**Analysis Report of the Data:**

Deep learning has paved its roots into our society these days because there are a lot of lifestyle changes in our society such as self driving cars, google assistance, spam email detection and much more. Deep machine learning has also evolved in healthcare where we use these models to detect whether a person is suffering from a disease or not. We also train, test, and validate these models to detect whether the patients have a brain tumor or not.

In healthcare a magnetic resonance imaging (MRI) is used to detect whether a person has a tumor or not and which category does it fit into. To solve this issue, we have been using deep learning models because by providing MRI imaging to the model we can detect whether the patient has a tumor or not or which kind of tumor. A brain tumor classifier was developed using CNN imaging widely used for image classification these days. The data set that was used would detect whether a person has a tumor or not. There are two categories in the dataset (yes)or(no).

**Model Building:**

* 1.First the necessary libraries were imported, and the CNN model is derived from TensorFlow on the core side and implemented by using karas. The images were imported using pre-processing using OpenCV.
* 2. Images are augmented using Gaussian filter applied to them and then converted into Grayscale and then into Gaussian blur filter this process includes augmentation, dilation, image erosion and image thresholding. Then blur remover is used to remove blur. Image speckle noise, high dimensional feature analysis using CNN.
* 3. There was also variable learning rate applied to avoid overfitting of the model. Tensor board log files are also stored event monetization is done and further in the model optimizer Adam is used, loss binary cross entropy loss, metric use accuracy also plots training loss and validation loss and training accuracy, Validation history was derived from model history and for parameter analysis we used F1 accuracy score precision recall.

**Analysis of the Model:**

The performance of the model was 57% after running the model which was not something we were looking for so to further improve the performance neurons and activation function was used and the accuracy reached to 65%.

**CNN Model:**

I used CNN because the time taken for tuning these parameters diminishes by using CNNs. CNNs are fully connected feed forward neural networks. CNNs are very effective in reducing the number of parameters without losing on the quality of models. Images have high dimensionality (as each pixel is considered as a feature) which suits the above-described abilities of CNNs.