

Doctor to patient ratio is 1:1000 according to WHO which is very small and we collectively as a group thought that this is a major issue and should be acted upon, looking at the current worldly matters that is covid and many more upcoming diseases.

This project focuses in investigating the probability of predicting the type of breast cancer (malignant or benign) from the given characteristics of breast mass computed from digitized images. This project will examine the data available and attempt to predict the possibility that a breast cancer diagnosis

METHODOLOGY/ MODULE WISE DESCRIPTION/:

Pre Processing — Data Preparation

We used the UCI Machine Learning Repository for breast cancer dataset.

Pre Processing — Data Exploration

We used google collab to work on this dataset. We will first go with importing the necessary libraries and import our dataset to google collab.

Phase 1 — Categorical Data

Categorical data are variables that contain label values rather than numeric values.The number of possible values is often limited to a fixed set.

Phase 2 — Feature Scaling

Most of the times, your dataset will contain features highly varying in magnitudes, units and range. But since, most of the machine learning algorithms use Eucledian distance between two data points in their computations. We need to bring all features to the same level of magnitudes. This can be achieved by scaling. This means that you’re transforming your data so that it fits within a specific scale.

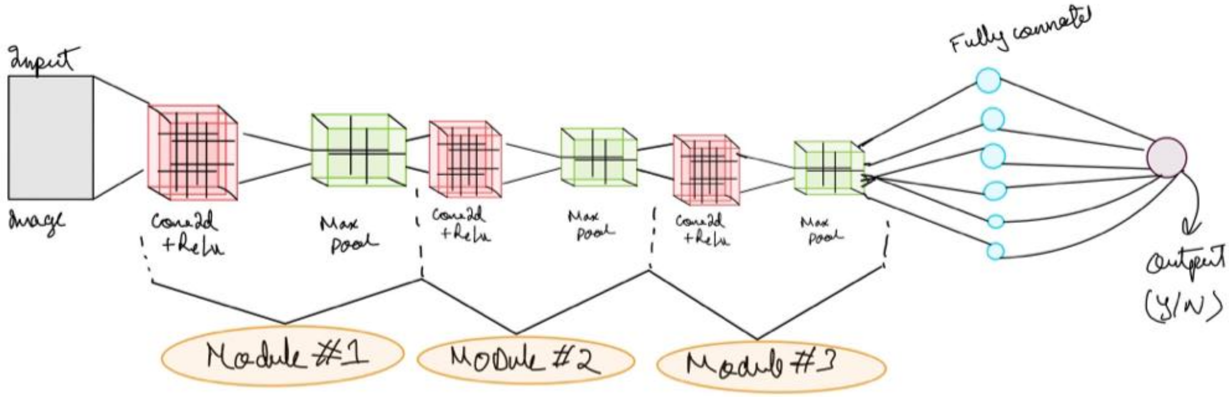
Phase 3 — Model Selection

This is the most exciting phase in Applying Machine Learning to any Dataset. It is also known as Algorithm selection for Predicting the best results. Usually Data Scientists use different kinds of Machine Learning algorithms to the large data sets. But, at high level all those different algorithms can be classified in two groups : supervised learning and unsupervised learning. A classification problem is when the output variable is a category like filtering emails “spam” or “not spam”

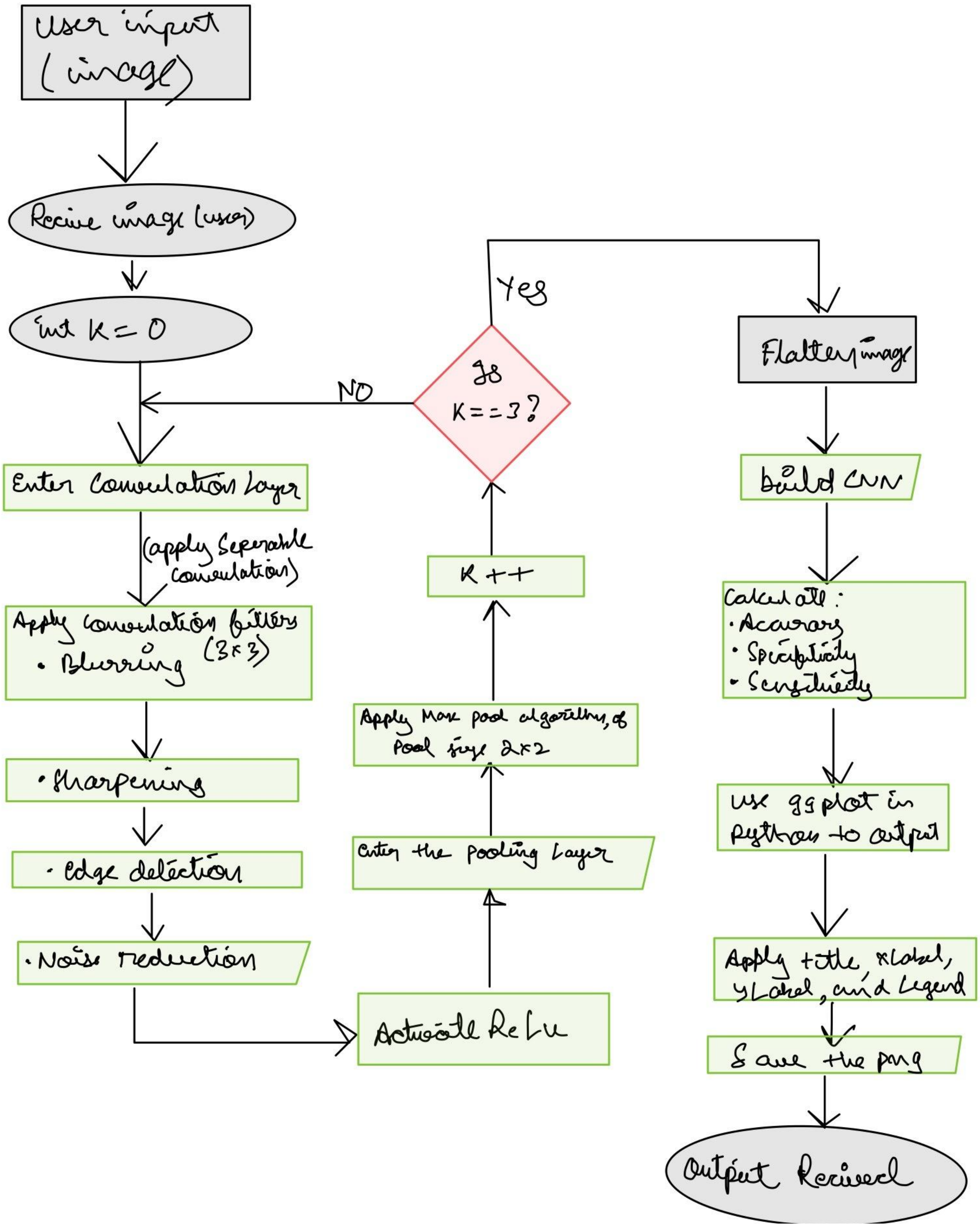
Unsupervised Learning :

