**CS-518 Computer Vision**

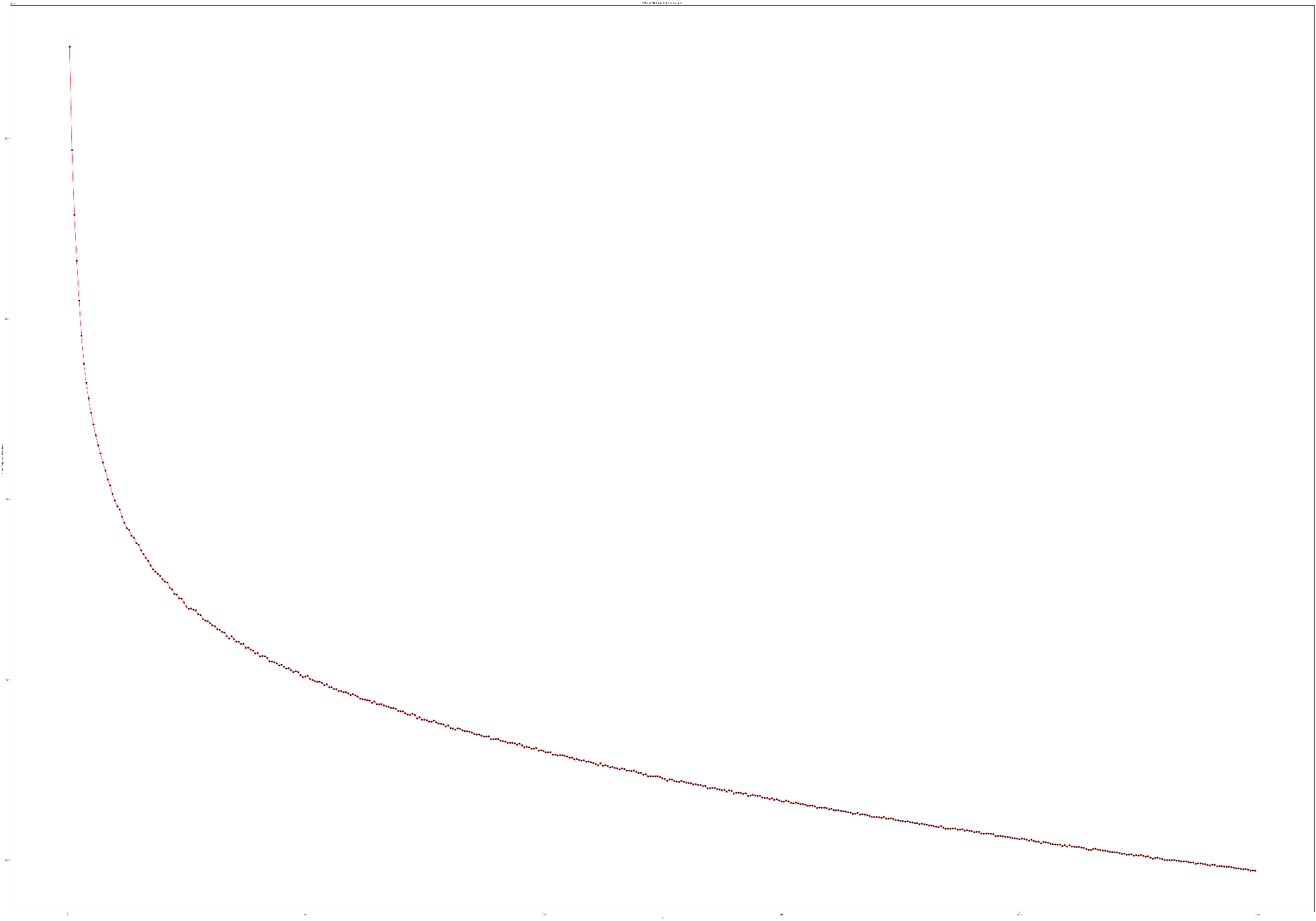
**Assignment02**

**Bag Of Visual Words