# Diabetic Retinopathy Detection

Authors: Aishwarya Vantipuli, Aaron Moskowitz, and Qiang Fang

**Introduction:**

Diabetic retinopathy is the leading cause of blindness in the working-age population of the developed world and is estimated to affect over 93 million people. Currently, detecting DR is a time-consuming and manual process that requires a trained clinician to examine and evaluate photographs of the retina.

With color fundus photography as input, the goal of the project is to introduce an automatic DR grading system capable of classifying images based on disease pathologies from four severity levels.

**Dataset:** https://www.kaggle.com/c/diabetic-retinopathy-detection/data

Diabetic retinopathy images were acquired from a Kaggle dataset of size 84GB. The dataset comprises high-resolution retina images taken under a variety of imaging conditions that vary in height and width between the low hundreds to low thousands of pixels.

The left and right fields are provided for every subject. Images are labeled with a subject id as well as either left or right (e.g. 1_left.jpeg is the left eye of patient id 1). A clinician has rated the presence of diabetic retinopathy in each image on a scale of 0 to 4, according to the following scale:

0 - No DR 1 - Mild 2 - Moderate 3 - Severe 4 - Proliferative DR

**Pre-Processing:**

During preprocessing, images will be cropped to isolate the circular colored image of the retina. We will try different methods to preprocess image using OpenCV, Otsu's Method, or other python techniques. We will re-scale the images to have the same radius, subtract the local average color, remove noise by Gaussian blur, clip the images to a smaller size (90%) to remove boundary effects, remove some of the variation between images due to differing lighting conditions or camera resolution.

Color images would likely be normalized to represent pixels in the range 0 to 1 across matrixes for red, blue and green before use in the models. There might be many color descriptors and texture descriptors and shape descriptors, so we can perform feature engineering with OpenCV or Python libraries.

**Approach and Methodology:**

We are planning to use ordinal regression, Random Forest or Support Vector Machines (SVM), and Convolutional Neural Network (CNN) models for image classification tasks and compare the results. Ordinal Regression will minimize log-likelihood loss when predicting hierarchical classes of disease severity. All models except for a CNN model will use Scikit-learn. We will also demonstrate the use of a fully trained or transfer learning-based CNN using TensorFlow and Keras.

A training set of at least 10 GB across 10 K records and a test set of at least 4,220 records of a size of 4.23 GB will be used to evaluate each model. 10 K-fold cross-validation on the training set is implemented for hyperparameter selection. Advanced models such as the CNN can use 15% of the training data as a validation set to make decisions related to using features of textures and shape detection, and on model architecture design.

**Model Evaluation:**

We will evaluate the results by using metrics like log-likelihood loss, sensitivity, specificity, precision, and recall depending on the outcome. ROC curve can be implemented for healthy (severity == 0) and sick (severity>0) to see how well the model works at just identifying the disease.