Unsupervised learning is the algorithm using information that is neither classified nor labelled and allowing the algorithm to act on that information without guidance.In our dataset we have the outcome variable or Dependent variable i.e Y having only two set of values, either M (Malign) or B(Benign). So we will use Classification algorithm of supervised learning

NEURAL NETWORK DIAGRAM:



FLOW DIAGRAM:


User input (image)
Recive image (user)
int K = 0
Enter Convolution Layer
(apply Seperable convolution)
Apply convolution filters (3×3)
• Blurring
• Sharpening
• Edge detection
• Noise reduction
Activate ReLu
enter the pooling Layer
Apply Max pool algorithm, of Pool size 2×2
K ++
Is K == 3?
NO
Yes
Flatten image
build CNN
Calculate:
• Accuracy
• Specificity
• Sensitivity
Use ggplot in python to output
Apply title, xlabel, ylabel, and Legend
Save the png
Output Recived

PSEUDOCODE: Model - CNN Conv1D

1) This trains and evaluates our model. Here, we'll import from keras, sklearn, cancernet, config, imutils, matplotlib, numpy, and os.

2) Initialize the following -

   NUM_EPOCHS=40; INIT_LR=1e-2; BS=32

3) Do the changes -

```
trainPaths=list(paths.list_images(config.TRAIN_PATH))
lenTrain=len(trainPaths)
lenVal=len(list(paths.list_images(config.VAL_PATH)))
lenTest=len(list(paths.list_images(config.TEST_PATH)))
trainLabels=[int(p.split(os.path.sep)[-2]) FOR p IN trainPaths]
trainLabels=np_utils.to_categorical(trainLabels)
```

4) Set them

```
    classTotals=trainLabels.sum(axis=0)
    classWeight=classTotals.max()/classTotals

    SET trainAug TO ImageDataGenerator(

            rescale=1/255.0,
            rotation_range=20,
            zoom_range=0.05,
            width_shift_range=0.1,
            height_shift_range=0.1,
            shear_range=0.05,
            horizontal_flip=True,
            vertical_flip=True,
            fill_mode="nearest")

valAug=ImageDataGenerator(rescale=1 / 255.0)

SET trainGen TO trainAug.flow_from_directory(
    config.TRAIN_PATH,
    class_mode="categorical",
    target_size=(48,48),
    color_mode="rgb",
    shuffle=True,
    batch_size=BS)
```

```
SET valGen TO valAug.flow_from_directory(

        config.VAL_PATH,
        class_mode="categorical",
        target_size=(48,48),
        color_mode="rgb",
        shuffle=False,
        batch_size=BS)

SET testGen TO valAug.flow_from_directory(

        config.TEST_PATH,
        class_mode="categorical",
        target_size=(48,48),
        color_mode="rgb",
        shuffle=False,
        batch_size=BS)
```

5) Doing model

```
model=CancerNet.build(width=48,height=48,depth=3,classes=2)
opt=Adagrad(lr=INIT_LR,decay=INIT_LR/NUM_EPOCHS)

model.compile(loss="binary_crossentropy",optimizer=opt,metrics=["accuracy"])


M=model.fit_generator(
        trainGen,
        steps_per_epoch=lenTrain//BS,
        validation_data=valGen,
        validation_steps=lenVal//BS,
        class_weight=classWeight,
        epochs=NUM_EPOCHS)
```

6) Obtain output

```
OUTPUT("Now evaluating the model")
testGen.reset()
pred_indices=model.predict_generator(testGen,steps=(lenTest//BS)+1)

pred_indices=np.argmax(pred_indices,axis=1)
```

```
OUTPUT(classification_report(testGen.classes, pred_indices,
target_names=testGen.class_indices.keys()))

cm=confusion_matrix(testGen.classes,pred_indices)
total=sum(sum(cm))

accuracy=(cm[0,0]+cm[1,1])/total
specificity=cm[1,1]/(cm[1,0]+cm[1,1])
sensitivity=cm[0,0]/(cm[0,0]+cm[0,1])

OUTPUT(cm)

OUTPUT(f'Accuracy: {accuracy}')

OUTPUT(f'Specificity: {specificity}')

OUTPUT(f'Sensitivity: {sensitivity}')

SET N TO NUM_EPOCHS

Output -
plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0,N), M.history["loss"], label="train_loss")
plt.plot(np.arange(0,N), M.history["val_loss"], label="val_loss")
plt.plot(np.arange(0,N), M.history["acc"], label="train_acc")
plt.plot(np.arange(0,N), M.history["val_acc"], label="val_acc")
plt.title("Training Loss and Accuracy on the IDC Dataset")
plt.xlabel("Epoch No.")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig('plot.png')
```

………0

2) The network we’ll build will be a CNN (Convolutional Neural Network) and call it CancerNet. This network performs the following operations:

