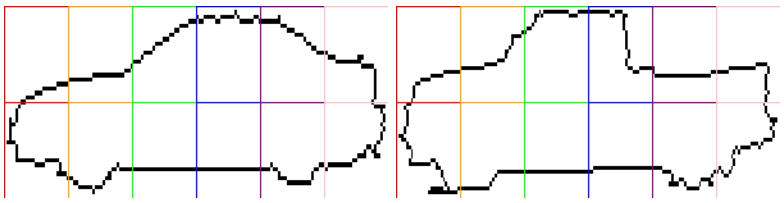
Adaptive Computation and Machine Learning Assignment2: Vehicle Classification

Tsitsi Marote 856182
University of the Witwatersrand

March 12, 2018

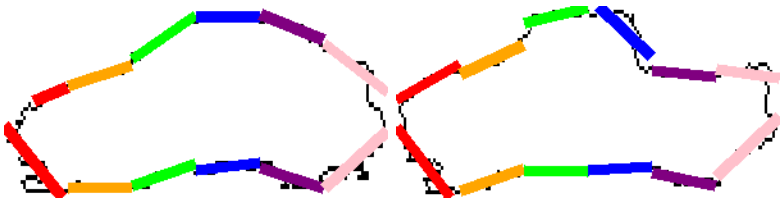
Question 1

- Firstly the images are examined thoroughly inorder to identify differences that could potentially make great features.
- It is noticed that all of the vans have a horizontal back which has a gradient of approximately 0 giving it a more orthorgonal shape at the back.
- On the other hand the cars have a more diagonal shape at the back.
- Distances between different points on the vehicle are then chosen as features.
- With this in mind, the images are resized giving them a width of 300 and height of 150.
- After the scaling process all the images are segmented using 12 equal rectangles giving 12 features in total.



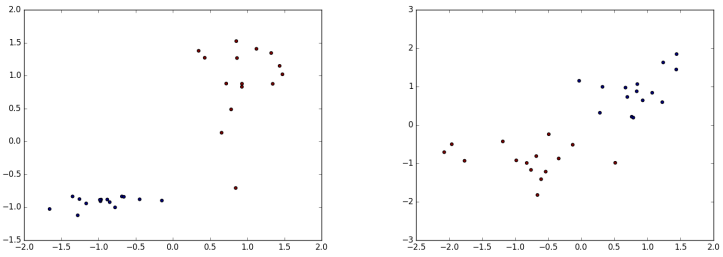
(a) Segmented car (b) Segmented van

- After segmentation RGB is used to see where borderlines are and straight lines are drawn for each feature around the vehicles.



(a) Highlighted car features (b) Highlighted van features

- After this it is seen that the first row(upper) features are most likely to be better classifiers.
- From first glance two features particularly stand out.
- The 4^{th} feature which is represented by the blue line (*see figure 3 and 4*) is diagonal for the van and horizontal for the car. This is consistent throughout the data.
- The 5^{th} feature which is represented by the purple line (*see figure 3 and 4*) is diagonal for the car and horizontal for the van.
- This is consistent for all vehicles in the dataset.
- For better visualisation of the features, scatterplots of the features against each other are plotted.



(a) Feature 3 against feature 4 (b) Feature 2 against feature 5

- From the scatterplots it is confirmed that the upper row features against each other do indeed make better features.
- In the end a feature vector of 12 distances is returned however feature 1 to 5 against each other seem to be doing the most work in terms of the classification.

Question 2

- Using cross-validation training data is *randomly* split into two, 15 for training and 15 for validation.
- First half is used for training purposes.
- The other half is kept for validation.
- Function KNN_error is used to calculate errors between training and validation.
- K begins at 1 and is incremented after each iteration.
- If the error starts increasing, the last K that gave a minimum error is returned and if the error is 0 then that K is returned.
- So ultimately K that gives smallest error is chosen in the end.

Question 3

- Firstly α is set equal to 0.1.
- θ is initialised to 0's.
- The Batch Gradient Descent is used to update θ
- θ is updated until stopping criteria is satisfied which is when the maximum absolute difference between the current and previous θ is less than 10^{-3}
- The final θ elements are approximately in the range of -0.1 and 0.09.

Question 4

- In this case there are no errors in terms of the classification meaning there is no need to improve the algorithm.
- Perhaps given a bigger database with more variation, misclassification could occur.
- That would also mean that other features would have to be considered in order not to misclassify.