Report 1st Assignment:

**What is image classification:**

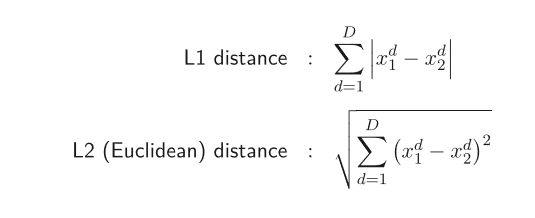
In image classification an ML algorithm learns to predict the class of unseen samples and assigns a class value called label. In classification the number of class label values is finite and discrete.

**What is the purpose of the training, validation and test sets and why do we need all of them?**

The training set is used for classifier training. Hyperparameters are tested and selected using a validation set. Test set is used for the final estimate of performance. The performance estimate is not valid, if the test set is used during training or validation. We get overfitting.

**How do knn classifiers work?**

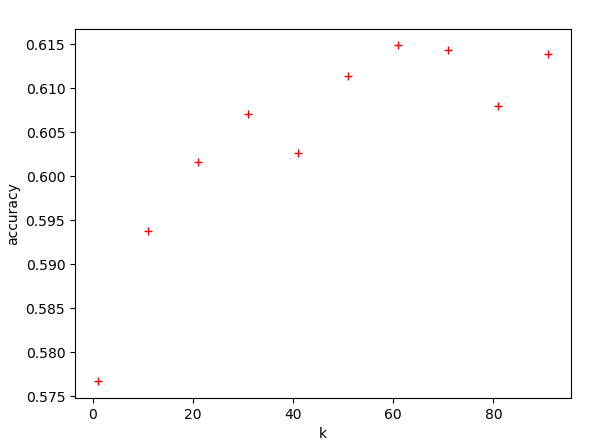
The knn classifier computes the distance between a sample and its k closest neighbors. To identify the k closest neighbors of one sample x, the distance (either L1 or L2) to all other samples has to be calculated. As a distance measure either L1 or L2 (Euclidian) can be used. The distance is computed over all dimensions.



Then the classifier looks at the label of the k nearest neighbors and assigns the most frequent label to sample x. If one wants to know how certain the classifier is, the frequency of each class label can be calculated among the k closest samples then the largest value is emphasized using the softmax function. There are also other functions to make w (vector of label frequencies) a valid probability mass function. Different k values – so the number of neighbors to be considered for classification - can be tested using a validation set. The final estimate of the accuracy is then done using a test set.

**Results knn\_cats\_dogs.py**

In the following plot the validation accuracies for different k-values can be seen:



Best accuracy using validation set is 0.6148946594806467 for k = 61.

Best accuracy using test set is 0.6175 using best k of validation set.

The knn classifier performs poorly, because it is based on average image similarity. As feature vectors we use all pixels of an image. Thus the classifier has no understanding oft he image. Dimension is large and input space is sparsely occupied, which is why distance measures become unintuitive. The solution here ist he extraction of discriminative features.