-Use 3×3 CONV filters
-Stack these filters on top of each other

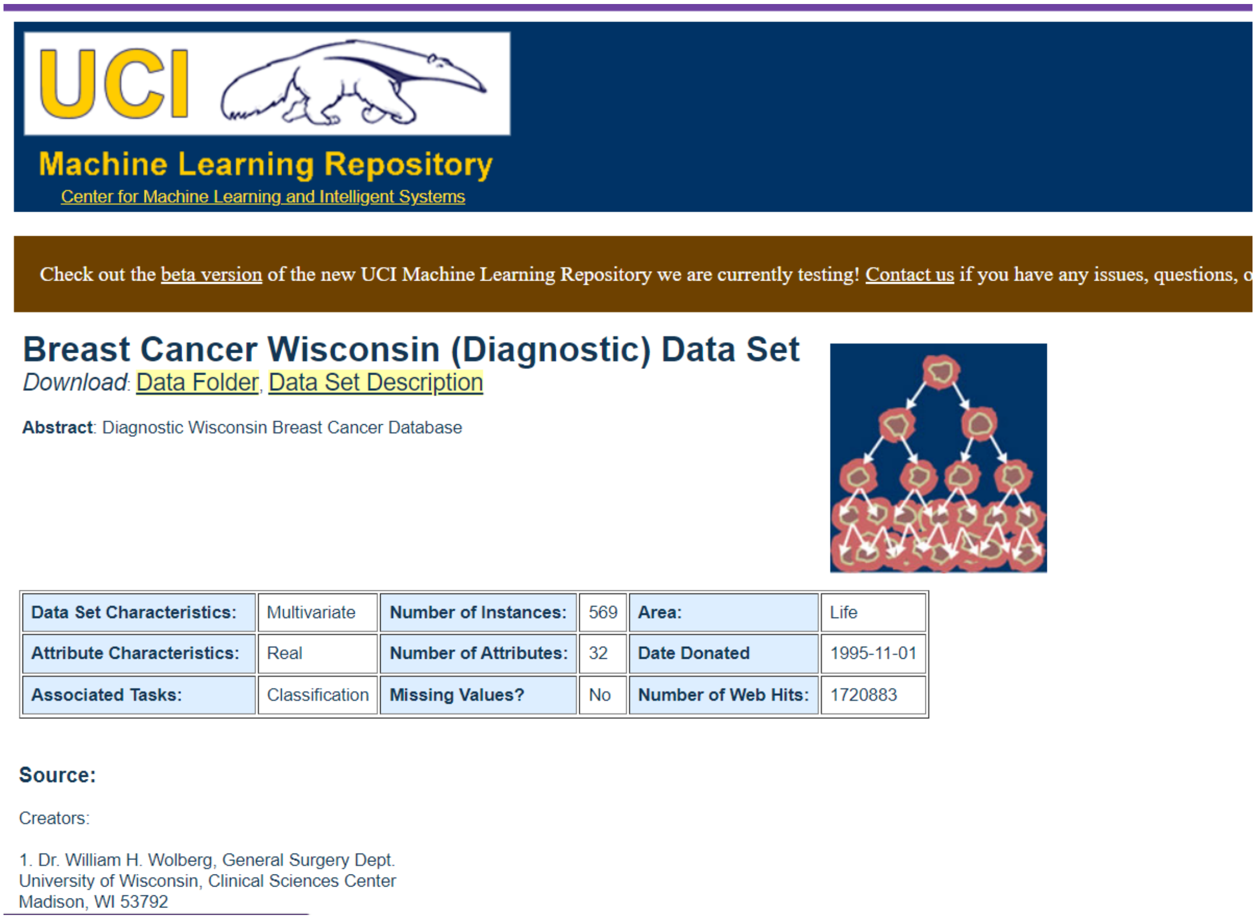
-Perform max-pooling
-Use depthwise separable convolution (more efficient, takes up less memory)

```
DEFINE CLASS CancerNet:
    @staticmethod
    DEFINE FUNCTION build(width,height,depth,classes):

        model=Sequential()
        shape=(height,width,depth)
        channelDim=-1

        IF K.image_data_format()=="channels_first":
            shape=(depth,height,width)
            channelDim=1

        model.add(SeparableConv2D(32, (3,3), padding="same",INPUT_shape=shape))
        model.add(Activation("relu"))
        model.add(BatchNormalization(axis=channelDim))
        model.add(MaxPooling2D(pool_size=(2,2)))
        model.add(Dropout(0.25))
```

—---

EXPERIMENT AND RESULTS:

Pre Processing — Data Preparation



UCI
Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Check out the beta version of the new UCI Machine Learning Repository we are currently testing! Contact us if you have any issues, questions, o

Breast Cancer Wisconsin (Diagnostic) Data Set

Download: Data Folder, Data Set Description

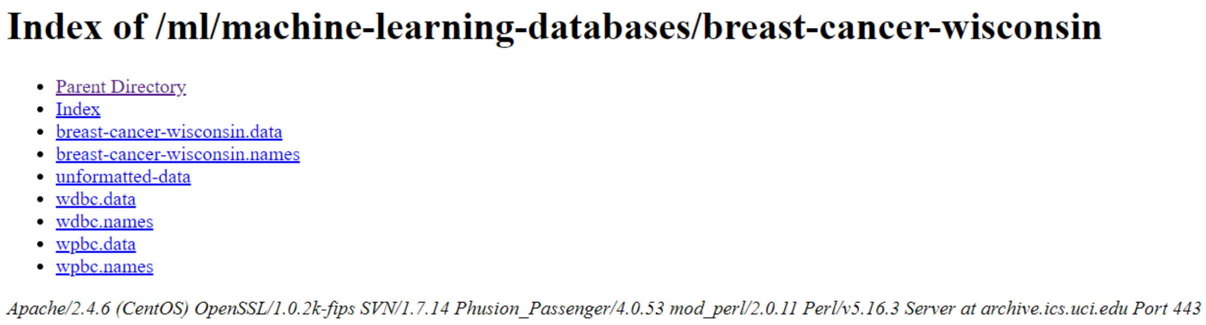
Abstract: Diagnostic Wisconsin Breast Cancer Database

Data Set Characteristics:	Multivariate	**Number of Instances:**	569	**Area:**	Life
Attribute Characteristics:	Real	**Number of Attributes:**	32	**Date Donated**	1995-11-01
Associated Tasks:	Classification	**Missing Values?**	No	**Number of Web Hits:**	1720883

Source:

Creators:

1. Dr. William H. Wolberg, General Surgery Dept.
University of Wisconsin, Clinical Sciences Center
Madison, WI 53792



Index of /ml/machine-learning-databases/breast-cancer-wisconsin

- Parent Directory
- Index
- breast-cancer-wisconsin.data
- breast-cancer-wisconsin.names
- unformatted-data
- wdbc.data
- wdbc.names
- wpbc.data
- wpbc.names

Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips SVN/1.7.14 Phusion_Passenger/4.0.53 mod_perl/2.0.11 Perl/v5.16.3 Server at archive.ics.uci.edu Port 443

