Name : Xinyi(Blanca) Bai ID: x652

ECE 5470 Classification, Machine Learning, and Neural Network Review

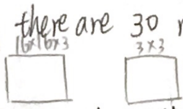
Due December ~~1~~2

Instructions:
These questions are to be answered on this document which should be submitted to blackboard in pdf format. This document is made available in both word (.docx) and pdf formats. The answers may be added to the document in many ways including handwritten and scanned. Converting the final answers to a pdf format for submission is the responsibility of the student.

For questions that ask for an explanation or definition it should be able provide a good response in one to three sentences in the space provided following the questions. For numerical questions shoe your working.

1. Suppose your input is 16x16 color (RGB) image, and you use a 3x3 convolutional layer with 30 neurons (kernels). What are the total number of weights in the hidden layer?

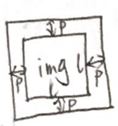
Ans: Because RGB image has 3 channels, so we need 3 dimention vectors to represent the input 16x16 matrix. In conv layer, the kernel should be 3x3x3. Considering there are 30 neurons, the total weights in hidden layer is: $3\times3\times3\times30=810$.



2. You have an input volume that is 63x63x16 and convolve it with 3 filters that are each 7x7, and stride of 1. You want the ("same convolution" setting in TensorFlow). How large is the padding?

Ans:

$63+7-1=63+P+P$



P: padding
$L=63$

Original input: $L\times L\times16$
input with padding: $(L+2P)\times(L+2P)\times16$
because each filter is 7×7 and stride is 1,
so the output is: $(L+2P-6)\times(L+2P-6)\times16$
so $L=L+2P-6$, so $P=3$

the padding is 3.

3. Which of the following statement is true about k-NN algorithm?

1. k-NN performs much better if all of the data have the same scale
2. k-NN works well with a small number of input variables (p), but struggles when the number of inputs is very large
3. k-NN makes no assumptions about the functional form of the problem being solved

no assumptions for which distribution of the modele, only 1 parameter as k

a) 1 and 2
b) 1 and 3
c) Only 1
d) All of the above
f) None of the abov

Ans: d)

4. A logistic regression classifier, which identifies an abject as one of 4 classes (a-d) , provides individual probability estimates for an image of a=0.4, b=0.2, c=0.9, and d=0.8
   a. Which class is the object identified as?

   b. what is the probability of correct classification?

Ans: a. the object should be identified as "c class", because c has higest estimation probability as 0.9.

b. $0.9/(0.4+0.2+0.9+0.8) = 0.9/2.3 = 0.3913$

5. What is data augmentation and how does it improve classifier performance.

Ans: ① data augmentation is to apply transformations to the original data in order to derive additional data for model training. common transformation, such as: add noise, flip data, rotate data, crop, scale....
② it can increase the amount of trainning dataset in avoid of overfit caused by small scale of dataset. translating dataset will focus the CNN to look everywhere and increase detecting accuracy.

6. Below is a CNN model specified in Keras

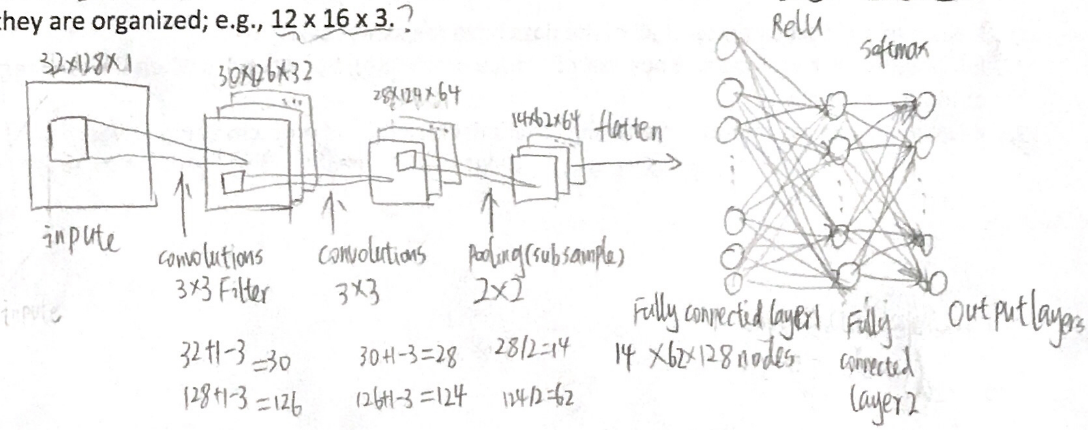
default stride (1, 1)

```
input_shape = (1, 32, 128)
model = Sequential()                       → 32 output channel
model.add(Conv2D(32, kernel_size=(3, 3),   → 3×3 moving window
                 activation='relu' activate function
64 output        input_shape=input_shape))
channel
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten()) → 128 nodes
model.add(Dense(128, activation='relu')) fully connected layer
model.add(Dropout(0.5))
model.add(Dense(num_classes, activation='softmax')) → output layer

model.compile(loss=keras.losses.categorical_crossentropy,
              optimizer=keras.optimizers.Adadelta(),
              metrics=['accuracy'])
```

Draw a block diagram of the model and clearly mark which block have trainable weights how they are organized; e.g., 12 x 16 x 3. ?

