Deep Learning Worksheet 3:
Neural Networks and Minimization

Note: All questions have equal weight.

Note: Multiple answers are sometimes possible, only the correct combination of answers counts.

Submission deadline: 23:59 MEST, Friday, May 19th, 2018.

Characterization of the learning problem

Q1: Multiple applications of PCA

Consider a dataset $\mathbf{X} \in \mathbb{R}^{T \times n}$. We perform PCA on the data $\mathbf{X}$ by computing

$$\mathbf{C}^{(1)} = \mathbf{X}^\top \mathbf{X}$$
$$\mathbf{C}^{(1)} \left(\sigma_i^{(1)}\right)^2 = \mathbf{w}_i^{(1)} \left(\sigma_i^{(1)}\right)^2$$

and projecting onto the $m < n$ principal components $\mathbf{W}_{1...m}^{(1)} = [\mathbf{w}_1, ..., \mathbf{w}_m]$ with largest eigenvalues:

$$\mathbf{Y} = \mathbf{X}\mathbf{W}_{1...m}^{(1)}.$$

We perform PCA a second time by computing:

$$\mathbf{C}^{(2)} = \mathbf{Y}^\top \mathbf{Y}$$
$$\mathbf{C}^{(2)} \left(\sigma_i^{(2)}\right)^2 = \mathbf{w}_i^{(2)} \left(\sigma_i^{(2)}\right)^2$$

projecting onto the single principal component $\mathbf{w}_1^{(2)}$ with largest eigenvalue $\left(\sigma_1^{(2)}\right)^2$, resulting in $\mathbf{z} \in \mathbb{R}^T$.

Question: Which of the following is true:

1. $\sigma_1^{(2)} < \sigma_2^{(2)}$
2. $\mathbf{w}_1^{(1)} = \mathbf{w}_2^{(2)}$
3. $\mathbf{Y}_{*,1} = \mathbf{z}$
4. $\left\|\mathbf{w}_1^{(1)} - \mathbf{w}_1^{(2)}\right\| = 1$

Q2: Autoencoder

Consider grayscale image data with 128×128 pixels. We consider two autoencoder architectures with Rectified Linear Units as nonlinearities:

- **Dense**:
  - Input layer
  - Dense hidden layer with 256 neurons
  - Dense output layer.
- **Convolutional**:
  - Input layer

– Strided convolutional layer with 64 filters, stride 8, "valid" padding (no zeros added) and a filter size of 8×8.

– Max-Pooling across the 64 channels.

– Strided transposed convolutional layer with 64 filters, stride 8 that recover the input dimensions as output.

– Sum-Pooling across the 64 channels.

Question: Ignore the bias parameters. How many parameters do we have?

	A	B	C	D	E	F	G
Parameters dense	32,768	4,194,304	4,194,304	8,388,608	8,388,608	16,777,216	16,777,216
Parameters convolutional	1,048,576	4,096	8,192	8,192	16,384	8,192	16,384

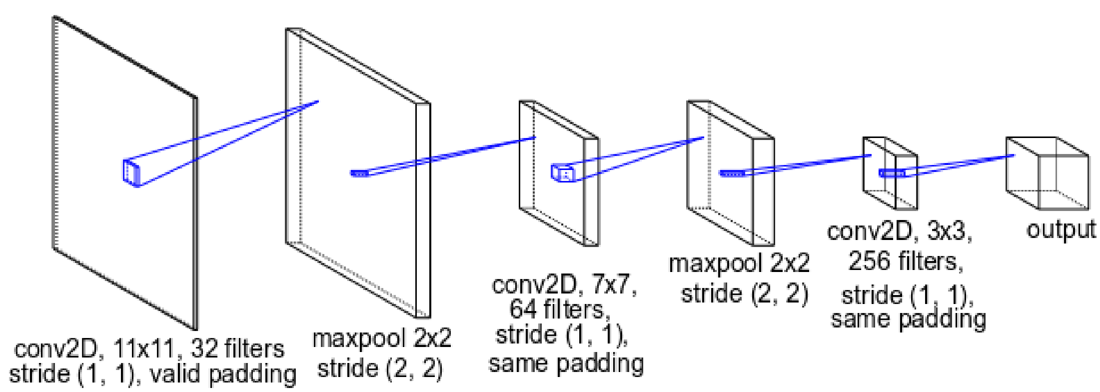
Q3: PCA

Given a real-valued dataset $\mathbf{X}$ and the covariance matrix $\mathbf{C} = \mathbf{X}^\top \mathbf{X}$. Which of the following statements is **not** true

1. The PCA eigenvalues can come in complex conjugate pairs.
2. $\mathbf{C} = \mathbf{C}^\top$
3. The PCA eigenvalues are always non-negative

Q4: Convolutional NN

Given a data set with 130×130 px images and 3 channels, consider the following convolutional neural network:



What is the output dimension?

- $30 \times 30 \times 256$
- $30 \times 30 \times 524288$
- $60 \times 60 \times 256$
- $60 \times 60 \times 524288$

Q5: Kernels

We have a Gaussian function in two dimensions given by

$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$.

One can use such a Gaussian to parameterize a convolution kernel K by, e.g., spanning an odd-sized equidistant grid centered around $(0, 0)^T$, evaluating G on the grid points, and normalizing the resulting matrix with respect to the sum of its values.

Can K be written as outer product $K = \mathbf{u}\mathbf{v}^T$?

1. Yes, in this case
2. No
3. Yes, this can be done for any kernel

Q6: Output dimensions

Consider a convolution kernel of size $n \times n$, a stride $s > 0$, and an input image size of $w \times h$. Assuming w, h, and n being divisible by the stride s, what is the output dimension when applying valid padding?

1. $[(w - n)/s + 1] \times [(h - n)/s + 1]$
2. $[(w - 2n)/s + 1] \times [(h - 2n)/s + 1]$
3. $[wh/s + 1] \times [wh/s + 1]$
4. $[(w - n) * s + 1] \times [(h - n) * s + 1]$
5. $[(w - n)/s - 1] \times [(h - n)/s - 1]$

Q7: Sparsity in convolutional nets

Given images of dimension 8×10 and a network of depth d that contains the same convolutional layer d times with kernel size 5×5, stride 1, and same padding. At which minimum depth does every feature pixel in the last layer depend on all input pixels?

1. 4
2. 5
3. 6
4. Such a convolutional layer never depends on all pixels of the input image.

Q8: Valid padding

Given a concatenation of d identical convolutional layers with valid padding, kernel size 4×4, stride 2 and input image size 20×20. At which depth d of the network do the kernel dimensions exceed the dimensions of the layer's input?

1. $d = 2$
2. $d = 3$
3. $d = 4$
4. $d = 5$