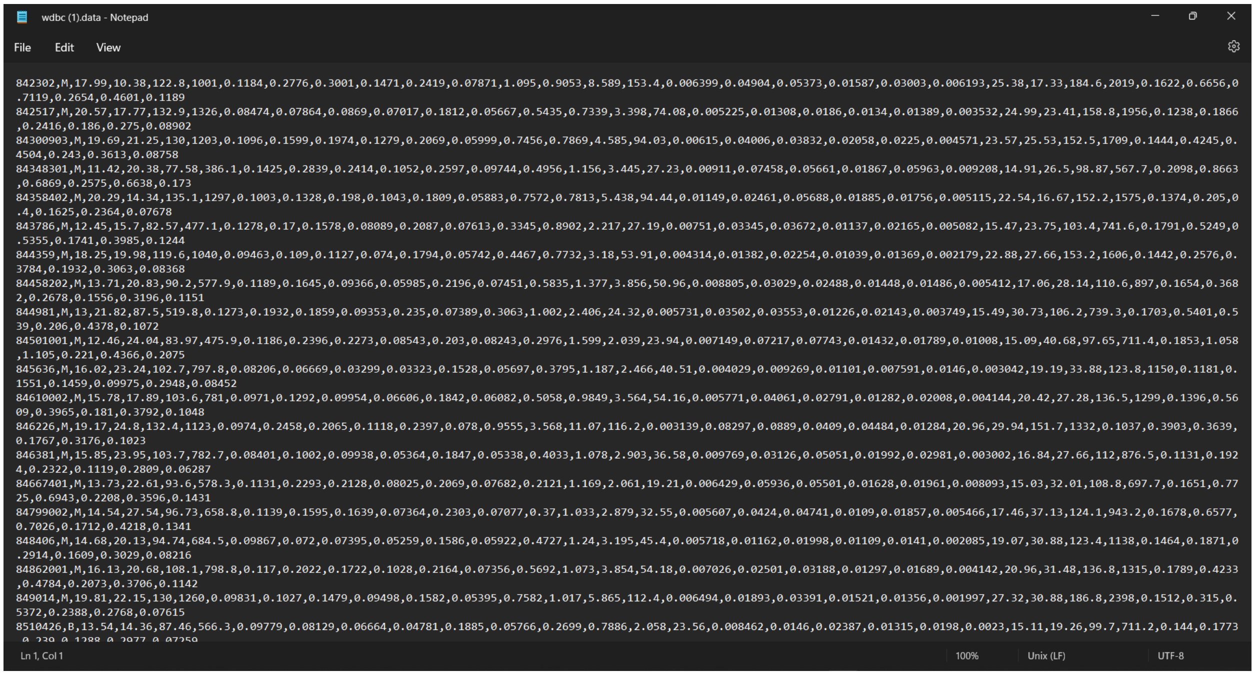
Index of breast-cancer-wisconsin

02 Dec 1996	*326*	*Index*
05 Feb 1996	*124103*	*wdbc.data*
05 Feb 1996	*4708*	*wdbc.names*
01 Feb 1996	*44234*	*wpbc.data*
01 Feb 1996	*5671*	*wpbc.names*
16 Jul 1992	*19889*	*breast-cancer-wisconsin.data*
16 Jul 1992	*5657*	*breast-cancer-wisconsin.names*
16 Jul 1992	*21363*	*unformatted-data*

Pre Processing — Data Exploration

```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import Sequential
from    tensorflow.keras.layers    import    Flatten,    Dense,    Dropout,
BatchNormalization
from tensorflow.keras.layers import Conv1D, MaxPool1D
from tensorflow.keras.optimizers import Adam
import pandas as pd
import numpy as np
import  seaborn as sns
import matplotlib.pyplot as plt
from sklearn import datasets, metrics
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

Phase 1 — Categorical Data



```
842302,M,17.99,10.38,122.8,1001,0.1184,0.2776,0.3001,0.1471,0.2419,0.07871,1.095,0.9053,8.589,153.4,0.006399,0.04904,0.05373,0.01587,0.03003,0.006193,25.38,17.33,184.6,2019,0.1622,0.6656,0
.7119,0.2654,0.4601,0.1189
842517,M,20.57,17.77,132.9,1326,0.08474,0.07864,0.0869,0.07017,0.1812,0.05667,0.5435,0.7339,3.398,74.08,0.005225,0.01308,0.0186,0.0134,0.01389,0.003532,24.99,23.41,158.8,1956,0.1238,0.1866
,0.2416,0.186,0.275,0.08902
84300903,M,19.69,21.25,130,1203,0.1096,0.1599,0.1974,0.1279,0.2069,0.05999,0.7456,0.7869,4.585,94.03,0.00615,0.04006,0.03832,0.02058,0.0225,0.004571,23.57,25.53,152.5,1709,0.1444,0.4245,0.
4504,0.243,0.3613,0.08758
84348301,M,11.42,20.38,77.58,386.1,0.1425,0.2839,0.2414,0.1052,0.2597,0.09744,0.4956,1.156,3.445,27.23,0.00911,0.07458,0.05661,0.01867,0.05963,0.009208,14.91,26.5,98.87,567.7,0.2098,0.8663
,0.6869,0.2575,0.6638,0.173
84358402,M,20.29,14.34,135.1,1297,0.1003,0.1328,0.198,0.1043,0.1809,0.05883,0.7572,0.7813,5.438,94.44,0.01149,0.02461,0.05688,0.01885,0.01756,0.005115,22.54,16.67,152.2,1575,0.1374,0.205,0
.4,0.1625,0.2364,0.07678
843786,M,12.45,15.7,82.57,477.1,0.1278,0.17,0.1578,0.08089,0.2087,0.07613,0.3345,0.8902,2.217,27.19,0.00751,0.03345,0.03672,0.01137,0.02165,0.005082,15.47,23.75,103.4,741.6,0.1791,0.5249,0
.5355,0.1741,0.3985,0.1244
844359,M,18.25,19.98,119.6,1040,0.09463,0.109,0.1127,0.074,0.1794,0.05742,0.4467,0.7732,3.18,53.91,0.004314,0.01382,0.02254,0.01039,0.01369,0.002179,22.88,27.66,153.2,1606,0.1442,0.2576,0.
3784,0.1932,0.3063,0.08368
84458202,M,13.71,20.83,90.2,577.9,0.1189,0.1645,0.09366,0.05985,0.2196,0.07451,0.5835,1.377,3.856,50.96,0.008805,0.03029,0.02488,0.01448,0.01486,0.005412,17.06,28.14,110.6,897,0.1654,0.368
2,0.2678,0.1556,0.3196,0.1151
844981,M,13,21.82,87.5,519.8,0.1273,0.1932,0.1859,0.09353,0.235,0.07389,0.3063,1.002,2.406,24.32,0.005731,0.03502,0.03553,0.01226,0.02143,0.003749,15.49,30.73,106.2,739.3,0.1703,0.5401,0.5
39,0.206,0.4378,0.1072
84501001,M,12.46,24.04,83.97,475.9,0.1186,0.2396,0.2273,0.08543,0.203,0.08243,0.2976,1.599,2.039,23.94,0.007149,0.07217,0.07743,0.01432,0.01789,0.01008,15.09,40.68,97.65,711.4,0.1853,1.058
,1.105,0.221,0.4366,0.2075
845636,M,16.02,23.24,102.7,797.8,0.08206,0.06669,0.03299,0.03323,0.1528,0.05697,0.3795,1.187,2.466,40.51,0.004029,0.009269,0.01101,0.007591,0.0146,0.003042,19.19,33.88,123.8,1150,0.1181,0.
1551,0.1459,0.09975,0.2948,0.08452
84610002,M,15.78,17.89,103.6,781,0.0971,0.1292,0.09954,0.06606,0.1842,0.06082,0.5058,0.9849,3.564,54.16,0.005771,0.04061,0.02791,0.01282,0.02008,0.004144,20.42,27.28,136.5,1299,0.1396,0.56
09,0.3965,0.181,0.3792,0.1048
846226,M,19.17,24.8,132.4,1123,0.0974,0.2458,0.2065,0.1118,0.2397,0.078,0.9555,3.568,11.07,116.2,0.003139,0.08297,0.0889,0.0409,0.04484,0.01284,20.96,29.94,151.7,1332,0.1037,0.3903,0.3639,
0.1767,0.3176,0.1023
846381,M,15.85,23.95,103.7,782.7,0.08401,0.1002,0.09938,0.05364,0.1847,0.05338,0.4033,1.078,2.903,36.58,0.009769,0.03126,0.05051,0.01992,0.02981,0.003002,16.84,27.66,112,876.5,0.1131,0.192
4,0.2322,0.1119,0.2809,0.06287
84667401,M,13.73,22.61,93.6,578.3,0.1131,0.2293,0.2128,0.08025,0.2069,0.07682,0.2121,1.169,2.061,19.21,0.006429,0.05936,0.05501,0.01628,0.01961,0.008093,15.03,32.01,108.8,697.7,0.1651,0.77
25,0.6943,0.2208,0.3596,0.1431
84799002,M,14.54,27.54,96.73,658.8,0.1139,0.1595,0.1639,0.07364,0.2303,0.07077,0.37,1.033,2.879,32.55,0.005607,0.0424,0.04741,0.0109,0.01857,0.005466,17.46,37.13,124.1,943.2,0.1678,0.6577,
0.7026,0.1712,0.4218,0.1341
848406,M,14.68,20.13,94.74,684.5,0.09867,0.072,0.07395,0.05259,0.1586,0.05922,0.4727,1.24,3.195,45.4,0.005718,0.01162,0.01998,0.01109,0.0141,0.002085,19.07,30.88,123.4,1138,0.1464,0.1871,0
.2914,0.1609,0.3029,0.08216
84862001,M,16.13,20.68,108.1,798.8,0.117,0.2022,0.1722,0.1028,0.2164,0.07356,0.5692,1.073,3.854,54.18,0.007026,0.02501,0.03188,0.01297,0.01689,0.004142,20.96,31.48,136.8,1315,0.1789,0.4233
,0.4784,0.2073,0.3706,0.1142
849014,M,19.81,22.15,130,1260,0.09831,0.1027,0.1479,0.09498,0.1582,0.05395,0.7582,1.017,5.865,112.4,0.006494,0.01893,0.03391,0.01521,0.01356,0.001997,27.32,30.88,186.8,2398,0.1512,0.315,0.
5372,0.2388,0.2768,0.07615
8510426,B,13.54,14.36,87.46,566.3,0.09779,0.08129,0.06664,0.04781,0.1885,0.05766,0.2699,0.7886,2.058,23.56,0.008462,0.0146,0.02387,0.01315,0.0198,0.0023,15.11,19.26,99.7,711.2,0.144,0.1773
,0.239,0.1288,0.2977,0.07259
```

Phase 2 — Feature Scaling

Most of the times, your dataset will contain features highly varying in magnitudes, units and range. But since, most of the machine learning algorithms use Eucledian distance between two data points in their computations. We need to bring all features to the same level of magnitudes. This can be achieved by scaling. This means that you're transforming your data so that it fits within a specific scale.

