PAPER C416

MACHINE LEARNING FOR IMAGING

Tuesday 17 March 2020, 15:00
Duration: 120 minutes
Post-processing time: 30 minutes
Answer THREE questions

Paper contains 4 questions

1 **Basics and Image Segmentation**

a Machine learning basics

i) How many parameters does a simple logistic regression model have where raw pixel values are used as input features?

ii) Explain the difference between L1 and L2 regularisation.

iii) Why do we prefer to split data into training, validation and test sets?

iv) Name three mechanisms for fixing high variance?

b Segmentation

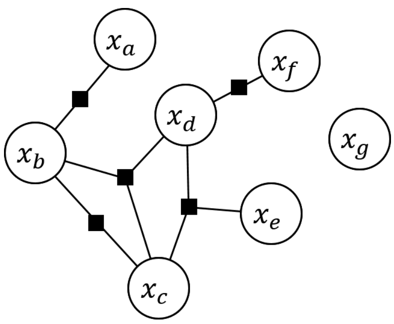
i) Explain the partial volume effect.

ii) Assume a segmentation algorithm A obtains a sensitivity of 0.7 and a precision of 0.8, while algorithm B obtains a sensitivity of 0.85 and a precision of 0.65. Which algorithm performs better overall?

iii) Segmentation performance for a new algorithm is reported in terms of its false discovery rate $FDR = \frac{FP}{FP+TP} = 0.2$ and false negative rate $FNR = \frac{FN}{FN+TP} = 0.25$. Is this algorithm better than A and/or B above? Explain your reasoning.

iv) Give the definition of the metric *accuracy* in terms of true and false positives, and true and false negatives. Why is this metric not very useful for assessing the performance of image segmentation algorithms?

v) Given the following factor graph of a Markov random field, write down the factorised joint distribution $p(x_a, x_b, x_c, x_d, x_e, x_f, x_g)$.



The two parts carry, respectively, 25% and 75% of the marks.

2 **Neural Nets and Image Registration**

a Convolutional Neural Networks

i) Given the following convolutional layers:

```
c1 = Conv2d(in=1, out=1, kernel=3, stride=1, pad=0)
c2 = Conv2d(in=1, out=1, kernel=2, stride=2, pad=0)
c3 = Conv2d(in=1, out=1, kernel=5, stride=1, pad=2)
c4 = Conv2d(in=1, out=1, kernel=3, stride=1, pad=2)
c5 = Conv2d(in=1, out=1, kernel=4, stride=1, pad=0)
```

What is the output size for an input of size $1 \times 64 \times 64$ for the following six forward passes:

```
x1 = c2(c1(x));  x3 = c4(c1(x));  x5 = c3(c1(c2(x)))
x2 = c3(c4(x));  x4 = c2(c5(x));  x6 = c1(c5(c3(x)))
```

ii) Design a simple encoder-decoder network with five convolutional layers and five transpose convolutional layers that maps an input of size $1 \times 64 \times 64$ to an output of $10 \times 64 \times 64$, with an intermediate, central representation between the encoder and decoder of size $32 \times 10 \times 10$.

iii) Calculate the output values of the following convolution once with and once without zero-padding.

$$\begin{array}{|c|c|c|}\hline 2 & 3 & 2 \\ \hline 4 & 3 & 2 \\ \hline \end{array} * \begin{array}{|c|c|}\hline -1 & 0 \\ \hline -2 & 3 \\ \hline \end{array}$$

b Registration

i) Sketch the iterative process of intensity-based image registration. What are the three main components of this process?

ii) In brief, what is a spatial transformer network?

iii) Design a simple spatial transformer network with two convolutional, two max pooling and two fully connected layers for estimating the parameters of a 2D rigid transformation for inputs of size $1 \times 64 \times 64$.

iv) Name two different loss functions that can be used to train a spatial transformer network for intensity-based, pairwise image registration.

The two parts carry, respectively, 65% and 35% of the marks.

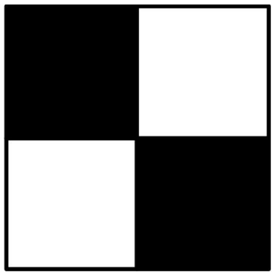
3 **Object detection and localisation**

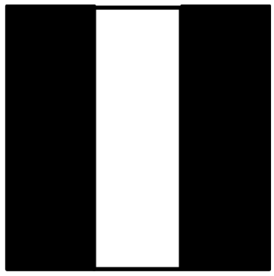
a Haar features

i) Describe how you can *efficiently* calculate a large number of different Haar features for a given image I. Give the details of each step of the calculation. Use a diagram where useful.

