## Architectures

We investigated several alterations on convolutional neural networks for feature extraction, into densely connected networks for progression prediction. The framework of the feature extraction has three parts: First, convolution blocks are performed in a manner similar to the DenseNet(Huang et al., n.d.) architecture, where previous feature maps are concatenated immediately to later layers. Second, a transition block reduces the number of filters by a fraction ranging from 0.5 to 1, where 1 indicates no reduction and 3D average pooling with a stride of 2. Third, feature extraction ends with a flattening of the extracted features. The framework of the densely connected network has two parts: First, a set number of dense connections followed by dropout layer. Second, the model predicts two classes, progression and no-progression, with a soft-max activation.

We investigated several parameters in both the feature extractor, densely connected prediction, and parameters outside of the architecture.

Feature extraction parameters:

* 4, 8, 16, or 32 filters as the number of filters to start the feature extraction
* 4, 8, 16, or 32 as the rate of increase in the number of filters (growth rate)
* 1, 2, 3, 4, or 5 convolution blocks in each dense layers
* 1, 2, 3, or 5 dense convolution blocks followed by transition layers
* 0.5, 0.75, or 1.0 the fraction of filters to decrease by each transition

Densely connected parameters:

* 0, 1, 2, 3, 4, 5, or 7 number of densely connected layer
* 64, 128, 256, or 512 number of connections in each layer
* 0.5 or 0.0 dropout for fraction of connections to drop

Other hyper-parameters:

* Min and max learning rate were created on a model by model basis
* Adam or Stochastic gradient descent optimizers
* Categorical cross entropy, or Cosine loss

Due to the limited, and biased (more non-progression than progression) nature of our dataset, we quickly noticed a problem of model overfitting. Recent work has shown that a large part of this issue can be related to the usage of softmax + crossentropy loss function(Barz & Denzler, n.d.), and propose the usage of a cosine loss function. The cosine loss focuses on increasing the similarity between the L2-norm of the prediction and the ground truth, ‘bounding’ the loss to a unit sphere. In contrast, the cross entropy and softmax contains exponential and log functions, allowing arbitrarily high values to appear.