Question 1: SVM

Question 1a: role of parameter C

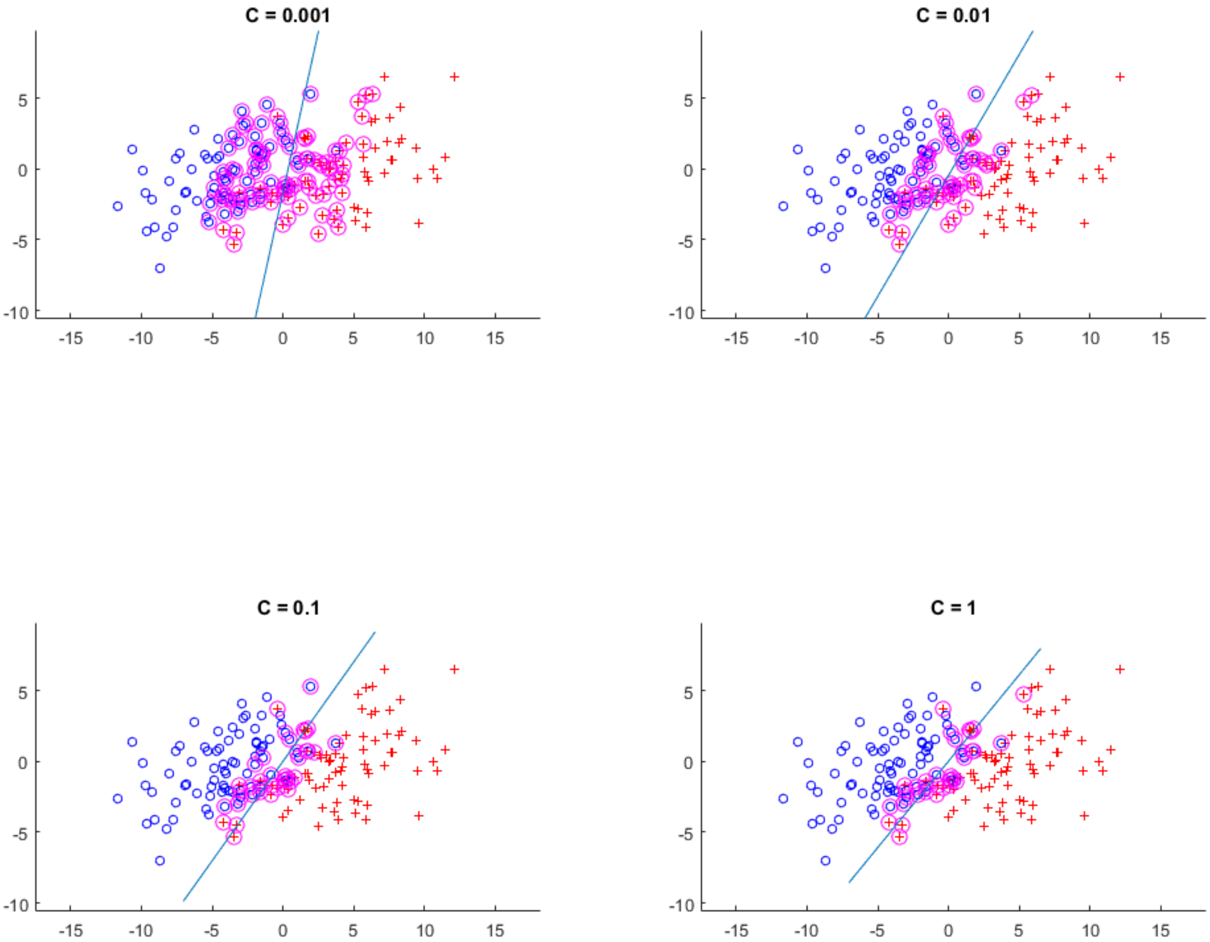

Figure 1: Plots of data and SVM decision boundary for various values of C. Support vectors highlighted in pink. Data sampled from 2D Gaussian distributions. Means at [0, 4] and [0, -4], Covariance matrices diagonal, with sigma=7.5 in the horizontal direction and sigma=5 in the vertical direction

Meaning of parameter C:

- The parameter represents a penalty for having a point within the margin or completely misclassified.
- The parameter influences the width of margin. A higher penalty means the optimum is reached with less (training data) points in the margin or misclassified, thus the margin will tend to be narrower for higher values of C (in typical point distributions, such as those in the figure). The decision boundary will also tend to be more sensitive to outliers (see for example the plot for C = 1). This may lead to poor generalization.