```
X_train = X_train.reshape(455,30,1)
X_test = X_test.reshape(114, 30, 1)
```

Phase 3 — Model Selection

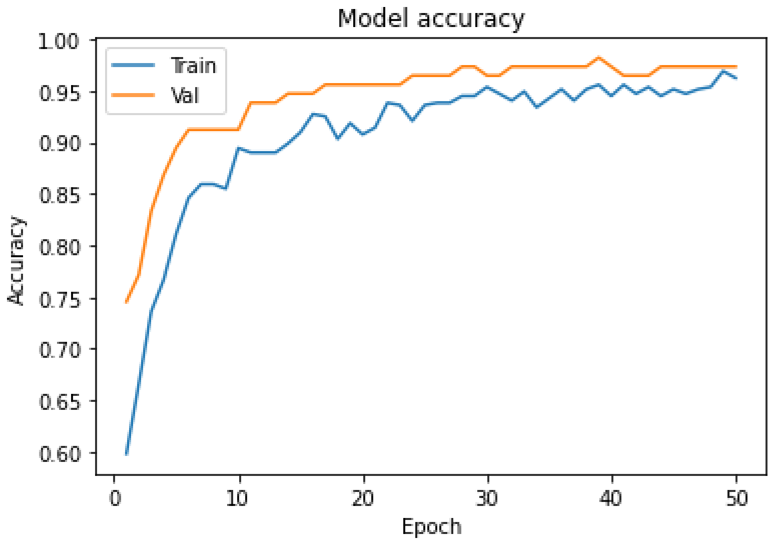
This is the most exciting phase in Applying Machine Learning to any Dataset. It is also known as Algorithm selection for Predicting the best results. Usually Data Scientists use different kinds of Machine Learning algorithms to the large data sets. But, at high level all those different algorithms can be classified in two groups : supervised learning and unsupervised learning. A classification problem is when the output variable is a category like filtering emails “spam” or “not spam”

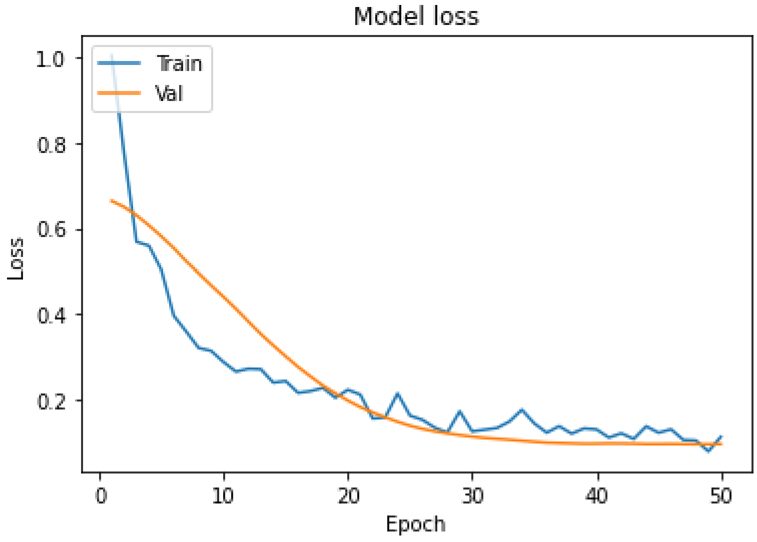
```
epochs = 50
model = Sequential()
model.add(Conv1D(filters=32, kernel_size=2, activation='relu', input_shape
= (30,1)))
model.add(BatchNormalization())
model.add(Dropout(0.2))