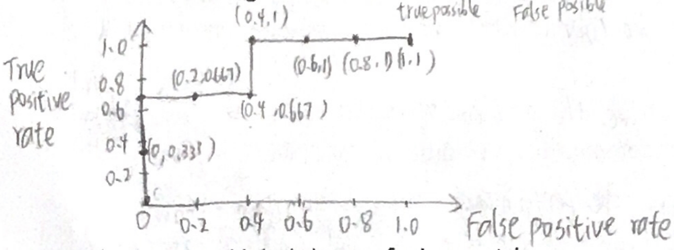
Ans:



7. In a two-class task the results from logistic regression classifier are as follows: (sorted in order of probability of correct class)

	Index	Confidence positive	Correct class
+ √	1	.97	+
+ √	2	.93	+
+ ×	3	.72	-
+ ×	4	.51	-
- ×	5	.45	+
- √	6	.33	-
- √	7	.25	-
- √	8	.20	-

a. Carefully sketch or plot the ROC graph.
Indicate the value of each point on the graph.
Show working. y: TPR true possible x: FPR False posible



threthold =	TPR (y)	FPR (x)	
0.97	$\frac{1}{3}$	0	$(0,\frac{1}{3})$
0.93	$\frac{2}{3}$	0	$(0,\frac{2}{3})$
0.72	$\frac{2}{3}$	$\frac{1}{5}$	$(\frac{1}{5},\frac{2}{3})$
0.51	$\frac{2}{3}$	$\frac{2}{5}$	$(\frac{2}{5},\frac{2}{3})$
0.45	1	$\frac{2}{5}$	$(\frac{2}{5},1)$
0.33	1	$\frac{3}{5}$	$(\frac{3}{5},1)$
0.25	1	$\frac{4}{5}$	$(\frac{4}{5},1)$
0.20	1	1	$(1,1)$

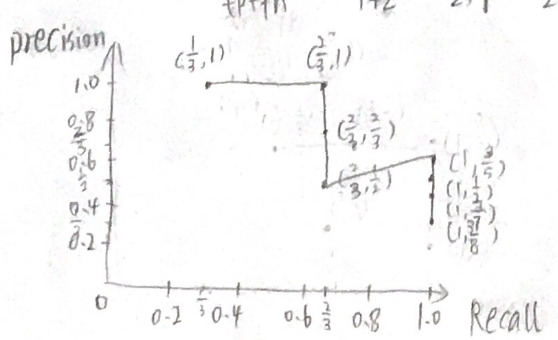
b. Draw and label the confusion matrix.

Predict \ Actual	+	−
positive	2	2
negative	1	3

fn: I guess negative, but false.

c. Extra credit: carefully draw the Precision Recall Curve (PRC)

		0.97	0.93	0.72	0.51	0.45	0.33	0.25	0.20
y: Precision	$=\frac{tp}{tp+fp}$	$\frac{1}{1+0}$	$\frac{2}{2+0}$	$\frac{2}{2+1}$	$\frac{2}{2+2}$	$\frac{3}{3+2}$	$\frac{3}{3+3}$	$\frac{3}{3+4}$	$\frac{3}{3+5}$
x: Recall	$=\frac{tp}{tp+fn}$	$\frac{1}{1+2}$	$\frac{2}{2+1}$	$\frac{2}{2+1}$	$\frac{2}{2+1}$	$\frac{3}{3+0}$	$\frac{3}{3+0}$	$\frac{3}{3+0}$	$\frac{3}{3+0}$



8. For the images used in in the Lab 7 tutorial, how are they preprocessed before being input to the classifier? List each preprocessing step and its purpose:

Ans: step1: the ~~input~~ image is 28×28 pixels in gray color, gray color only has 1 dimension,
so the image matrix would be of 28×28×1, elements are integers within the range of 0 to 255.
There are 28×28=784 values in a matrix, which can be converted into a 784-dimensional vector.

step2: in each vector, the color values are scales into float-point value, and standarlize them in the range of 0 to 1.

9. How is a ResNet convolutional layer different than a traditional CNN layer?

Ans: In ResNet, it has residual mapping module which allows the network has more layers and hyperparameters. ResNet has more than 100 layers. It has less pooling layer, more ~~sub~~ subsamping layer. ResNet doesn't have Dropout layers, it has "bottleneck" structure. (1×1 kernel size and 3×3 kernal size).
In traditional CNNs, when increasing the conv layers, the performance of model will decrease, and overfit will appears.
ResNet use its residual module to solve the problem of overfit due to many conv layers.

10. What is the advantage of a Reset layer compared to a traditional CNN layer?

Ans: In ResNet, when increasing conv layers, the performance increases, overfit is avoided by risidual structure so that it can have over 100 Layers. Traditional CNN cannot contain many conv layers.

ResNet use more subsampling instead of pooling, which increase the propogation rate.
ResNet doesn't use Dropout layer, ~~it~~ instead, it has BN to do normalization, which increase the trainning speed.