

# Brain Tumor Classification

Josue Antonio and Olakunle Abawonse

## Introduction

This project aims to develop an accurate and efficient method for classifying brain tumors from MRI scans using features obtained through topological methods. Traditional methods of diagnosing brain tumors can be unreliable and time-consuming, making it crucial to explore the potential of topological methods. We obtained our dataset from Kaggle, which includes 4 labeled folders of MRI scans representing glioma, meningioma, pituitary tumors, and a control group with no tumors. The dataset consists of 2 sets of scans, one for training and one for testing. The training set includes 4 folders with a total of 2727 MRI scans, while the testing set includes 4 folders with a total of 394 MRI scans. Our dataset is significant because it provides a diverse range of brain tumors, which is important for training and testing our model's accuracy.

To analyze our image data, we'll apply the sublevel set filtration method. This will allow us to identify the significant features in each MRI scan image. Next, we'll compute the mean persistence landscapes, which will summarize the persistence information across different sublevel sets. We'll then use these mean persistence landscapes as our training features for a deep neural network for which we'll employ the MLPClassifier from scikit learn.

## Methods and Results

### 1.) Forming a simplicial complex from an image ( [3], [7] )

Given a two-dimensional image  $I$ , one starts with the empty complex  $K$ . We then assign a vertex to each pixel in  $I$  and add these vertices to  $K$ . We form 1-simplices from these vertices if the associated pixels are adjacent in  $I$  (diagonal pixels are treated as adjacent). Similarly, we form 2-simplices if three vertices are mutually adjacent.

### 2.) Filtration on $K$ ( [3] )

Let  $V$  be set of vertices of  $K$  and  $f : V \rightarrow \mathbb{R}$  be pixel intensity. Let  $f_{min}, f_{max}$  denote the minimum and maximum values of  $f$ . Let  $i_1 < i_2 < i_3 \dots$  be an increasing sequence of positive real numbers.

Let  $K_i = \{S \in K : \forall v \in S, f(v) \leq i\}$ . Then the sequence of subcomplexes  $K_{f_{min}} \subseteq K_{f_{min}+i_1} \subseteq K_{f_{min}+i_2} \cdots \subseteq K_{f_{max}} = K$  is a filtration of  $K$ .

**3.)** For each MRI image, we will divide the image into horizontal and vertical slices of about equal intervals of 20. We will apply the above filtration to our horizontal and vertical slices. We will then compute the corresponding persistent diagrams in 0 and 1 dimensional homologies. For the purpose of this experiment we will only train our deep neural network with features from the 1-dim homology.

Code for generating Persistent diagrams

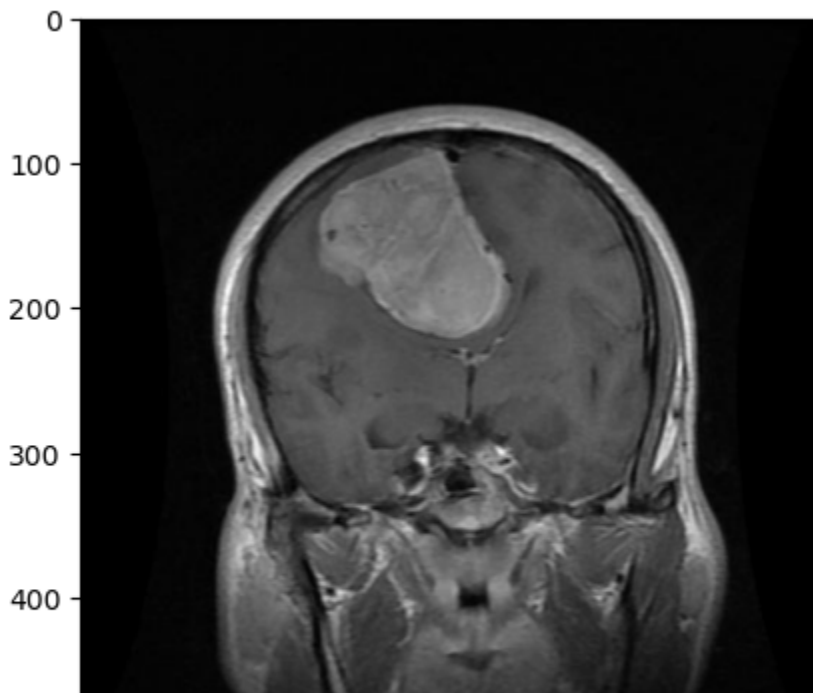
```
dgm1 = lower_star_img(img[160:180,:])
plt.figure(figsize=(6, 6))
plot_diagrams(dgm1, lifetime=True, title = 'Degree 1 Persistence Diagram of Horizontal Slice')
plt.show()
```