model.add(Conv1D(filters=64, kernel_size=2, activation='relu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Flatten())
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.5))

model.add(Dense(1, activation='sigmoid'))
```

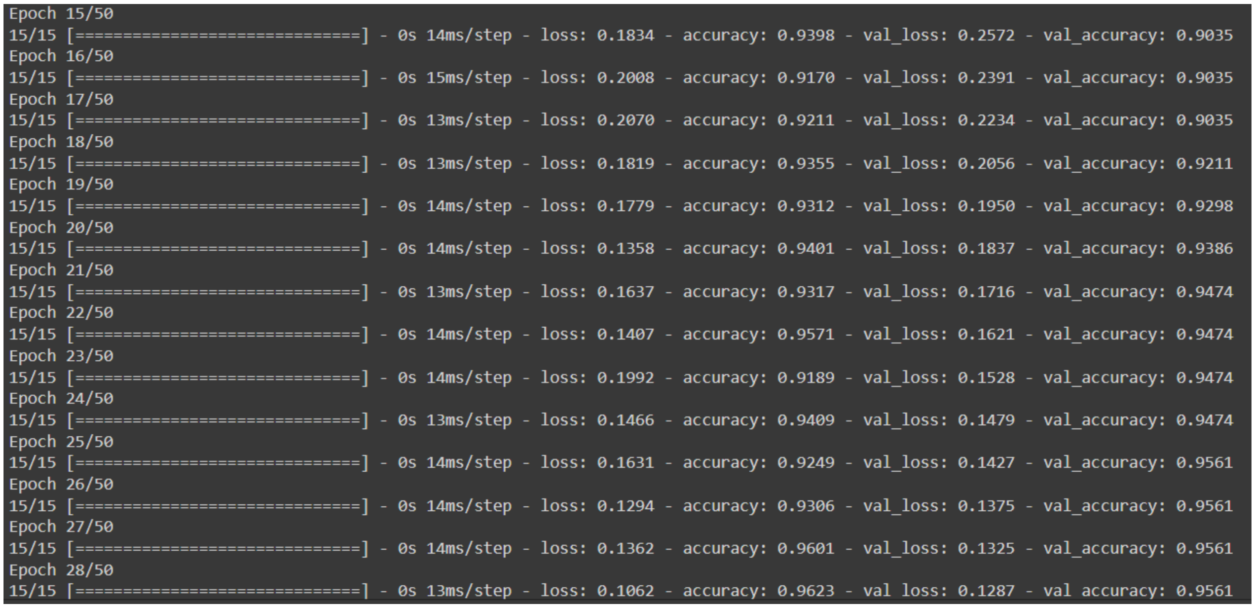
Unsupervised Learning :

```
history = model.fit(X_train, y_train, epochs=epochs,
validation_data=(X_test, y_test), verbose=1)
```

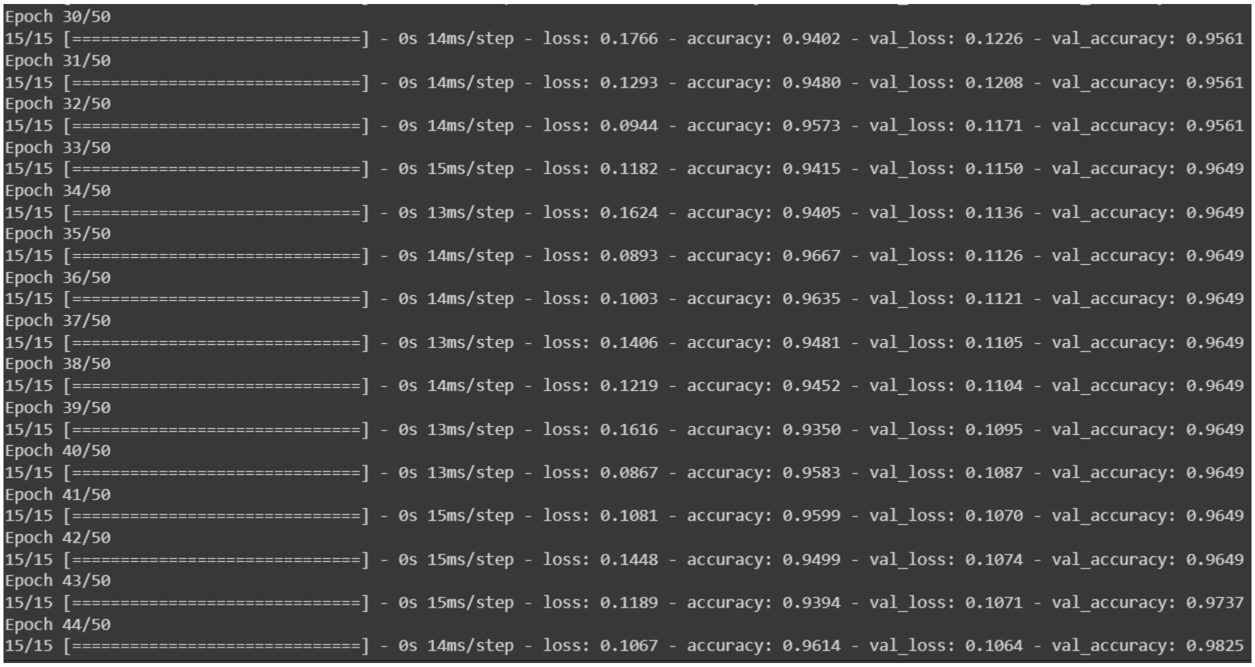
Results:







```
Epoch 15/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1834 - accuracy: 0.9398 - val_loss: 0.2572 - val_accuracy: 0.9035
Epoch 16/50
15/15 [==============================] - 0s 15ms/step - loss: 0.2008 - accuracy: 0.9170 - val_loss: 0.2391 - val_accuracy: 0.9035
Epoch 17/50
15/15 [==============================] - 0s 13ms/step - loss: 0.2070 - accuracy: 0.9211 - val_loss: 0.2234 - val_accuracy: 0.9035
Epoch 18/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1819 - accuracy: 0.9355 - val_loss: 0.2056 - val_accuracy: 0.9211
Epoch 19/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1779 - accuracy: 0.9312 - val_loss: 0.1950 - val_accuracy: 0.9298
Epoch 20/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1358 - accuracy: 0.9401 - val_loss: 0.1837 - val_accuracy: 0.9386
Epoch 21/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1637 - accuracy: 0.9317 - val_loss: 0.1716 - val_accuracy: 0.9474
Epoch 22/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1407 - accuracy: 0.9571 - val_loss: 0.1621 - val_accuracy: 0.9474
Epoch 23/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1992 - accuracy: 0.9189 - val_loss: 0.1528 - val_accuracy: 0.9474
Epoch 24/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1466 - accuracy: 0.9409 - val_loss: 0.1479 - val_accuracy: 0.9474
Epoch 25/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1631 - accuracy: 0.9249 - val_loss: 0.1427 - val_accuracy: 0.9561
Epoch 26/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1294 - accuracy: 0.9306 - val_loss: 0.1375 - val_accuracy: 0.9561
Epoch 27/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1362 - accuracy: 0.9601 - val_loss: 0.1325 - val_accuracy: 0.9561
Epoch 28/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1062 - accuracy: 0.9623 - val_loss: 0.1287 - val_accuracy: 0.9561
```



