- **GitHub Repository:** https://github.com/stjoha81/Capstone_2023_2024/tree/main
- **Summary of Research papers and articles:**

  - **2017: Classification of Brain MRI Tumor Images: A Hybrid Approach**
    - https://www.researchgate.net/publication/321750158_Classification_of_Brain_MRI_Tumor_Images_A_Hybrid_Approach
    - This paper draws on previous research and applies a hybrid approach for determining whether a brain tumor is benign or malignant. The researchers used 25 MRI scans and applied preprocessing steps (discrete wavelet transformation and principal component analysis. A genetic algorithm was applied for feature extraction from the image data, and then a support vector machine was trained and used to classify the data. Accuracy was achieved between 80%-90%.

  - **2019: Multi-Classification of Brain Tumor Images Using Deep Neural Network.**
    - https://ieeexplore.ieee.org/abstract/document/8723045
    - The researchers made use of two datasets consisting of a total of 3,580 images, with the goal of distinguishing between three types of tumors (meningioma, glioma, and pituitary) and further classifying glioma tumors into one of three grades based on WHO standards. Preprocessing consisted of downsizing the images to improve processing time as well as augmentation (flipping, mirroring, adding noise), resulting in just over 15,000 images being used. A convolutional neural network with 16 layers was built. The CNN was built using a 3x3 convolution kernel and ReLU for an activation function. Softmax function layer was applied prior to classification. The model achieved 96% accuracy for the initial goal of tumor classification, and 98% accuracy for the secondary goal of classifying glioma grades.

  - **2022: Accurate Brain Tumor Detection Using Deep Convolutional Neural Network**
    - https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9468505/
    - Code: https://github.com/saikat15010/Brain-Tumor-Detection
    - Code reproduction: https://github.com/stjoha81/Capstone_2023_2024/blob/78c07ff152a444a72890fec0b260323e1d4a4f5f/Reproduction%20of%202022%20Accurate%20Brain%20Tumor%20Detection%20Using%20Deep%20Convolutional%20Neural%20Network.ipynb
    - The researchers used 3,064 MRI brain scan images with a fine tuned CNN based on VGG16. The CNN has a total of 23 layers. It adds 4 Dense layers, one GlobalAveragePooling2D layer, one Dropout layer, and a Softmax layer for fine tuning the VGG16 model. The convolutional layers employed different sized kernels and used ReLU activation. This

approach yielded accuracy as follows: 96.7% meningioma, 97.2 glioma, 99.5 pituitary. The average accuracy was 97.8%.

- **2022: Image Segmentation with U-Net**
  - https://www.analyticsvidhya.com/blog/2022/10/image-segmentation-with-u-net/
  - Code: https://www.analyticsvidhya.com/blog/2022/10/image-segmentation-with-u-net/
  - U-Net was developed in 2015 for work on biomedical images. Its use has since spread to other areas of computer vision, including autonomous driving and satellite imaging. The U-Net architecture starts off similarly to a CNN, but adds an output (or expansion) portion that leverages skip connections that pass in information from the input (or contraction) portion of the architecture. For preprocessing, the MRI scan images were cropped to square shapes to simplify processing and normalized. A 34 layer U-Net model was created, and dice loss was used as the key evaluation metric. The model was trained for 40 epochs, with a dice score of 0.67 being achieved. The authors did not provide complete nor correctly functioning code to reproduce.

- **2023: Image Segmentation: Train a U-Net Model to Segment Brain Tumors**
  - https://blog.ovhcloud.com/image-segmentation-train-a-u-net-model-to-segment-brain-tumors/
  - Code: https://github.com/ovh/ai-training-examples/blob/main/notebooks/computer-vision/image-segmentation/tensorflow/brain-tumor-segmentation-unet/notebook_image_segmentation_unet.ipynb
  - Code reproduction: https://github.com/stjoha81/Capstone_2023_2024/blob/78c07ff152a444a72890fec0b260323e1d4a4f5f/Reproduction%20of%202023%20Image%20segmentation%20-%20Train%20a%20U-Net%20model%20to%20segment%20brain%20tumors.ipynb
  - The authors used a U-Net architecture to detect brain tumor segmentation and make predictions using MRI scans. The ~300 samples consist of 4 types of image scans (“modalities”) for each patient, along with the “ground truth” image segmentation that was manually created by experts. Each of the 4 brain scans consist of over 100 image slices, which can be stacked to represent a 3D object. To reduce computing resource needs, and for efficiency, only 2 modalities plus the manual segmentation results were used. The range of images used for each modality was reduced to reduce images slices with little or no useful information. The images were preprocessed (including conversion into an easily readable format, resizing, and loading into Numpy arrays), and the ground truth was

one-hot encoded to be used as the y values. A 32 layer U-Net model was created that used softmax activation at the output layer. Categorical cross entropy was used for the loss function, and the model was trained for 35 epochs. This was a mult-classification model, used to predict whether a given image represented one of these 4 classes: no tumor, necrotic or non-enhancing tumor, a peritumoral edema, or an enhancing tumor . Multiple evaluation metrics were used, including accuracy (99%) and dice coefficient (0.60). Post processing was used to address a false positive.

- **2023: Enhanced Brain Tumor Classification using Graph Convolutional Neural Network Architecture**
    - https://www.nature.com/articles/s41598-023-41407-8
    - Researchers utilized a GNN to create input for their 26 layer CNN. The GNN modifies the data in one area of an image by combining it with data from nearby areas. The goal of this approach was to better identify pixel similarity based on proximity to other pixels, specifically by considering non-Euclidean distances in images. For preprocessing, data augmentation and resizing were performed, and a Gaussian filter was applied to normalize the images. The graph nodes created were an average of the data from its nearest pixel neighbors. The CNN has alternating layers consisting of convolution, max pooling, and batch pooling layers, along with two dropout layers. The model achieved 95% accuracy when trying to detect glioma, meningioma, pituitary or no tumor.

- **<u>Conclusion/Analysis:</u>**

The coursework I have completed so far prepared me very well to be able to understand the research papers and articles. I was able to think through each step of the process of reproducing the authors' code. This was valuable because there were changes needed due to file paths on my computer being different than what the researchers used, and because changes were necessary due to package changes or changes in function arguments. I also encountered a couple of issues specific to my Apple Macbook computer that necessitated workarounds. Some papers did a better job than others in describing what they were doing, and I was able to apply my artificial intelligence and machine learning knowledge to figure out any gaps in the descriptions the researchers provided. Some aspects that were new to me, such as using post processing after training a model to try to address issues of false positives. Also there was a clear progression with respect to accuracy as time progressed and more sophisticated models were used. Finally, the combination of a graph based model and a CNN did not have as high an accuracy as models from other papers, but it did address a pixel proximity challenge the others did not.

Overall I found this to be both a challenging and worthwhile exercise to help me better prepare for creating my own capstone model. I will need to consider not only network architecture and hyperparameters, but also comparison of metrics across models to decide on the best architecture to use.