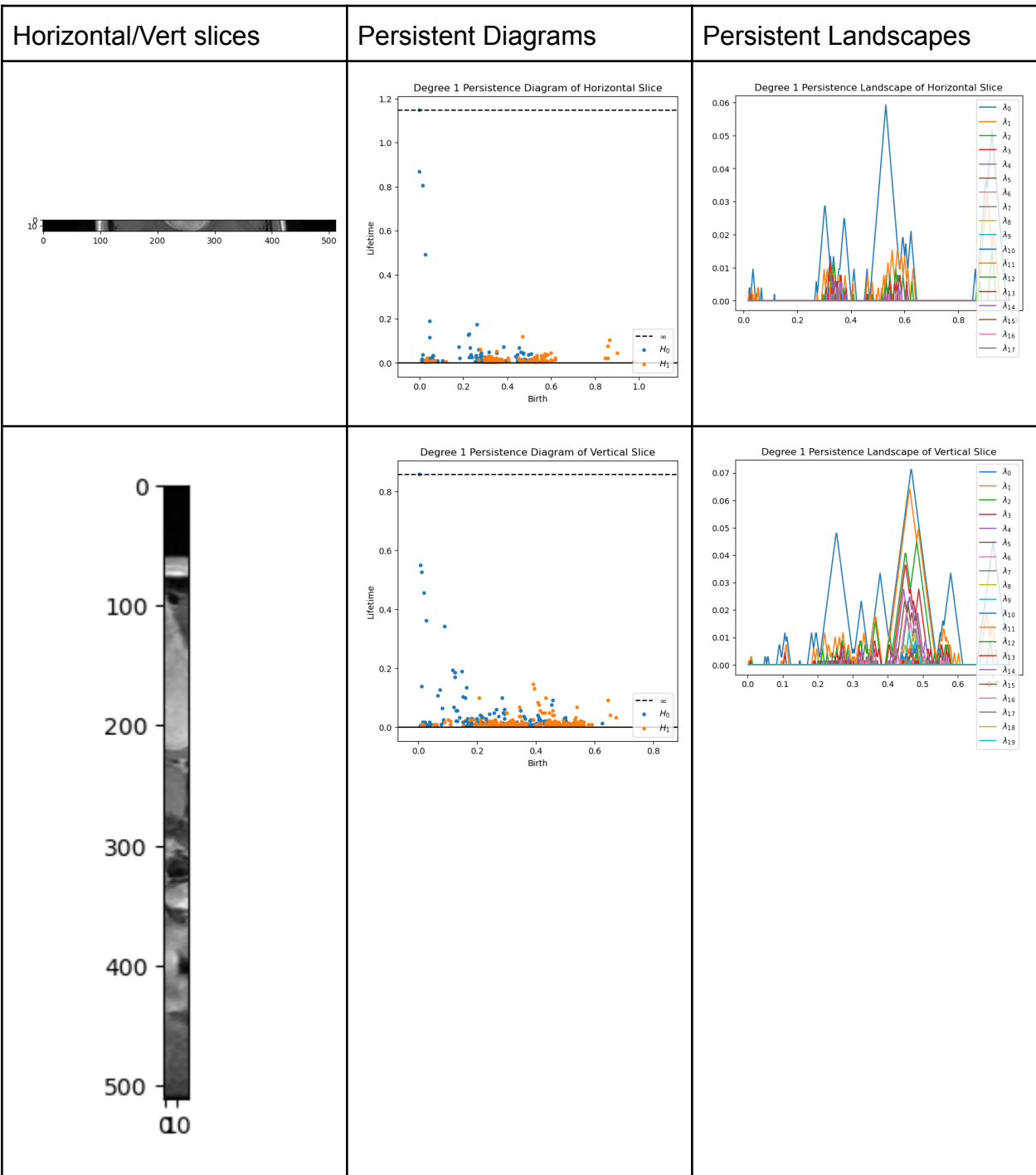
Code for generating Persistent landscapes

```
import persim
hor_slice1_ls = PersLandscapeApprox(dgms = dgm1, hom_deg = 1)
plot_landscape_simple(hor_slice1_ls, title='Degree 1 Persistence Landscape of Horizontal Slice')
```

