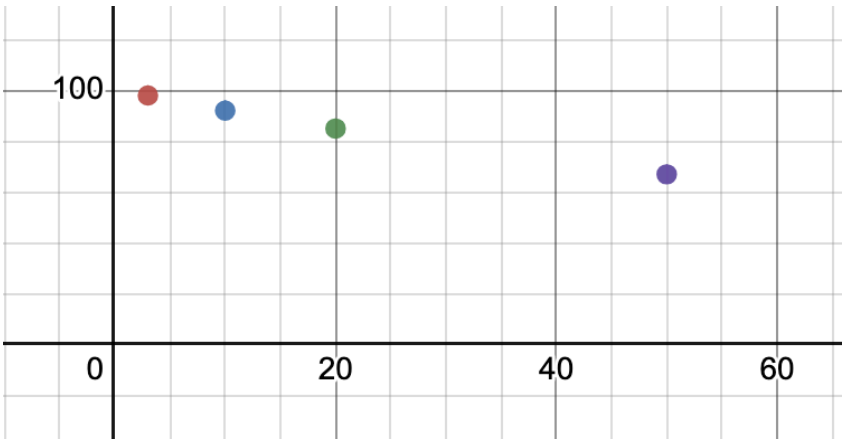
MAIS 202 - DELIVERABLE 2

Deliverable Description:

1) Our project aims to reproduce the Google Quickdraw by using The Quick, Draw! Dataset encompassing 345 categories of images. Our objective is to prompt users to draw specific objects and have the computer accurately recognize their drawings on the screen.
2) For our dataset, we selected "The Quick, Draw!" Dataset provided by Google, which features 345 diverse categories of objects for user input recognition. So far we weren't able to fully train the model with 345 categories and only made it to 50 due to processing capabilities. However, our test trials were successful with around 85% accuracy for 20 categories. Our biggest concern currently is accuracy, which presumably will go down by increasing the number of categories. To address this, we are reevaluating our Neural Network's architecture through hyperparameter tuning.
3) Currently, we have not made a decision to alter our model. More thought has to be given towards this.
    a) We used the following libraries: tensorflow and numpy. We are using two convolutional layers paired with max pooling layers which are then fed in as inputs to a flattened vector connected to a dense layer of 64 neurons. This dense layer connects to an output layer with 50 nodes, each corresponding to a category. The layer uses a ReLU while the output is activated through the softmax function. We are considering implementing additional deep layers before output.
    b) Random and less computationally expensive
    c) We're currently facing a challenge with our model's performance, which appears to be underfitting. In fact, it is achieving an accuracy rate of only about 50% when handling 50 different categories. This level of accuracy is quite low, indicating that the model is struggling to learn effectively from the training data. This is due to the fact that we are encountering significant computational limitations. Our current computers are insufficient to continue the training process, which limits the performance of our model.
    d) As mentioned before, we faced some challenges while implementing the model, especially when it came to computing capabilities. In addition, we had a discrepancy concerning accuracy when testing the model. We haven't implemented any solutions yet, but we're thinking of adding an additional layer and possibly using cloud computing.
4) Throughout our tests, we processed datasets of varying sizes: 3 (1 epoch), 10 (6 epochs), 20 (3 epochs), and 50 (1 epoch). With the last test run, we weren't able to compute more epochs

due to computational issues. Here is a graph of our model's performance with x axis being the number of categories considered and y axis being the % accuracy.



To enhance our model, we are considering upping the number of layers by potentially incorporating dense layers. Presently, we are unsure of the possibility of running our neural network locally. Aside accuracy issues, we would need to train the model on a external cloud computer due to computational constraints:

5) While we've made notable progress, particularly in achieving high accuracy for a limited number of categories, the objective is to diversify user inputs. To achieve this, we are actively working on refining the architecture of our Neural Network by contemplating the addition of layers and exploring the balance between convolutional and dense layers.