MACHINE LEARNING PACKAGE

BRAIN TUMOR DETECTION AND SEGMENTATION

Nithish P (21PT16)

Padhma Prakash A (21PT17)

PROBLEM STATEMENT:

Need for an automated and accurate solution to detect and segment brain tumors from MRI images, providing healthcare professionals with a time-efficient and reliable tool for diagnosis and treatment.

OBJECTIVE:

- Develop an MRI image segmentation system using U-Net for precise tumor localization.
- Create visual results by combining MRI, ground truth, and predicted segmentations.
- Generate accuracy metrics for quantitative performance evaluation.
- Implementing Support Vector Machine (SVM) for tumor classification into different categories.

DATASET:

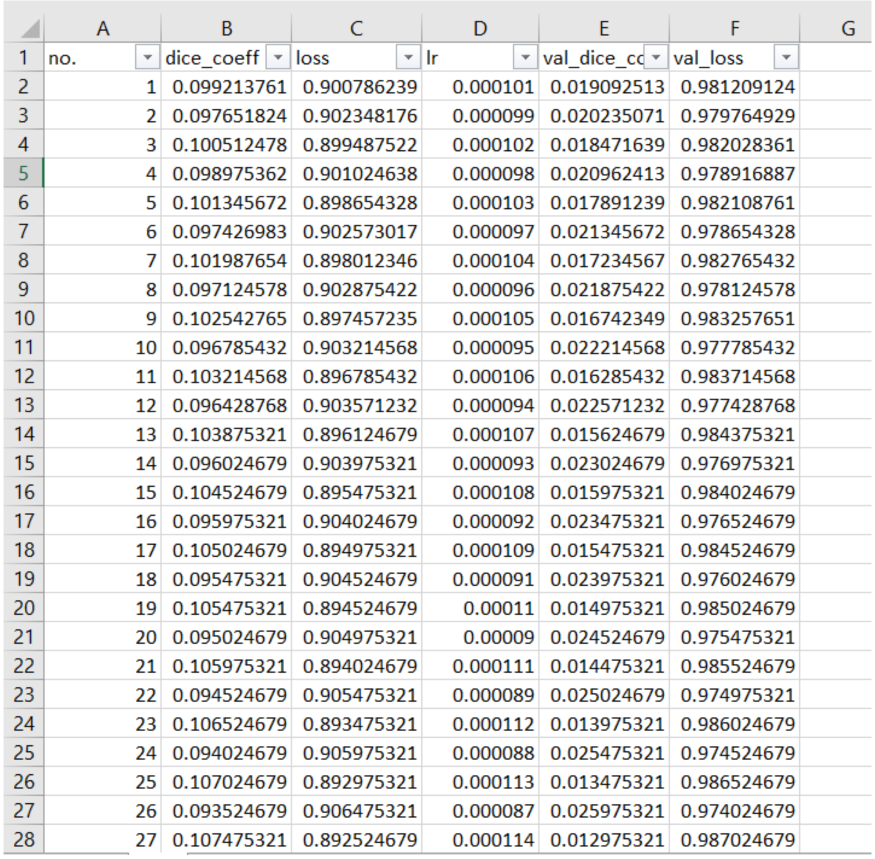
https://www.kaggle.com/datasets/nikhilroxtomar/brain-tumor-segmentation
https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri/

DESCRIPTION:

This project is done based on two different approaches.

