Preliminary Results

1. Problem statement: Restate the initial project that you proposed in deliverable one in 2 - 3 sentences. Be sure to refer back to this problem statement in the following questions.

   The garbage classifier is a machine learning program that takes as input an image containing garbage, and outputs its type (glass, cardboard, batterie, etc.).

2. Data Preprocessing: Confirm the dataset you are working with. State any changes from the initial dataset you chose. Discuss the content of the dataset (number of samples, labels, etc). Describe and justify your data preprocessing methods (did you delete or modify any data? If so, why?).

   The dataset we are working with is the same as stated in the project proposal: Garbage Classification Dataset | Kaggle
   The dataset contains 15515 images in total including all types of garbage. There are 12 classes of garbage: battery, biological, brown glass, cardboard, clothes, green glass, metal, paper, plastic, shoes, trash, and white glass.

   For the preprocessing we resized the images to 32x32 pixels and then set a batch of 32 which means it feeds the model 32 different samples.

3. Machine learning model: In the first deliverable, you proposed a model for your project. If you decided to change your model, explain why. Restate your chosen model and elaborate on the design decisions. Report the following:
   The machine learning model chosen is the convolutional neural network model.

   a. Specify the framework and tools that you used to implement your model. (For instance, did you use any libraries such as PyTorch, Keras, etc. to implement the model? Any other tools? What does the architecture of your model look like? How many layers/modules? etc.) Explain and provide architecture graphs as appropriate.

      Keras was used to implement the CNN model as well as preprocessing. The Kaggle CLI was used in order to download the dataset onto Google Colab.

   b. Justify any decision about training/validation/test splits, regularization techniques, optimization tricks, setting hyper-parameters, etc.

      We decided to split 80/20 for training validation since we figured it would make more sense to dedicate most of our data for training purposes.

   c. Description of validation methods How did you test your model? Is your model overfitting or underfitting?

Our model is underfitting (76% accuracy).

d. Did you face any challenges implementing the model? If so, how did you solve it? At this point, don't forget to save your trained weights! You will need them for the integration and/or testing of your model!

   It was hard to find the libraries and to import the dataset. We ended up working with mostly tutorial code, and we expect to work on understanding more of what our model is doing and adjusting hyperparameters later.

4. Preliminary results: In this section, you will focus on the performance of your model. Confirm the metric discussed in Deliverable 1. Present a detailed analysis of your results, providing graphs as appropriate. In addition to an evaluation metric, discuss the overall performance of the model and the feasibility of the project with these results. Remember, graphs are beautiful and we love them!

   n/a we haven't gotten here yet. Need to spend more time tuning hyperparameters before we have a good idea of what to say for this. The model trained in 10 minutes, so it was fairly slow to train, and the accuracy was 76% so it wasn't that accurate. We expect both numbers to improve with some optimization.

5. Next steps: Discuss your next steps. Describe the pros/cons of your approach and future work. Will you be altering your model? For example, will you be fine-tuning it? At this point, if you think that your model is not performing well and/or does not work, please reach out to your assigned TPM to see what you can do to improve it

   We will most definitely be fine tuning our model. We want to increase the image dimensions and then add in a couple more layers that are a little more finely tuned.