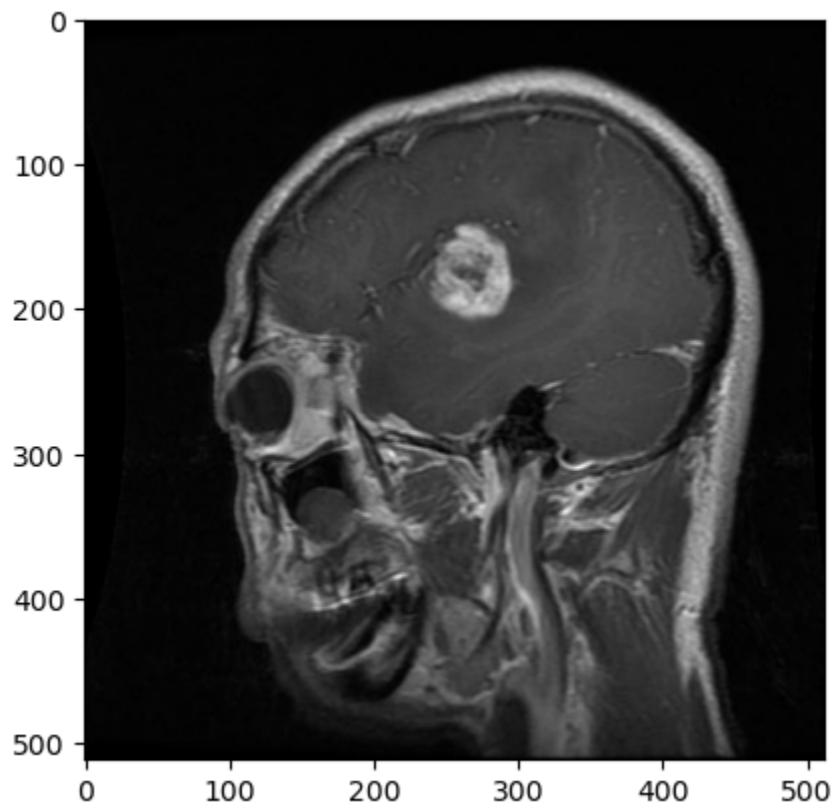
We obtained the following results:

### **Meningioma tumor**

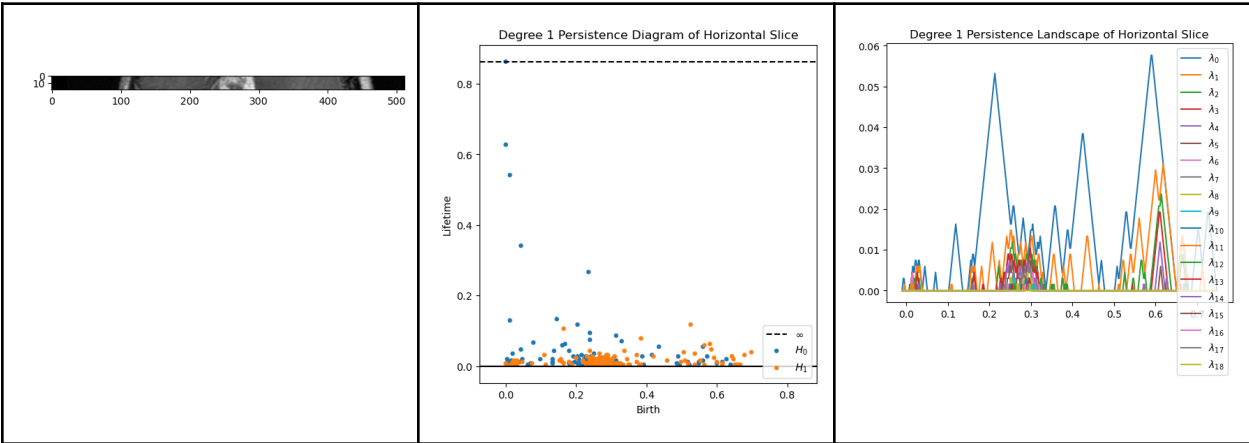


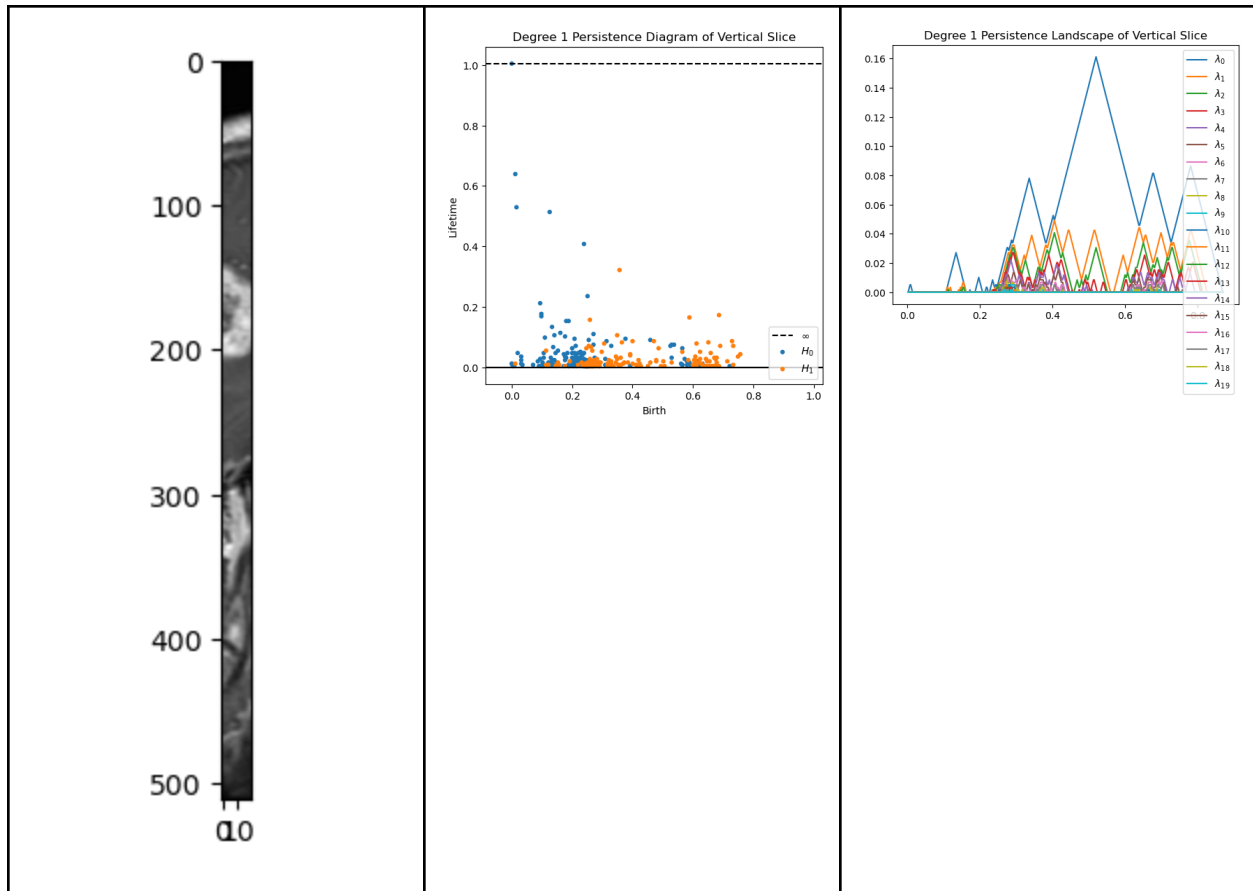


Glioma tumor



Horizontal/Vertical slices	Persistent Diagrams	Persistent Landscapes
----------------------------	---------------------	-----------------------