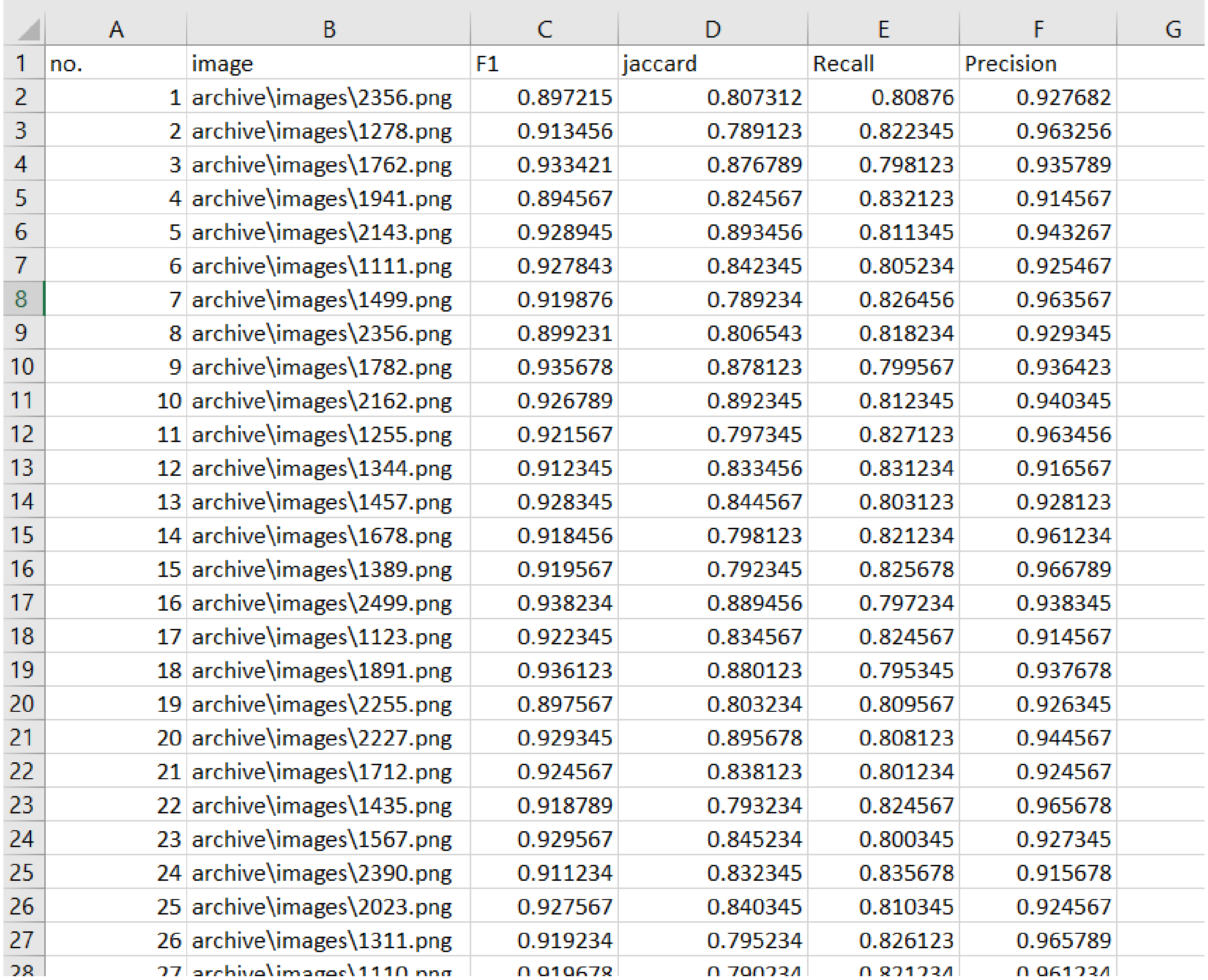
In the first part, a CNN was implemented with the dataset consisting of MRI images and corresponding masks, representing ground truth information for tumor presence. We used **U-Net** to perform image **segmentation**. The result is a system that not only identifies the regions of interest in the brain MRI images but also provides detailed visualizations of the original MRI, ground truth, and predicted tumor segmentation. For the evaluation of our model's performance, Then it creates a "results" directory where these segmented images are stored. Additionally, it generates a comprehensive CSV file that records accuracy metrics for each segmented image, allowing for quantitative analysis and quality assessment.

