Executive Summary

The diagnosis of blood-based diseases often involves identifying and characterizing patient blood samples. Current process involves cell sorters but automated methods to detect and classify blood cell subtypes can have important medical applications. For this project my goal is to build an image classifier using Tensorflow to recognize the different cell types and to examine the challenges and the feasibility of building such a model. We hope to show that image classification models might one day be used in the field producing results very fast and inexpensively without having to go through additional manual steps or following additional protocols.

Project Objectives, Challenges and Scope

To build this Blood Cell classification model, I plan to use the Blood Cell Images dataset available from Kaggle (<https://www.kaggle.com/paultimothymooney/blood-cells>) and this dataset contains 12,500 augmented images of blood cells (JPEG) with accompanying cell type labels (CSV). In this dataset, there are approximately 3,000 images for each of 4 different cell types grouped into 4 different folders (according to cell type); Eosinophil, Neutrophil, Lymphocyte, Monocyte.

Image classification problems require large number of images for the training process. In the absence of the number of images needed for model training, image augmentation and transfer learning techniques are used to address the issue.

\* The Blood Cell Images dataset contains only 410 original images which were later augmented and a larger dataset of 12000 images was created. Therefore, we will not perform additional augmentation.

\* We also plan to limit the project to building a relatively simple Convolutional Neural Network (CNN) model, training and evaluating the model and we will not perform transfer learning techniques from other pre-trained image models.

Reading and processing 12000 images has its own challenges. Trying to read all the images at once, converting them to float32 tensors would require very large amounts of memory and it would not be practical. In this project my goal is to address this issue by reading a batch of images that matches the batch input for model training and managing memory requirements more efficiently.

Model Success Metrics

Production ready models would need to match or exceed the accuracy, sensitivity and specificity of the current techniques. And to achieve this level of performance metrics would probably require very deep models and/or transfer learning. Our goal is not to build a production ready model but to examine the steps and understand the challenge of building such a model, so we will not try to match or exceed the accuracy of the current techniques.

However, we will consider our model a success if:

\* The overall accuracy is 90% or better

\* True Positive Rate and True Negative Rates are above 80% for each of the Blood Type class