Question 3: Performance Evaluation

Question 3b: Comparison of HOG with/without L2 normalization

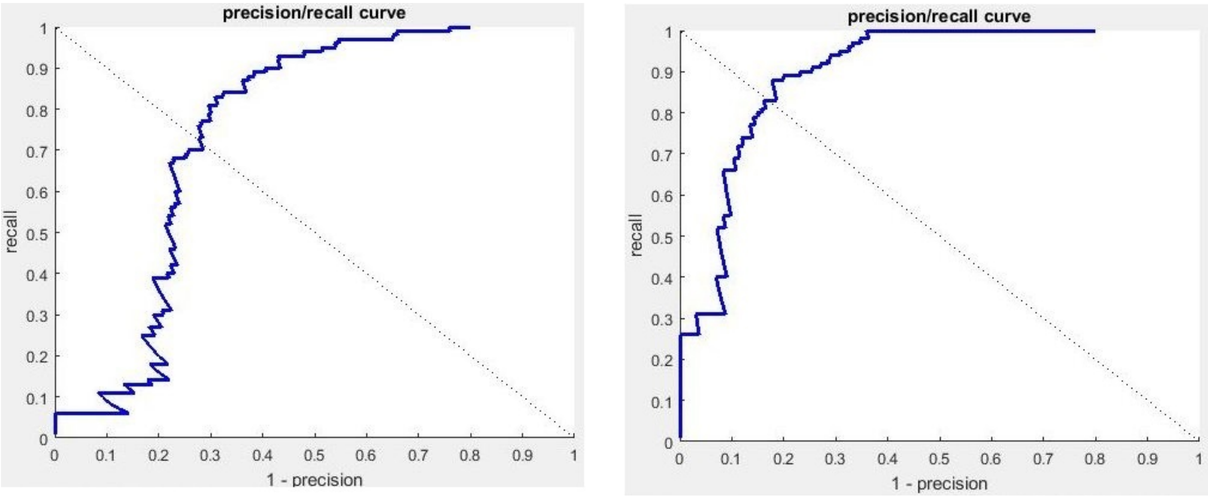

Figure 2: Recall-precision curve with/without block normalization for Histogram of Oriented Gradients. Left: without normalization, Right: with normalization. Cell size 16, block size 4 for both curves.

Question 3c: Comparison of HOG with cell size 8 or 16px

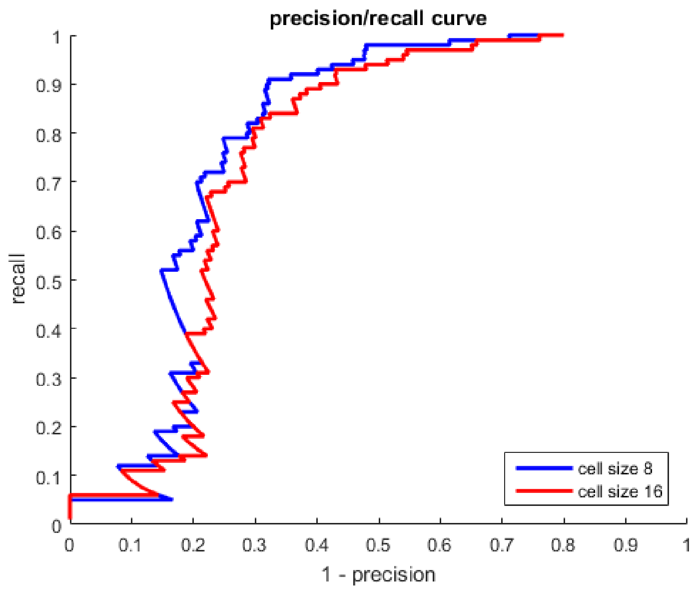

Figure 3: Recall-precision curve for different cell sizes of Histogram of Oriented Gradients.

Question 3d: Summary of comparisons

Observations:

- L2 normalization significantly improves performance at practically all levels of precision. With L2, we keep 1.0 precision up to recall values of almost 0.3, and we reach 1.0 recall before dropping below 0.6 precision. Without L2, 1.0 precision is attained only for very low values of recall (below 0.1), and we do not reach 1.0 recall before dropping precision below 0.3. In their original CVPR 2005 paper Dalal and Triggs suggest that normalization helps invariance to illumination changes (which tend to be multiplicative in nature, and thus can be helped by normalization).

- At first glance there seems to be very little difference in performance between cell size 8 and 16. We have combined the two recall-precision curves into a single plot to highlight the differences. The plot shows cell size 8 performs slightly better almost everywhere along the curve. We believe this may be due to smaller cells producing longer feature vectors (essentially more features), which may help in training a more discriminative classifier.

Question 3e: How to use this for detection?

- We could use this classifier in combination with a sliding window over the entire image, returning the windows where the classifier fires with the highest scores (possibly with non-maxima suppression). (To save computing time, we could only run on a random subset of these windows, sacrificing some possible detections if such a sacrifice was acceptable in a particular application)

- The classifier is built to recognize people in windows of size 128x64. This assumption is far too strong in order to recognize people of arbitrary (pixel) size in arbitrary images. To fix this, we could run multiple different sliding windows over the image, with different scales and aspect ratios. To make our system work with these windows of various shapes sizes, we could set a number of cells instead of cell size, and vary the cell size in order to fit the set number of cells into a window of arbitrary size. This way we could get feature vectors of the same length for arbitrary window size. Another possibility could be to run multiple different sized sliding windows, but rescale the patches to 128x64 before running our classifier on them.