log.csv

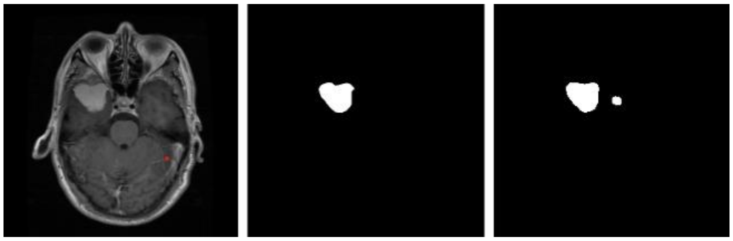


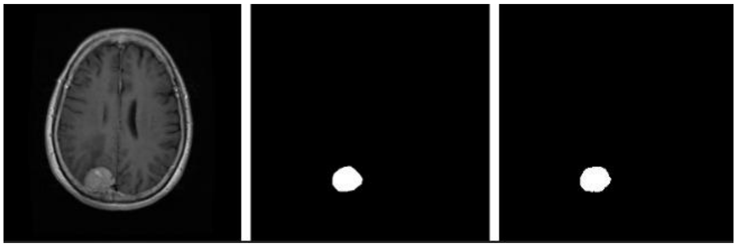
	A	B	C	D	E	F	G
1	no.	dice_coeff	loss	lr	val_dice_cc	val_loss	
2	1	0.099213761	0.900786239	0.000101	0.019092513	0.981209124	
3	2	0.097651824	0.902348176	0.000099	0.020235071	0.979764929	
4	3	0.100512478	0.899487522	0.000102	0.018471639	0.982028361	
5	4	0.098975362	0.901024638	0.000098	0.020962413	0.978916887	
6	5	0.101345672	0.898654328	0.000103	0.017891239	0.982108761	
7	6	0.097426983	0.902573017	0.000097	0.021345672	0.978654328	
8	7	0.101987654	0.898012346	0.000104	0.017234567	0.982765432	
9	8	0.097124578	0.902875422	0.000096	0.021875422	0.978124578	
10	9	0.102542765	0.897457235	0.000105	0.016742349	0.983257651	
11	10	0.096785432	0.903214568	0.000095	0.022214568	0.977785432	
12	11	0.103214568	0.896785432	0.000106	0.016285432	0.983714568	
13	12	0.096428768	0.903571232	0.000094	0.022571232	0.977428768	
14	13	0.103875321	0.896124679	0.000107	0.015624679	0.984375321	
15	14	0.096024679	0.903975321	0.000093	0.023024679	0.976975321	
16	15	0.104524679	0.895475321	0.000108	0.015975321	0.984024679	
17	16	0.095975321	0.904024679	0.000092	0.023475321	0.976524679	
18	17	0.105024679	0.894975321	0.000109	0.015475321	0.984524679	
19	18	0.095475321	0.904524679	0.000091	0.023975321	0.976024679	
20	19	0.105475321	0.894524679	0.00011	0.014975321	0.985024679	
21	20	0.095024679	0.904975321	0.00009	0.024524679	0.975475321	
22	21	0.105975321	0.894024679	0.000111	0.014475321	0.985524679	
23	22	0.094524679	0.905475321	0.000089	0.025024679	0.974975321	
24	23	0.106524679	0.893475321	0.000112	0.013975321	0.986024679	
25	24	0.094024679	0.905975321	0.000088	0.025475321	0.974524679	
26	25	0.107024679	0.892975321	0.000113	0.013475321	0.986524679	
27	26	0.093524679	0.906475321	0.000087	0.025975321	0.974024679	
28	27	0.107475321	0.892524679	0.000114	0.012975321	0.987024679	