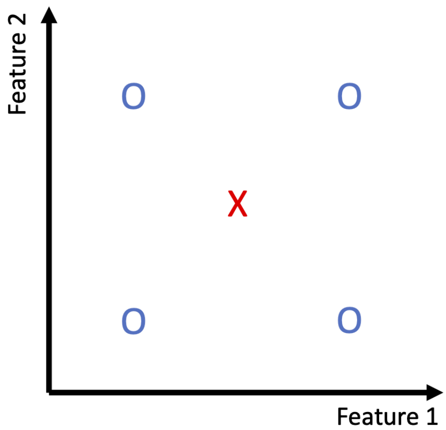
ii) For the image below calculate the value of the two Haar features shown below (using the efficient algorithm described in i). You can assume black regions have a weight of 0 while white regions have a weight of 1.

2	8	7	9	1	5
7	4	9	11	12	8
13	1	3	1	5	5
7	6	2	12	14	3
10	8	9	12	11	5
1	3	7	8	7	13





b Assume a set of images is characterised by two Haar features and is to be classified into positive (O) and negative examples (X):



Consider training a boosting classifier using decision stumps.

i) Which examples will have their weights increased at the end of the first iteration?

ii) How many iterations will it take to achieve zero training error (and why)?

c The R-CNN algorithm was one of the first algorithms that uses convolutional neural networks for object detection.

i) Briefly describe the key steps of the R-CNN algorithm for object detection.

ii) What are the disadvantages of the R-CNN algorithm?

iii) Explain how fast/faster R-CNN overcomes these disadvantages.

iv) How can an object detection method like faster R-CNN be modified to perform detection as well as segmentation of object instances?

The three parts carry, respectively, 40%, 20%, and 40% of the marks.

4 Convolutional Neural Networks (CNNs) for Classification

a Neural networks often consist of layers that include an activation function that introduces a non-linearity.

i) Give one example of an activation function that is well suited to build deep neural networks. Give one example of an activation function that is NOT suited to build deep neural networks. Justify your answer very briefly.

ii) What is the purpose of a non-linear activation function between the layers of neural networks? Why is this important?

b You are given a multi-layer CNN with a specific architecture. For each layer, calculate the number of weights, number of biases and the size of the associated feature maps. The notation follows the convention:

(a) CONV-K-N denotes a convolutional layer with N filters, each them of size $K \times K$, padding and stride parameters are always 0 and 1 respectively;

(b) POOL-K indicates a $K \times K$ pooling layer with stride K and padding 0;

(c) FC-N stands for a fully-connected layer with N neurons.

Layer	Feature map dimensions	No. of weights/biases
INPUT	128 x 128 x 3	n/a
CONV-9-32		
POOL-2		
CONV-5-64		
POOL-2		
CONV-5-64		
POOL-2		
FC-4		

c A specific CNN classifies images into four ($n = 4$) different classes, i.e. `car, truck, plane, ship`. You decide to use cross-entropy loss to train this CNN. Recall that the cross-entropy (CE) loss for a single example is defined as follows:

$$\mathcal{L}_{CE}(\hat{y}, y) = -\sum_{i=1}^{n} y_i \log \hat{y}_i$$

Here $\hat{y} = (\hat{y}_1, \cdots, \hat{y}_n)$ represents the predicted probability distribution over the classes and $y = (y_1, \cdots, y_n)$ is the ground truth vector, which is 0 everywhere except for the correct class, e.g. $y = (1, 0, 0, 0)$ for `car`, and $y = (0, 0, 1, 0)$ for `plane`.

i) Suppose you are given an example image of a plane. If the model correctly predicts the resulting probability distribution as $\hat{y} = (0.25, 0.25, 0.29, 0.21)$, what is the value of the cross-entropy loss?

ii) After initial training, the model now incorrectly predicts `ship` with distribution $\hat{y} = (0.0, 0.0, 0.45, 0.55)$ for the same image. What is the new value of the cross-entropy loss for this example?

iii) You observe that this model achieves lower loss for an incorrect prediction than for a correct prediction. Explain what may lead to this phenomenon.

iv) Given your conclusions from iii), you decide to train your neural network with the accuracy as the loss instead of the cross-entropy loss. What are the potential problems with this approach?

d You want to adopt the CNN classifier from part c) to a new problem. Now each image can have multiple classes associated with them, e.g. `car, sports-car, electric-car, truck`.

i) Propose a way to label new images, where each example can simultaneously belong to multiple classes.

ii) To avoid extra work, you decide to retrain a new model with the same architecture (softmax output activation with cross-entropy loss). Explain why this is problematic.

iii) Suggest a different activation function for the last layer and a loss function that are better suited for this multi-class labeling task.

The four parts carry, respectively, 20%, 20%, 30%, and 30% of the marks.