Given the task of predicting the number of Instagram likes of an Instagram post, I decided to classify Instagram likes. Given the predictors and images in the dataset, I was able to extract more features of the images, such as sharpness, contrast, and blurriness, to aid my predictive model. Given these image features and the number of followers the posting user had at the time, I predicted what class of likes an Instagram post was in (0=low, 1=medium, 2=high).

Before deciding on this classification approach, I conducted exploratory data analysis based on the 'instagram_data.csv' dataset. My first point of interest was seeing whether the quantitative predictors in my dataset were associated with the number of likes of an Instagram post. After plotting "likes" v. time and the number of followers, I noticed some clustering patterns and immediately knew that I would have to use the actual images in some way for my model.

Before creating classes and assigning them to my data, I extracted the sharpness, contrast, and blurriness of each image and added these quantitative predictors to my dataset. Upon scatter plotting these features, I noticed negative associations among sharpness and blurriness with the number of likes, which made sense to me because photos that are too sharp or too blurry might be lower quality or more unclear. I saw a positive association between contrast and likes, which made sense to me because of the visual clarity or aesthetic appeal these photos have.

After extracting and investigating the image metadata, I used K-means clustering to group my data into the 3 classes I mentioned earlier. I used K means clustering to label my data with 3 clusters to ensure that each group represented the natural properties of the dataset. Class 0 (1431 to 218688 likes) had 71.57% in it, class 1 had 24.02% in it (218801 to 574494 likes), and class 2 (575590 to 2161369 likes) had 4.41% in it. Once the labeling was done, I decided to use five classification models: Logistic Regression, KNN Classifier, SVM Classifier, Random Forest Classifier, and Gradient Boosting Classifier.

For each of the five models I used, I trained each with parameters I manually set and reported their training and test accuracies. The models that had the highest test accuracies were the Gradient Boosting classifier (0.7992) and the KNN classifier (0.7847). Next, I investigated the KNN model further to see if I could change k enough to make the test accuracy higher than the Gradient Boosting classifier by using GridSearch (although there was only 1 parameter I was interested in changing). Due to time restrictions, I decided not to investigate if I could improve the accuracy of the Gradient Boosting classifier.

Next, I plotted a feature contributions graph for the KNN classifier to see which features contributed the most to the model. It seems that the number of followers was the most important feature by a huge margin, then sharpness, and then blurriness. Overall, I believe that this project can be adapted to predict how successful an Instagram post will be. The creation of classes for the images can be modified based on the criteria someone has for a post being "successful" or "highly liked." Since I had to work with just the data given, I used K means clustering because this was a method I knew was supposed to reflect the "natural properties" of the data, but since the number of likes was skewed, a different methodology might work better. I think the use of oversampling or undersampling could have been implemented to even the group sizes out, which might have made the model more interpretable and robust. Despite the limitations of my model, I have confidence that using sharpness, contrast, and blurriness can predict whether a post will be successful or not based on a user-defined threshold.