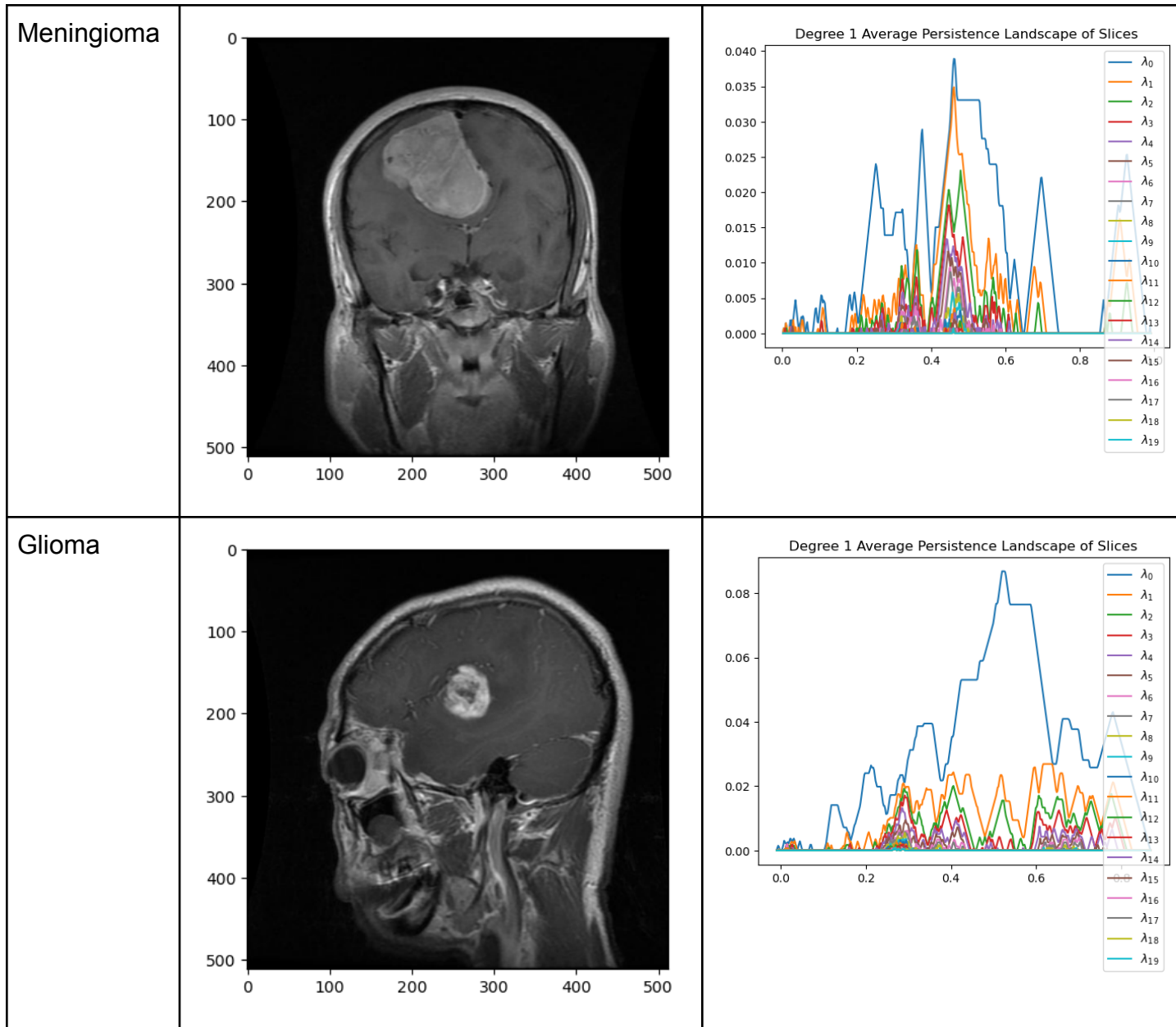




**4)** Each MRI image is divided into about 60 to 80 horizontal and vertical slices depending on the size of the image. So we computed the mean persistence landscape for each MRI image.

Code for generating mean persistent landscapes

```
landscapes = []
landscapes.append(hor_slice1_ls)
landscapes.append(ver_slice1_ls)
avg_ls = average_approx(landscapes)
plot_landscape_simple(avg_ls,title='Degree 1 Average Persistence Landscape of Slices')
```



## 5.) Training with a deep neural network (MLPClassifier)

We experimented with only 600 training images, consisting of 150 images from each folder. Our testing data is of size 120, consisting of 30 images from each folder.

The mean persistence landscape associated with each image is used as our training input.

Code for training a deep neural network

```
sc_X = StandardScaler()
X_trainscaled = sc_X.fit_transform(new_data)
X_testscaled = sc_X.fit_transform(test_data)

clf = MLPClassifier(hidden_layer_sizes=(256,256,256,256,256,128,128,128,128,64),
learning_rate = 'adaptive', random_state=1).fit(X_trainscaled, new_labels)
y_test = test_data_labels
y_pred=clf.predict(X_testscaled)
print(clf.score(X_testscaled, test_data_labels))
```

The link to our GitHub repository is:

<https://github.com/soabawonse/Topological-methods-for-brain-tumor-classification>

The best result we got so far is an accuracy of 76% with the hidden layer sizes (5 x 256, 4 x 128, 2 x 64) and a constant learning rate of 0.001.

**6.) Possible further work:** We are yet to find a suitable deep neural network to train our features on. We experimented on a number of occasions on various hidden\_layer\_sizes to get the best accuracy of 76%. We also hope to be able to train with more data and with a deeper neural network for an overall better accuracy.

## References

1.) Bubenik, P. Statistical Topological Data Analysis using Persistence Landscapes. <https://arxiv.org/abs/1207.6437>, 2015.

2.) Bubenik, P., Dlotko, P. A persistence landscapes toolbox for topological statistics. <https://arxiv.org/abs/1501.00179>, 2015.

3.) A. Adcock, D. Rubin, G. Carlsson Classification of hepatic lesions using the matching metric

4.) A. Adcock, E. Carlsson, G. Carlsson The ring of algebraic functions on persistence barcodes

5.) <https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri>.

6.)<https://www.kaggle.com/code/jaykumar1607/brain-tumor-mri-classification-tensorflow-cnn>.

7.)<https://ripser.scikit-tda.org/en/latest/notebooks/Lower%20Star%20Image%20Filtrations.html>