based image classification on MNIST dataset**

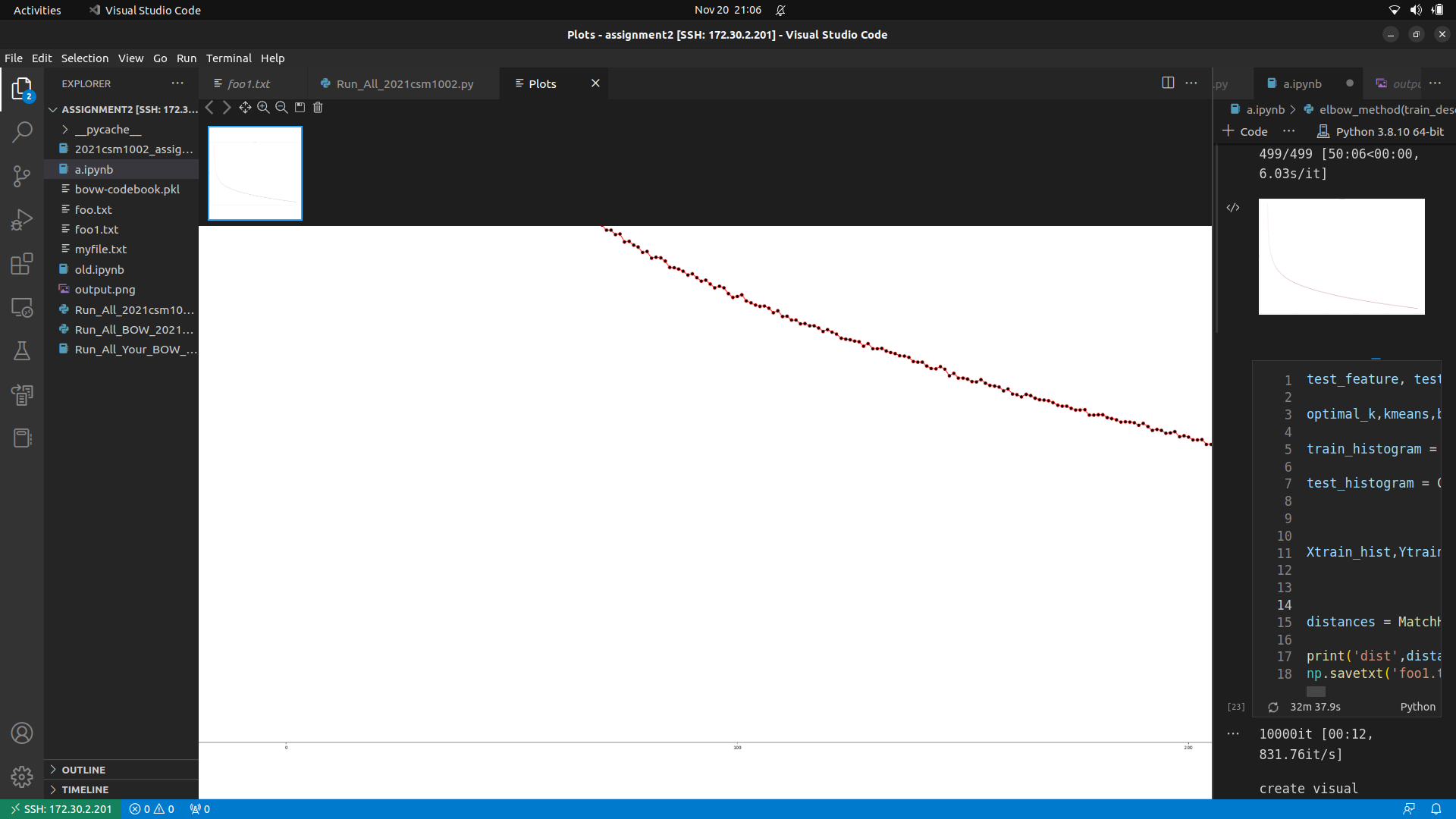
Algorithm Steps:

* **Dataset:** dataset is downloaded using tensorflow library which is divided into train and test.
* **Feature Extraction**: Use SIFT(Scale Invariant Feature Transform) to extract features such as keypoint and descriptors from train and test dataset.
* **KMeans Algorithm & selection of optimal k:** Apply Custom K Means algorithm to create a visual dictionary. The optimal value of k is decided using the **Elbow Method.**

Train descriptors are sampled for finding the optimal value of k. The elbow method iterates over a range of 1 to 500 which computes the sum of squared distances which is the distance of point from its representation and inertia. The optimal value of k is decided by the point where the graph becomes linear or elbow point. For such a range of values of the optimal value of k is 200.



The experiment is conducted over different values of range such as up to 100 , 200 and 500. The observed elbow points are 40 , 120 and 200.It can be seen by zoom in the image opening in visual studio code. I have attached the screenshot of the zoomed version of image in current directory.

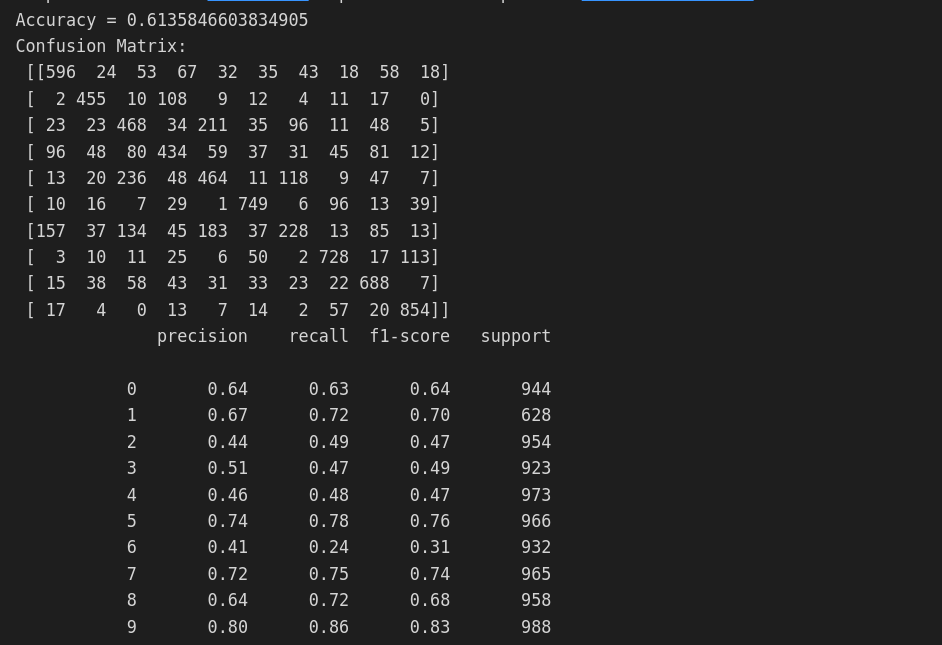


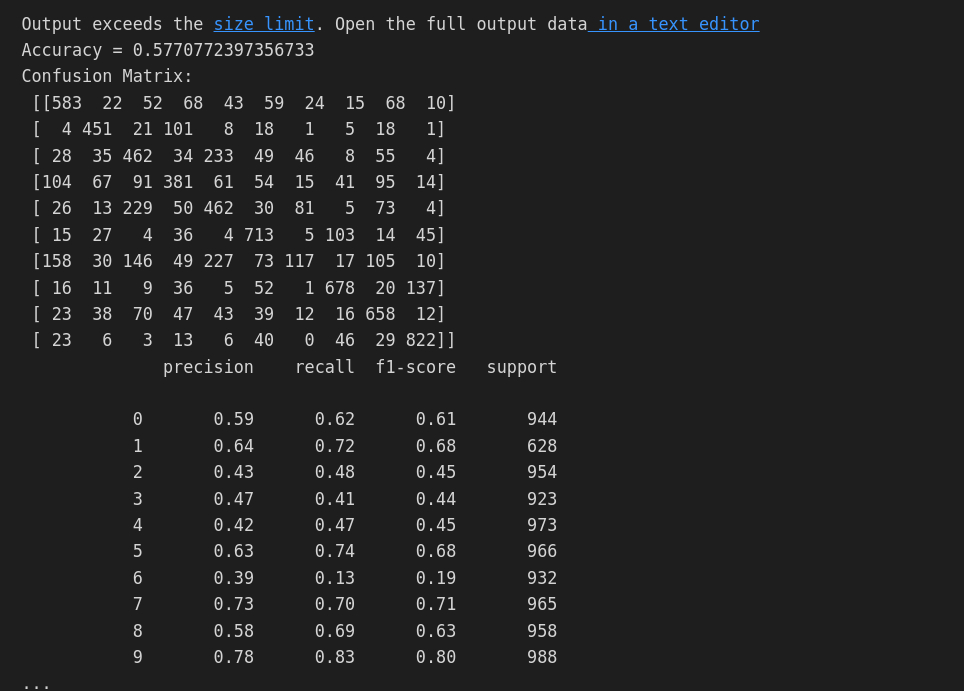
**Saving Codebook and Distance from closest word:** After applying the k-means , the generated cluster centers are saved as bovw\_codebook,pkl in the current directory and the distance of closest centroid points during k-means algorithm are saved in file in the current directory.

**Compute Histograms:** Create histograms from train and test features. Then create Xtrain and Xtest histograms for performing the classification.

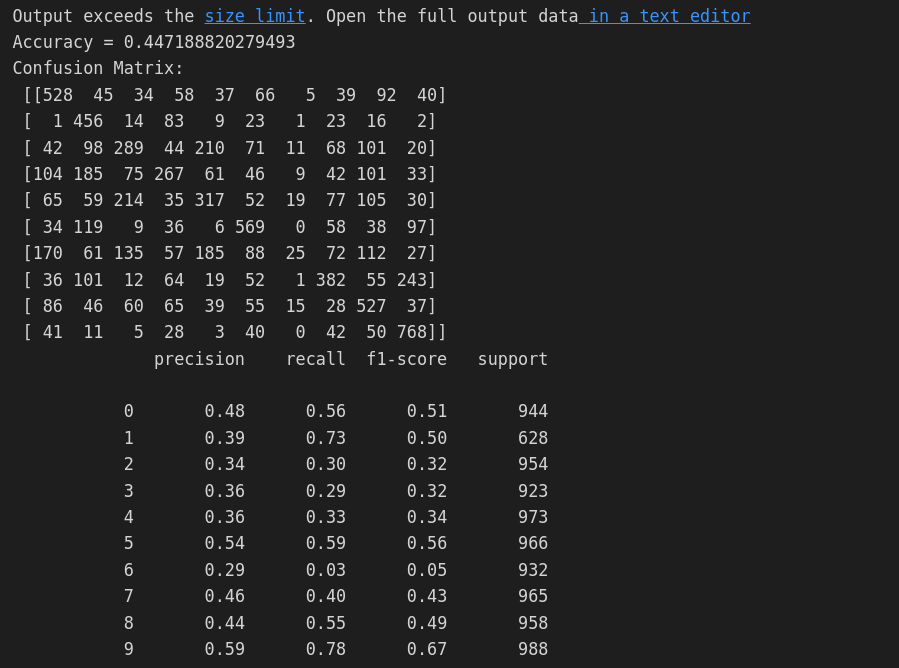
**Match Histogram & Vector Quantization:** Vector quantization basically saves the distances from k nearest cluster centers. Match histogram used train and test histograms along with its labels which use SVM classification and predict the accuracy along with classification report.

**Results:**

**k=200**

**k=100**

**k=70**



k=400 (builtin k means)