

Homework 4 Questions

Instructions

- 2 ethical implications questions, which will be expanded on in discussion sections.
- 5 technical questions.
- Write code where appropriate; feel free to include images or equations.
- Please make this document anonymous.
- This assignment is **fixed length**, and the pages have been assigned for you in Gradescope. As a result, **please do NOT add any new pages**. We will provide ample room for you to answer the questions. If you *really* wish for more space, please add a page *at the end of the document*.
- **We do NOT expect you to fill up each page with your answer.** Some answers will only be a few sentences long, and that is okay.

Ethical Implications

Q1: Models can only perform as well as the data we give them. One specific concern is that high overall accuracy can mask poor performance in subgroups. Joy Buolamwini's 2018 paper, [Gender Shades](#) found that while Microsoft's gender classification algorithm achieved 94% overall accuracy, it achieved 100% accuracy on white male faces and only 79.2% accuracy on darker female faces. In response to the report, Microsoft [significantly improved](#) their algorithm's performance by collecting and using more representative training data.

In this assignment, we train a scene recognition model using data from Lazebnik et al. 2006. It is impossible to completely mitigate biases in a dataset. Please review the data to check for potential biases: look at images in the data/train and data/test directories and consider their class labels.

Please list at least three potential biases in the dataset. If the outdoor images in the dataset were used to train a model for object recognition for a self-driving car, what are some consequences that could arise as a result of these biases? [3-4 sentences]

A1: Your answer here

Q2: It can be hard to find and collect representative data. One approach that companies and researchers have used is web scraping, which pulls data from different websites to easily create large datasets. This practice has come under increasing scrutiny, however. In 2021, France became the latest country to find that Clearview AI, which operates a facial recognition platform for law enforcement use, [violated privacy laws](#) by scraping 10 billion images of people's faces from Facebook, YouTube, and other websites without their consent.

Given your answer to the previous question, please find additional scene data that could be used in conjunction with and fills some of the gaps in the Lazebnik et al. 2006 dataset. Here are two places to find datasets: [Google](#) and [Kaggle](#).

Please provide a link to your dataset, and describe how it addresses some of the gaps in the project dataset and any limitations of the new data. How was this new data collected? How was the Lazebnik et al. 2006 dataset collected? If you had to collect new scene data yourself in the real world, how would you do it? [6-7 sentences]

A2: Your answer here

Technical Questions

Q1:

- (a) Define these common terms in machine learning:
 - (i) Bias
 - (ii) Variance
- (b) Define these terms in the context of evaluating a classifier:
 - (i) Overfitting
 - (ii) Underfitting
- (c) How do overfitting and underfitting relate to bias and variance?

Please answer on the next page.

A1: Your answer here.

Q2: Suppose we create a visual word dictionary using SIFT and k-means clustering for a scene recognition algorithm. Examining the SIFT features generated from our training database, we see that many are almost equidistant from two or more visual words.

- (a) Why might this affect classification accuracy?
- (b) Given the situation, describe *two* methods to improve classification accuracy, and explain why they would help.
These can be for k-means, or otherwise.

A2: Your answer here.

Q3: The way that the bag of words representation handles the spatial layout of visual information can be both an advantage and a disadvantage.

- (a) Describe an example scenario for each of these cases.
- (b) Describe a modification or additional algorithm which might overcome the disadvantage.
- (c) How might we determine whether bag of words is a good model?

A3: Your answer here.

Q4: Given a linear classifier such as SVM which separates two classes (binary decision), how might we use multiple linear classifiers to create a new classifier which separates k classes?

Below, we provide pseudocode for a linear classifier. It trains a model on a training set, and then classifies a new test example into one of two classes. Please edit the pseudo-code to convert this into a multi-class classifier.

Hints: See slides in supervised learning crash course deck, plus your own research. You can take either the one vs. all (or one vs. others) approach or the one vs. one approach in the slides; please declare which approach you take.

More hints: Be aware that:

1. The input labels in the multi-class case are different, and you will need to match the expected label input for the `train_linear_classifier` function
2. You need to make a new decision on how to aggregate or decide on the most confident prediction

Note: A more efficient software application would separate the classifier training and testing into two different functions so that the model could be reused without retraining. Feel free to ignore this for now.