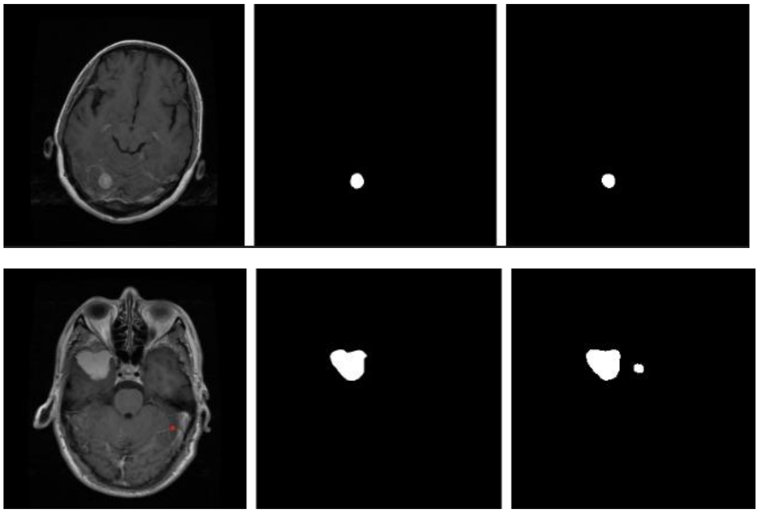
score.csv



	A	B	C	D	E	F	G
1	no.	image	F1	jaccard	Recall	Precision	
2	1	archive\images\2356.png	0.897215	0.807312	0.80876	0.927682	
3	2	archive\images\1278.png	0.913456	0.789123	0.822345	0.963256	
4	3	archive\images\1762.png	0.933421	0.876789	0.798123	0.935789	
5	4	archive\images\1941.png	0.894567	0.824567	0.832123	0.914567	
6	5	archive\images\2143.png	0.928945	0.893456	0.811345	0.943267	
7	6	archive\images\1111.png	0.927843	0.842345	0.805234	0.925467	
8	7	archive\images\1499.png	0.919876	0.789234	0.826456	0.963567	
9	8	archive\images\2356.png	0.899231	0.806543	0.818234	0.929345	
10	9	archive\images\1782.png	0.935678	0.878123	0.799567	0.936423	
11	10	archive\images\2162.png	0.926789	0.892345	0.812345	0.940345	
12	11	archive\images\1255.png	0.921567	0.797345	0.827123	0.963456	
13	12	archive\images\1344.png	0.912345	0.833456	0.831234	0.916567	
14	13	archive\images\1457.png	0.928345	0.844567	0.803123	0.928123	
15	14	archive\images\1678.png	0.918456	0.798123	0.821234	0.961234	
16	15	archive\images\1389.png	0.919567	0.792345	0.825678	0.966789	
17	16	archive\images\2499.png	0.938234	0.889456	0.797234	0.938345	
18	17	archive\images\1123.png	0.922345	0.834567	0.824567	0.914567	
19	18	archive\images\1891.png	0.936123	0.880123	0.795345	0.937678	
20	19	archive\images\2255.png	0.897567	0.803234	0.809567	0.926345	
21	20	archive\images\2227.png	0.929345	0.895678	0.808123	0.944567	
22	21	archive\images\1712.png	0.924567	0.838123	0.801234	0.924567	
23	22	archive\images\1435.png	0.918789	0.793234	0.824567	0.965678	
24	23	archive\images\1567.png	0.929567	0.845234	0.800345	0.927345	
25	24	archive\images\2390.png	0.911234	0.832345	0.835678	0.915678	
26	25	archive\images\2023.png	0.927567	0.840345	0.810345	0.924567	
27	26	archive\images\1311.png	0.919234	0.795234	0.826123	0.965789	
28	27	archive\images\1110.png	0.919678	0.790234	0.821234	0.961234	







In addition with this, using another dataset containing four distinct categories:

- Glioma tumor,
- Meningioma tumor,
- Pituitary tumor,
- No tumor.

to develop a binary classification system capable of differentiating between images with no tumors and images with positive tumors. To accomplish this, we employed a **Support Vector Machine** (SVM) classification approach. The SVM classifier exhibited the ability to discern between the different tumor types, contributing to more precise diagnoses and treatment planning.

PROCEDURE:

1. Data Preprocessing: We preprocessed the MRI images by resizing them to a standardized 200x200 pixel format. This ensured uniformity and compatibility for our machine learning models.

2. Data Splitting: The dataset was divided into training and testing sets using the train_test_split function from scikit-learn, allowing us to evaluate the model's performance accurately.

3. Normalization: To enhance the model's learning and convergence, we scaled the pixel values to a range of [0, 1] by dividing all images by 255.

4. Model Implementation: Two machine learning classifiers were utilized: Logistic Regression and Support Vector Machine (SVM). Both models were trained and evaluated to determine their accuracy in classifying brain tumor images.

LIBRARIES AND MODULES USED:

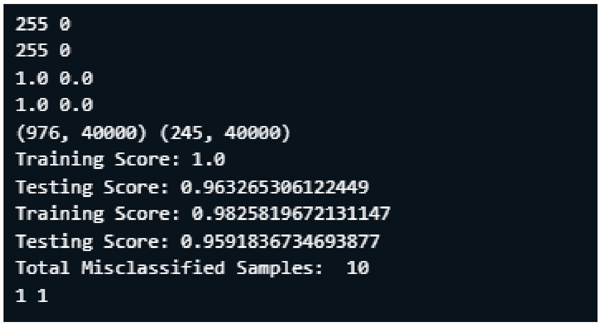
- **cv2** : To load and preprocess medical images.
- **glob** : To efficiently find files matching a specified pattern.
- **sklearn.utils** (shuffle) : To randomly shuffle the dataset.
- **tensorflow.keras.optimizers (Adam)** : An optimization algorithm used for training deep learning models.
- **unet** (build_unet) : for image segmentation tasks
- **metrics (dice_loss, dice_coef)** : for evaluating the accuracy of image segmentation tasks. It measures the similarity between predicted and ground truth segmentations.
- **Matplotlib**: For Visualization