```
Epoch 30/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1766 - accuracy: 0.9402 - val_loss: 0.1226 - val_accuracy: 0.9561
Epoch 31/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1293 - accuracy: 0.9480 - val_loss: 0.1208 - val_accuracy: 0.9561
Epoch 32/50
15/15 [==============================] - 0s 14ms/step - loss: 0.0944 - accuracy: 0.9573 - val_loss: 0.1171 - val_accuracy: 0.9561
Epoch 33/50
15/15 [==============================] - 0s 15ms/step - loss: 0.1182 - accuracy: 0.9415 - val_loss: 0.1150 - val_accuracy: 0.9649
Epoch 34/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1624 - accuracy: 0.9405 - val_loss: 0.1136 - val_accuracy: 0.9649
Epoch 35/50
15/15 [==============================] - 0s 14ms/step - loss: 0.0893 - accuracy: 0.9667 - val_loss: 0.1126 - val_accuracy: 0.9649
Epoch 36/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1003 - accuracy: 0.9635 - val_loss: 0.1121 - val_accuracy: 0.9649
Epoch 37/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1406 - accuracy: 0.9481 - val_loss: 0.1105 - val_accuracy: 0.9649
Epoch 38/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1219 - accuracy: 0.9452 - val_loss: 0.1104 - val_accuracy: 0.9649
Epoch 39/50
15/15 [==============================] - 0s 13ms/step - loss: 0.1616 - accuracy: 0.9350 - val_loss: 0.1095 - val_accuracy: 0.9649
Epoch 40/50
15/15 [==============================] - 0s 13ms/step - loss: 0.0867 - accuracy: 0.9583 - val_loss: 0.1087 - val_accuracy: 0.9649
Epoch 41/50
15/15 [==============================] - 0s 15ms/step - loss: 0.1081 - accuracy: 0.9599 - val_loss: 0.1070 - val_accuracy: 0.9649
Epoch 42/50
15/15 [==============================] - 0s 15ms/step - loss: 0.1448 - accuracy: 0.9499 - val_loss: 0.1074 - val_accuracy: 0.9649
Epoch 43/50
15/15 [==============================] - 0s 15ms/step - loss: 0.1189 - accuracy: 0.9394 - val_loss: 0.1071 - val_accuracy: 0.9737
Epoch 44/50
15/15 [==============================] - 0s 14ms/step - loss: 0.1067 - accuracy: 0.9614 - val_loss: 0.1064 - val_accuracy: 0.9825
```

DATA SET:

We used the UCI machine learning Repository as our data set. This dataset contains a total of 569 instances which is basically a digitised image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. Each of the instances are divided into two classes based on 30 numeric and predictive attributes.

Attribute information:

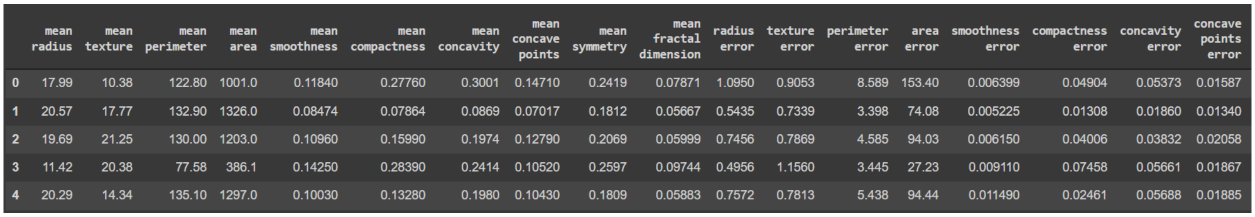
- Radius (mean of distances from centre to points on the perimeter)
- Texture (standard deviation of grey-scale values)
- Perimeter
- Area
- Smoothness (local variation in radius lengths)
- Compactness (perimeter^2 / area - 1.0)
- Concavity (severity of concave portions of the contour)
- Concave Points (number of concave portions of the contour)
- Symmetry
- Fractal Dimension ("coastline approximation" - 1)

The mean, standard error, and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features.

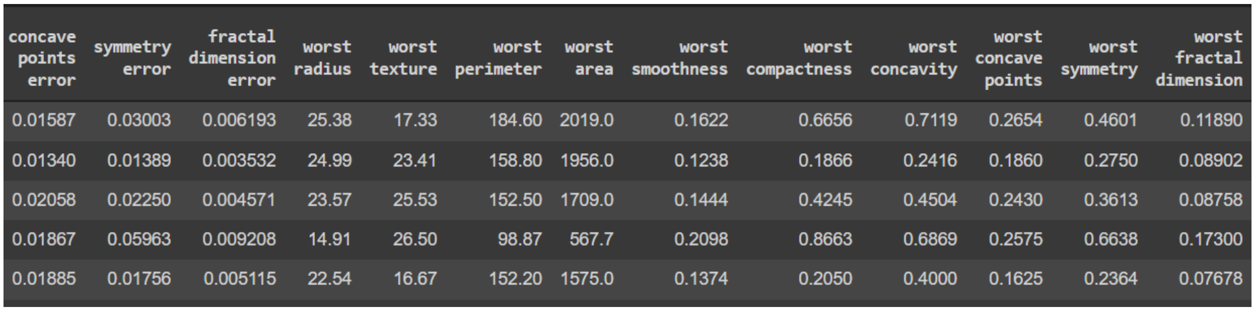
We have two classes:

- WDBC Malignant
- WDBC Benign

Malignant and benign are two types of breast cancer, malignant being dangerous/cancerous i.e. immediate attention should be brought to it whereas benign are not cancerous i.e. just a breast lump. Our main aim is to detect and classify tumors into these two classes.



	mean radius	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
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871	1.0950	0.9053	8.589	153.40	0.006399	0.04904	0.05373	0.01587
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667	0.5435	0.7339	3.398	74.08	0.005225	0.01308	0.01860	0.01340
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999	0.7456	0.7869	4.585	94.03	0.006150	0.04006	0.03832	0.02058
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744	0.4956	1.1560	3.445	27.23	0.009110	0.07458	0.05661	0.01867
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883	0.7572	0.7813	5.438	94.44	0.011490	0.02461	0.05688	0.01885



concave points error	symmetry error	fractal dimension error	worst radius	worst texture	worst perimeter	worst area	worst smoothness	worst compactness	worst concavity	worst concave points	worst symmetry	worst fractal dimension
0.01587	0.03003	0.006193	25.38	17.33	184.60	2019.0	0.1622	0.6656	0.7119	0.2654	0.4601	0.11890
0.01340	0.01389	0.003532	24.99	23.41	158.80	1956.0	0.1238	0.1866	0.2416	0.1860	0.2750	0.08902
0.02058	0.02250	0.004571	23.57	25.53	152.50	1709.0	0.1444	0.4245	0.4504	0.2430	0.3613	0.08758
0.01867	0.05963	0.009208	14.91	26.50	98.87	567.7	0.2098	0.8663	0.6869	0.2575	0.6638	0.17300
0.01885	0.01756	0.005115	22.54	16.67	152.20	1575.0	0.1374	0.2050	0.4000	0.1625	0.2364	0.07678

CONCLUSION:

Using our efficient program (with CNN and Keras) we can find out if a person has a tumour or not in an effective manner. Cancer, now being a “Secondary” disease, (first, now being Covid-19) can now be noticed by many AI softwares by using this program. The main objective of our program being reducing the workload on doctors will soon be achieved. There are several instances of technical faults in these types of models, but we have attempted to reduce them to a minimum so that our consumers have a positive experience.

REFERENCES:

1. Sathyakumar K, Munoz M, Singh J, et al. (August 25, 2020) Automated Lung Cancer Detection Using Artificial Intelligence (AI) Deep Convolutional Neural Networks: A Narrative Literature Review. Cureus 12(8): e10017. doi:10.7759/cureus.10017
2. D. Voth, "Using AI to detect breast cancer," in *IEEE Intelligent Systems*, vol. 20, no. 1, pp. 5-7, Jan.-Feb. 2005, doi: 10.1109/MIS.2005.14.
3. Nehmat Houssami, Georgia Kirkpatrick-Jones, Naomi Noguchi & Christoph I. Lee (2019) Artificial Intelligence (AI) for the early detection of breast cancer: a scoping review to assess AI’s potential in breast screening practice, Expert Review of Medical Devices, 16:5, 351-362, DOI: 10.1080/17434440.2019.1610387
4. José Luis Raya-Povedano , Sara Romero-Martín, Esperanza Elías-Cabot, Albert Gubern-Mérida, Alejandro Rodríguez-Ruiz, Marina Álvarez-Benito (May 4, 2021) AI-based Strategies to Reduce Workload in Breast Cancer Screening with Mammography and Tomosynthesis: A Retrospective Evaluation, https://doi.org/10.1148/radiol.2021203555
5. Kohlberger, T., Liu, Y., Moran, M., Chen, P. C., Brown, T., Hipp, J. D., Mermel, C. H., & Stumpe, M. C. (2019). Whole-Slide Image Focus Quality: Automatic Assessment and Impact on AI Cancer Detection. *Journal of pathology informatics*, *10*, 39. https://doi.org/10.4103/jpi.jpi_11_19
6. Alexander J. T. Wanders , Willem Mees, Petra A.M. Bun, Natasja Janssen, Alejandro Rodríguez-Ruiz, Mehmet Ufuk Dalmış, Nico Karssemeijer, Carla H.

van Gils, Ioannis Sechopoulos, Ritse M. Mann, Cornelis Jan van Rooden (February 8, 2022) https://doi.org/10.1148/radiol.210832

7. Merali, Z., Wang, J.Z., Badhiwala, J.H. *et al.* A deep learning model for detection of cervical spinal cord compression in MRI scans. *Sci Rep* 11, 10473 (2021). https://doi.org/10.1038/s41598-021-89848-3
8. Ranjbarzadeh, R., Bagherian Kasgari, A., Jafarzadeh Ghoushchi, S. *et al.* Brain tumor segmentation based on deep learning and an attention mechanism using MRI multi-modalities brain images. *Sci Rep* 11, 10930 (2021). https://doi.org/10.1038/s41598-021-90428-8
9. Hassan Ali Khan , Wu Jue, Muhammad Mushtaq and Muhammad Umer Mushtaq *et al.* Brain tumor classification in MRI image using convolutional neural network 2020, Volume 17, Issue 5: 6203-6216. doi: 10.3934/mbe.2020328
10. Noreen, Neelum & Palaniappan, Sellapan & Qayyum, Abdul & Ahmad, Iftikhar & Imran, Muhammad & Shoaib, Muhammad. (2020). A Deep Learning Model Based on Concatenation Approach for the Diagnosis of Brain Tumor. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.2978629.
11. Watanabe, A.T., Lim, V., Vu, H.X. *et al.* Improved Cancer Detection Using Artificial Intelligence: a Retrospective Evaluation of Missed Cancers on Mammography. *J Digit Imaging* 32, 625–637 (2019). https://doi.org/10.1007/s10278-019-00192-5
12. Ke Wang et al 2020 IOP Conf. Ser.: Earth Environ. Sci. 474 032030
13. Amin, J., Sharif, M., Haldorai, A. *et al.* Brain tumor detection and classification using machine learning: a comprehensive survey. *Complex Intell. Syst.* (2021). https://doi.org/10.1007/s40747-021-00563-y
14. Irmak, E. Multi-Classification of Brain Tumor MRI Images Using Deep Convolutional Neural Network with Fully Optimized Framework. *Iran J Sci Technol Trans Electr Eng* 45, 1015–1036 (2021). https://doi.org/10.1007/s40998-021-00426-9
15. Nadeem, M. W., Ghamdi, M., Hussain, M., Khan, M. A., Khan, K. M., Almotiri, S. H., & Butt, S. A. (2020). Brain Tumor Analysis Empowered with Deep

Learning: A Review, Taxonomy, and Future Challenges. *Brain sciences*, *10*(2), 118. https://doi.org/10.3390/brainsci10020118