Please answer on the next page.

A4: Your answer here.

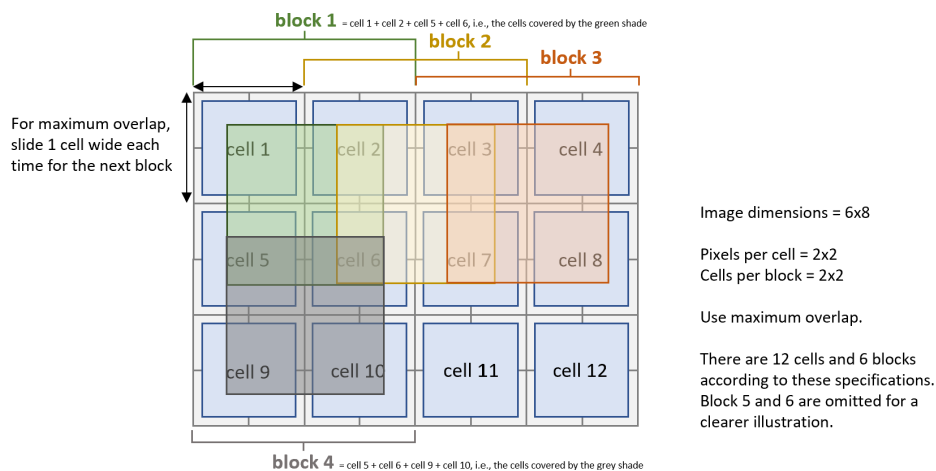
```
# Inputs
#   train_feats: N x d matrix of N features each d descriptor long
#   train_labels: N x 1 array containing values of either -1
#                 (class 0) or 1 (class 1)
#   test_feat: 1 x d image for which we wish to predict a label
#
#   Outputs: -1 (class 0) or 1 (class 1)
#
# Please turn this into a multi-class classifier for k classes.
# Inputs:
#   As before, except
#   train_labels: N x 1 array of class label integers from 0 to k-1
# Outputs:
#   A class label integer from 0 to k-1
#
def classify(train_feats, train_labels, test_feat)
    # Train classification hyperplane
    weights, bias = train_linear_classifier(train_feats, train_labels)
    # Compute distance from hyperplane
    test_score = weights * test_feats + bias

    return 1 if test_score > 0 else -1
```

Q5: In Homework 4, we will use a feature descriptor called HOG—‘Histogram of Oriented Gradients’. As its name implies, it works similarly to SIFT. In classification, we might extract HOG features across the entire image (not just at corners) to create visual words.

HOG creates a feature descriptor per image ‘block’. Each block is split into ‘cells’ covering pixels. HOG outputs a 9-bin histogram of oriented gradients per cell. We append these together to obtain the feature descriptor for each block. As a result, if we have (z, z) cells per block, the feature descriptor for each block will be of size $z \times z \times 9$.

Blocks can overlap as displayed in the diagram below.



When using HOG, the parameters such as pixels per cell and cells per block impact the resulting feature descriptor and so our performance on a classification task.

- (a) Given a 72×72 image, calculate the number of cells, blocks, and feature vector size that will occur when we extract HOG features with the following parameters using maximum overlap between blocks.

Scenario 1: Pixels per cell = 4×4 , cells per block = 4×4

Calculate:

Number of cells:

Number of blocks:

Dimensions of resulting feature descriptor:

Scenario 2: Pixels per cell = 8×8 , cells per block = 2×2 .

Calculate:

Number of cells:

Number of blocks:

Dimensions of resulting feature descriptor:

- (b) What are the pros and cons of the two parameter combinations? Which might you expect to have better performance?

Note: You may find it useful to look at the thesis of Navneet Dalal (co-inventor of HOG) for more on this topic. [\[Link to thesis\]](#) (pages 39, 41 in Section 4.3).

Please answer on the next page.

A5: Your answer here.

Feedback? (Optional)

Please help us make the course better. If you have any feedback for this assignment, we'd love to hear it!