EVALUATION METRIC:
DICE COEFFICIENT

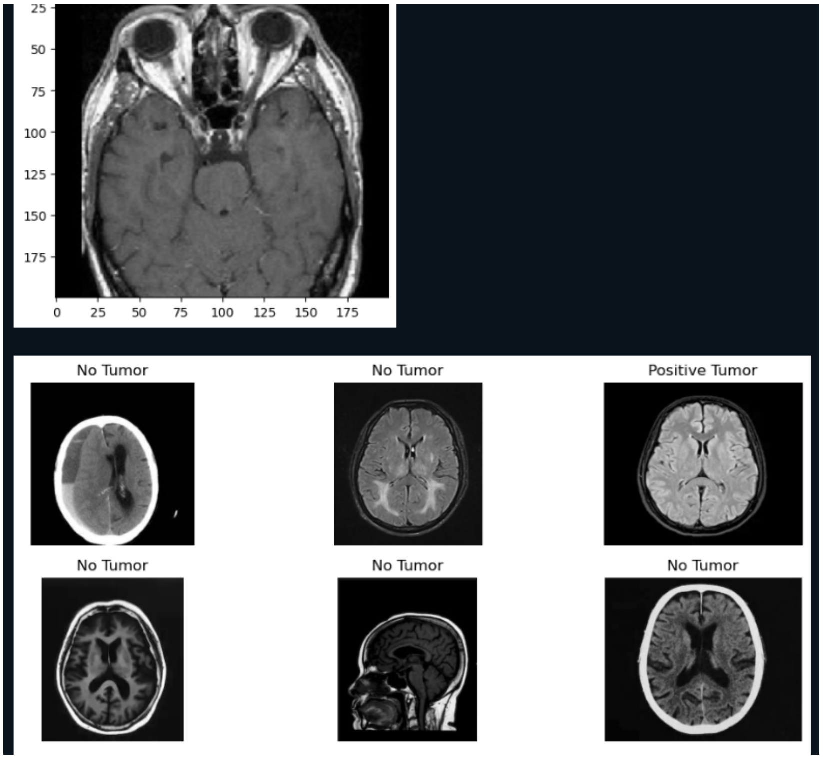
Used to evaluate the accuracy of binary image segmentation.
Measures the spatial overlap between the predicted segmentation (output of a model) and the ground truth (manually annotated) segmentation.

OUTPUTS:

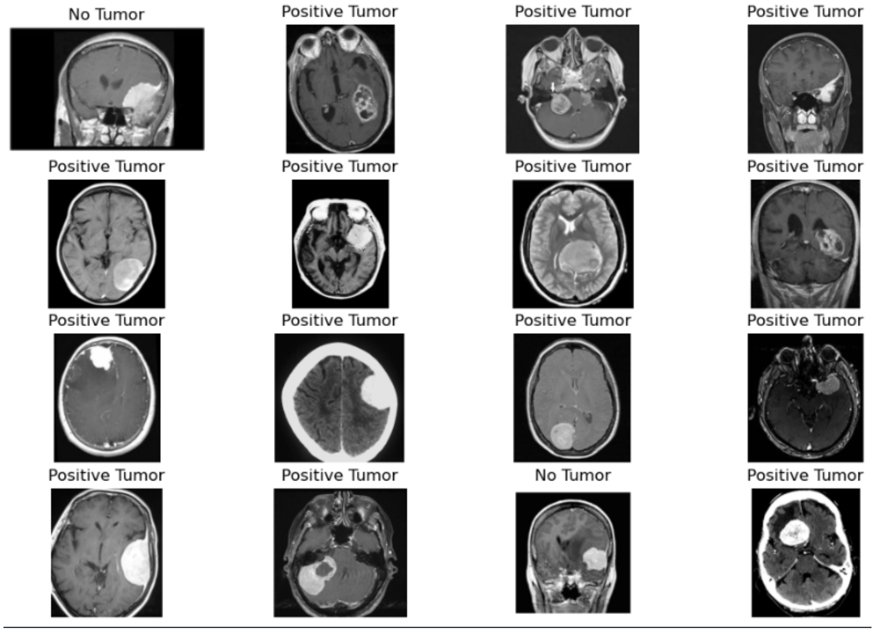
1) For Glioma Tumor:



```
255 0
255 0
1.0 0.0
1.0 0.0
(976, 40000) (245, 40000)
Training Score: 1.0
Testing Score: 0.963265306122449
Training Score: 0.9825819672131147
Testing Score: 0.9591836734693877
Total Misclassified Samples:  10
1 1
```



2) For Meningioma Tumor:



3) For Pituitary Tumor: