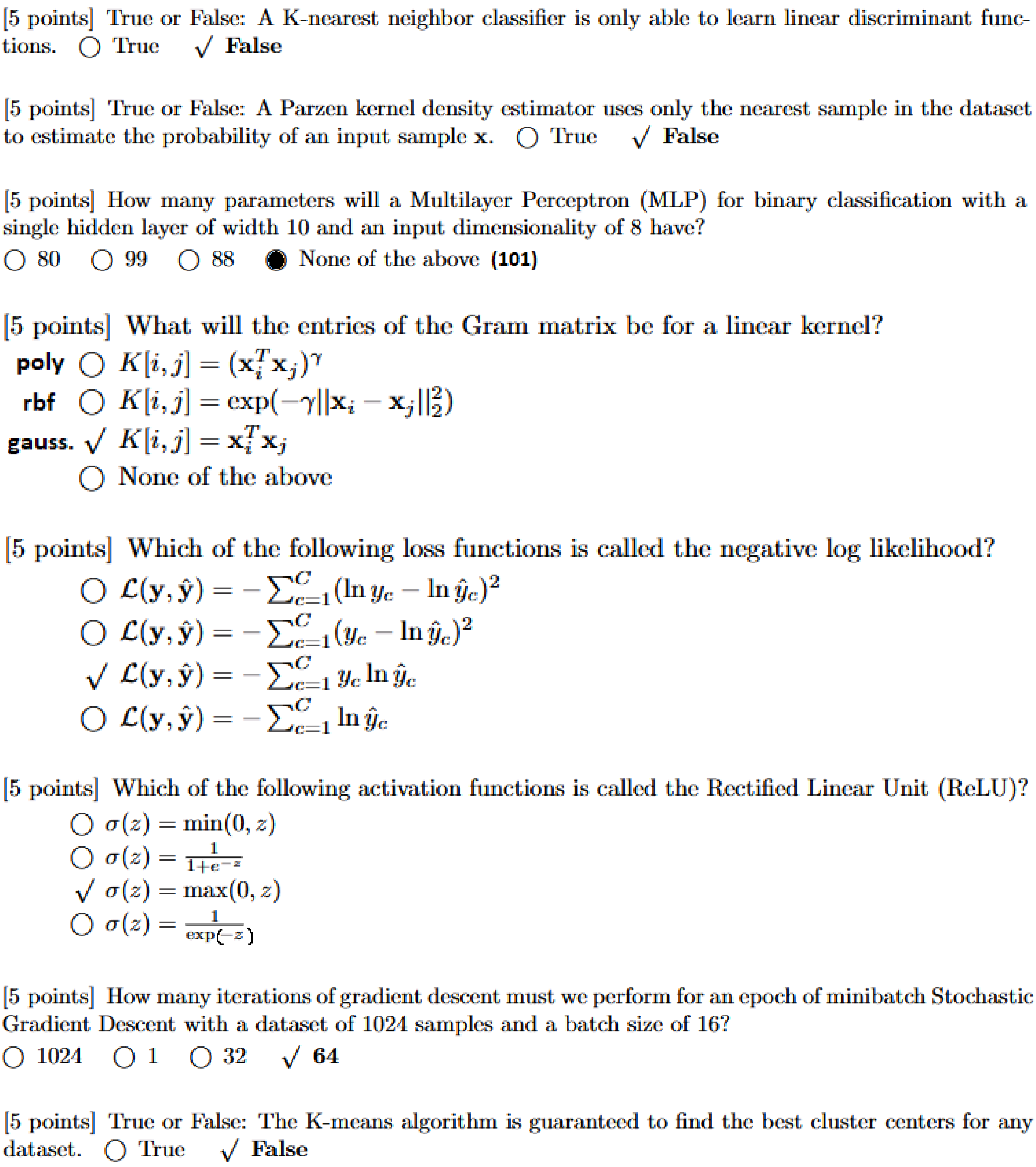


[5 points] True or False: A K-nearest neighbor classifier is only able to learn linear discriminant functions. ○ True ✓ **False**

[5 points] True or False: A Parzen kernel density estimator uses only the nearest sample in the dataset to estimate the probability of an input sample **x**. ○ True ✓ **False**

[5 points] How many parameters will a Multilayer Perceptron (MLP) for binary classification with a single hidden layer of width 10 and an input dimensionality of 8 have?
○ 80 ○ 99 ○ 88 ● None of the above **(101)**

[5 points] What will the entries of the Gram matrix be for a linear kernel?
poly ○ $K[i,j] = (\mathbf{x}_i^T\mathbf{x}_j)^\gamma$
rbf ○ $K[i,j] = \exp(-\gamma||\mathbf{x}_i - \mathbf{x}_j||_2^2)$
gauss. ✓ $K[i,j] = \mathbf{x}_i^T\mathbf{x}_j$
○ None of the above

[5 points] Which of the following loss functions is called the negative log likelihood?
○ $\mathcal{L}(\mathbf{y},\hat{\mathbf{y}}) = -\sum_{c=1}^{C}(\ln y_c - \ln \hat{y}_c)^2$
○ $\mathcal{L}(\mathbf{y},\hat{\mathbf{y}}) = -\sum_{c=1}^{C}(y_c - \ln \hat{y}_c)^2$
✓ $\mathcal{L}(\mathbf{y},\hat{\mathbf{y}}) = -\sum_{c=1}^{C} y_c \ln \hat{y}_c$
○ $\mathcal{L}(\mathbf{y},\hat{\mathbf{y}}) = -\sum_{c=1}^{C} \ln \hat{y}_c$

[5 points] Which of the following activation functions is called the Rectified Linear Unit (ReLU)?
○ $\sigma(z) = \min(0, z)$
○ $\sigma(z) = \frac{1}{1+e^{-z}}$
✓ $\sigma(z) = \max(0, z)$
○ $\sigma(z) = \frac{1}{\exp(-z)}$

[5 points] How many iterations of gradient descent must we perform for an epoch of minibatch Stochastic Gradient Descent with a dataset of 1024 samples and a batch size of 16?
○ 1024 ○ 1 ○ 32 ✓ **64**

[5 points] True or False: The K-means algorithm is guaranteed to find the best cluster centers for any dataset. ○ True ✓ **False**

[5 points] True or False: Projecting a dataset onto its first principal component minimizes the squared distances between the original and projected points. ( ) True ( ) False

[5 points] True or False: Multilayer Perceptrons (MLPs) generally outperform Convolutional Neural Networks (CNNs) on image classification problems. ( ) True ( ) False

[5 points] True or False: Adding residual connections to a CNN can make number of model parameters increase. ( ) True ( ) False

[5 points] Which of the following are advantages of Ensemble Models (e.g. Committees)? Bagging

Boosting

- √ **They reduce the variance of the resulting model.**
- ○ They are much more efficient than the base model.
- √ **They can reduce the expected error of the final model.**
- ○ The resulting model is nonlinear even if the base model is linear.

[5 points] Which of the following are requirements for applying backpropagation to compute gradients in a deep network?

- ○ The network must not be too deep.
- √ **The network must be a directed acyclic graph.**
- √ **All activation functions must be differentiable.** (Almost everywhere)
- ✓ All activation functions must be continuous.

[5 points] Which of the following are true of the Nadaraya-Watson estimator?

- ○ It only requires some of the training data at test time.
- √ **It is a nonparametric method.**
- √ **It estimates a nonlinear function of the input.**
- ○ It estimates a linear function of the input.

[5 points] What does the learning rate control in Stochastic Gradient Descent?

- √ **The size of gradient steps made in each iteration.**
- ○ The degree of nonlinearity in the model.
- ○ The number of iterations per epoch.
- √ **The speed at which the model learns.**

$$\theta_{i+1} = \theta_i - \eta \nabla_{\theta_i} L$$

[5 points] Which of the following models are nonparametric?

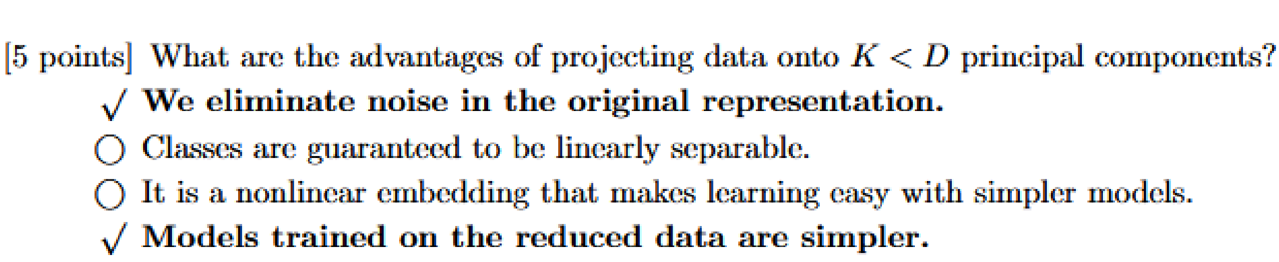
- ○ The Multilayer Perceptron (MLP).
- ○ Logistic regression.
- √ **The K-Nearest Neighbor Classifier**
- ○ Decision Trees.

[5 points] Which of the following are causes of the vanishing gradients when training neural networks?

- √ **Saturated inputs to activation functions with near-zero derivatives when saturated.**
- ○ Badly scaled input values.
- √ **Very deep models.**
- ○ Bad random initialization of the network parameters.

[5 points] What do residual connections in a Deep Neural Network do?

- √ **They help mitigate the problem of vanishing gradients.**
- √ **They make training deeper models possibile.**
- ○ They introduce additional nonlinear activations in the network.
- ○ None of the above



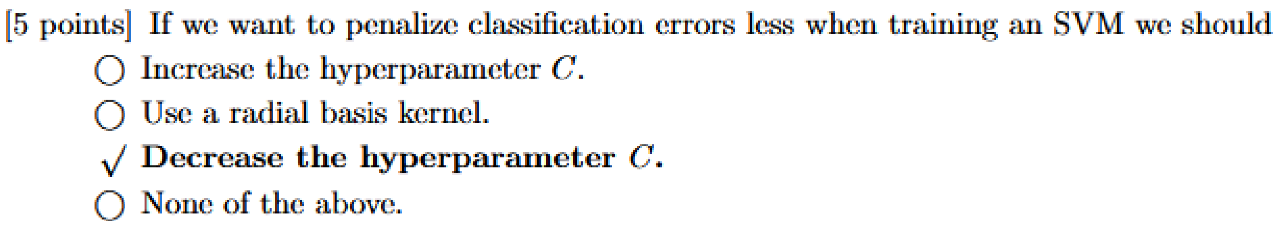
[5 points] What are the advantages of projecting data onto $K < D$ principal components?

✓ **We eliminate noise in the original representation.**

○ Classes are guaranteed to be linearly separable.

○ It is a nonlinear embedding that makes learning easy with simpler models.

✓ **Models trained on the reduced data are simpler.**



[5 points] If we want to penalize classification errors less when training an SVM we should

○ Increase the hyperparameter C.

○ Use a radial basis kernel.

✓ **Decrease the hyperparameter C.**

○ None of the above.

[5 points] How many parameters will a Convolutional Layer with Cout convolutions of size 3x3 have if it takes an input with Cin feature maps?
( ) 9 x Cin + Cout   [✓ ( ) 9 x Cin x Cout + Cout]   ( ) 9 x Cout + Cin   ( ) None of the above

[5 points] Consider one layer of weights (edges) in a convolutional neural network (CNN) for grayscale images, connecting one layer of units to the next layer of units. Which type of layer has fewer parameters to be learned during training?
( ) A convolutional layer with 10 3 x 3 filters. (100)
[✓ ( ) A max-pooling layer that reduces a 10 x 10 image to 5 x 5.] (0)
( ) A convolutional layer with 8 5 x 5 filters. (208)
( ) A fully-connected layer from 20 hidden units to 4 output units. (84)

[5 points] How many iterations of gradient descent must we perform for an epoch of minibatch Stochastic Gradient Descent with a dataset of N samples and a batch size of B?
[✓ ( ) N/B]   ( ) N   ( ) B x N   ( ) N – B

[5 points] Which of the following are true of the vanishing gradient problem for networks using sigmoid activation functions?
[✓ ( ) Deeper neural networks tend to be more subsceptible to vanishing gradients.]
( ) Networks with sigmoid units don't have this problem if they're trained with the cross-entropy loss function.
[✓ ( ) Using ReLU units instead of sigmoid untis can reduce this problem.]
( ) None of the above.

[5 points] Which of the following are true of convolutional neural networks (CNNs) for image analysis?
[✓ ( ) Pooling layers reduce the spatial resolution of the image.]
( ) They have more parameters than fully connected networks with the same number of layers and the same number of neurons in each layer.
( ) A CNN can be trained for unsupervised learning tasks, whereas an ordinary neural net cannot.
[✓ ( ) Filters in each layer tend to detect low-level features like edges and blobs.]

[5 points] In neural networks, nonlinear activation functions such as a sigmoid, tanh and ReLU
( ) speed up the gradient calculation in backpropagation, as compared to linear units.
( ) are applied only to the output units.
[✓ ( ) help to lear nonlinear decision boundaries.]
( ) always output values between 0 and 1.

[5 points] The numerical output of a sigmoid activation in a neural network
( ) is unbounded, encompassing all real numbers.
( ) is unbounded above, encompassing all non-negative real numbers.
( ) is bounded between 0 and 1.
( ) is bounded between -1 and 1.

[5 points] Which of the following are true of the Nadaraya-Watson estimator?
( ) It requires estimating D + 1 parameters, where D is the dimensionality of the inputs.
( ) It estimates a nonlinear function of the input.
( ) It requires all of the training data at test time.
( ) It estimates a polynomical function of the input.

[5 points] If, when training an MLP, the training error goes down and converges to a local minimum, but when testing on new data the test error is very high: What is probably going wrong and what can you do to fix it?
( ) Use the same training data but add two more hidden layers.
( ) The training data size is not large enough, collect more training data and retrain it.
( ) Use a different initialization and train the network several times. Use the average of predictions from all nets to predict test data.
( ) Play with learning rate and add regulatization term to the objective function.

[5 points] Which of the following is true about the K-means clustering algorithm?
( ) It is a supervised learning algorithm.
( ) It adjusts the number of clusters K.
( ) It will always terminate for any K.
( ) None of the above.

[5 points] Which of the following will increase variance in a deep neural network?
( ) Adding more hidden layers.
( ) Adding L2 regularization term to the loss.
( ) Removing hidden layers.
( ) Increasing the number of units in hidden layers.

[15 points] Show that a Committee Ensemble model using N bootstrapped linear regression models is a linear regression (i.e. that can be expressed as $\mathbf{w}^T\mathbf{x} + b$ for some $\mathbf{w}$ and b).

Note: Be sure to state all assumptions you make in your answer.

Solution: A committee model with N bootstrapped linear regression models has this form:

$$f(\mathbf{x};\theta) = \frac{1}{N}\sum_{n=1}^{N} \mathbf{w}_n^T\mathbf{x} + b_n$$

for $\theta = (\mathbf{w}_n, b_b)_{n=0}^{N}$. But then by linearity and commutativity of inner products we have:

$$\begin{aligned}
f(\mathbf{x};\theta) &= \frac{1}{N}\sum_{n=1}^{N} \left[\mathbf{w}_n^T\mathbf{x} + b_n\right] \\
&= \frac{1}{N}\sum_{n=1}^{N} \mathbf{w}_n^T\mathbf{x} + \frac{1}{N}\sum_{n=1}^{N} b_n \quad \text{(by linearity)} \\
&= \underline{\frac{1}{N}}\mathbf{x}^T\sum_{n=1}^{N} \mathbf{w}_n + \frac{1}{N}\sum_{n=1}^{N} b_n \quad \text{(by commutativity of inner product)} \\
&= \cancel{\frac{1}{N}}\hat{\mathbf{w}}^T\mathbf{x} + \hat{b}
\end{aligned}$$

$\vec{w}^T\vec{x} = \vec{x}^T\vec{w}$

For the new model parameters $\hat{\theta}$:

$$\hat{\mathbf{w}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{w}_n \text{ and } \hat{b} = \frac{1}{N}\sum_{n=1}^{N} b_n$$

□

$\sigma(x) = 2X$

[15 points] Show that a Multilayer Perceptron with two hidden layers with activation function $\sigma(x) = x$ is only capable of learning linear functions.

Solution: An MLP with two hidden layers computes the function:

$$\begin{aligned}
f(\mathbf{x}) &= \sigma\big(W_{\text{out}}\sigma(W_2\sigma(W_1\mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2) + \mathbf{b}_{\text{out}}\big) \\
&= W_{\text{out}}(W_2(W_1\mathbf{x} + \mathbf{b}_1) + \mathbf{b}_2) + \mathbf{b}_{\text{out}} \quad \text{(since } \sigma \text{ is the identity function)} \\
&= (W_{\text{out}}W_2W_1)\mathbf{x} + [W_{\text{out}}W_2\mathbf{b}_1 + W_{\text{out}}\mathbf{b}_2 + \mathbf{b}_{\text{out}}],
\end{aligned}$$

(by linearity, distribute Matrix Multiplications)

which is a linear (well, affine) function $f(\mathbf{x}) = W\mathbf{x} + \mathbf{b}$ for:

$$\begin{aligned}
W &= W_{\text{out}}W_2W_1 \\
\mathbf{b} &= W_{\text{out}}W_2\mathbf{b}_1 + W_{\text{out}}\mathbf{b}_2 + \mathbf{b}_{\text{out}}.
\end{aligned}$$

□

[15 points] Show that the first principal component of dataset $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$ is an eigenvector of the data covariance matrix.

Solution: We saw this in class. Consider a dataset $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ with $\mathbf{x}_n \in \mathbb{R}^D$. We seek a subspace of dimensionality $M = 1$ in which the projected points have maximum variance. The mean of the dataset $\mathcal{D}$ projected onto a basis vector $\mathbf{u}_1$ is:

$$\begin{aligned}\frac{1}{N}\sum_{n=1}^{N}\mathbf{u}_1^T\mathbf{x}_n &= \mathbf{u}_i^T\left\{\frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\right\} \\ &= \mathbf{u}_1^T\bar{\mathbf{x}},\end{aligned}$$

where $\bar{\mathbf{x}} = N^{-1}\sum_n \mathbf{x}_n$. The variance is then:

$$\frac{1}{N}\sum_{n=1}^{N}(\mathbf{u}_1^T\mathbf{x}_n - \mathbf{u}_1^T\bar{\mathbf{x}})^2 = \mathbf{u}_1^T\mathbf{S}\mathbf{u}_1$$

$$\text{where } \mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T.$$

We cannot simply maximize this – it is unbounded. We must constrain the optimization so that $\mathbf{u}$ has unit norm:

$$\mathbf{u}_1^* = \arg\max_{\mathbf{u}}\left[\mathbf{u}_1^T S\mathbf{u}_1 + \lambda_1(1 - \mathbf{u}_1^T\mathbf{u}_1)\right]$$

Setting the gradient of the right-hand side to zero and solving, we obtain:

$$\begin{aligned}\nabla_{\mathbf{u}_1}\mathbf{u}_1^T\mathbf{S}\mathbf{u}_1 + \lambda(1 - \mathbf{u}_1^T\mathbf{u}_1) &= \mathbf{0} \\ \implies \mathbf{S}\mathbf{u}_1 &= \lambda\mathbf{u}_1,\end{aligned}$$

which tells us that $\mathbf{u}_1$ must be an eigenvector of S with eigenvalue λ. □

[10 points (bonus)] Design a Deep Convolutional Neural Network (with at least three convolutional layers and one or more pooling layers) to classify MNIST images (input size 28 × 28). Draw the network (or write pseudocode for its definition) and indicate how many parameters each layer has and the sizes of the intermediate feature maps.

Solution: I will write pseudocode in tabular form for the definition of each layer (with corresponding numbers of parameters and size of the activations:

Layer	Type	Activation Size	# Parameters
1	Input	1 × 28 × 28	0
2	Conv2D(32, 1, 3, 3)	32 × 26 × 26	320 (32 * 3 * 3 + 32)
3	ReLU	32 × 26 × 26	0
4	Conv2D(32, 32, 3, 3)	32 × 24 × 24	9248
5	ReLU	32 × 2~~6~~4 × 2~~6~~4	0
6	MaxPool(2, 2)	32 × 1~~3~~2 × 1~~3~~2	0
7	Conv2D(16, 32, 3, 3)	16 × 1~~1~~0 × 1~~1~~0	4624
8	ReLU	16 × 1~~1~~0 × 1~~1~~0	0
9	Conv2D(16, 16, 3, 3)	16 × ~~9~~8 × ~~9~~8	2320
10	ReLU	16 × ~~9~~8 × ~~9~~8	0
11	MaxPool(2, 2)	16 × ~~5~~4 × ~~5~~4	0
12	Flatten()	~~400~~ 16^2	0
13	Linear(~~400~~ 16^2, 128)	128	~~51328~~ 32,768
14	ReLU	128	0
15	Linear(128, 64)	64	8256
16	ReLU	64	0
17	Linear(64, 10)	10	650

# Convolution

1 conv kernel

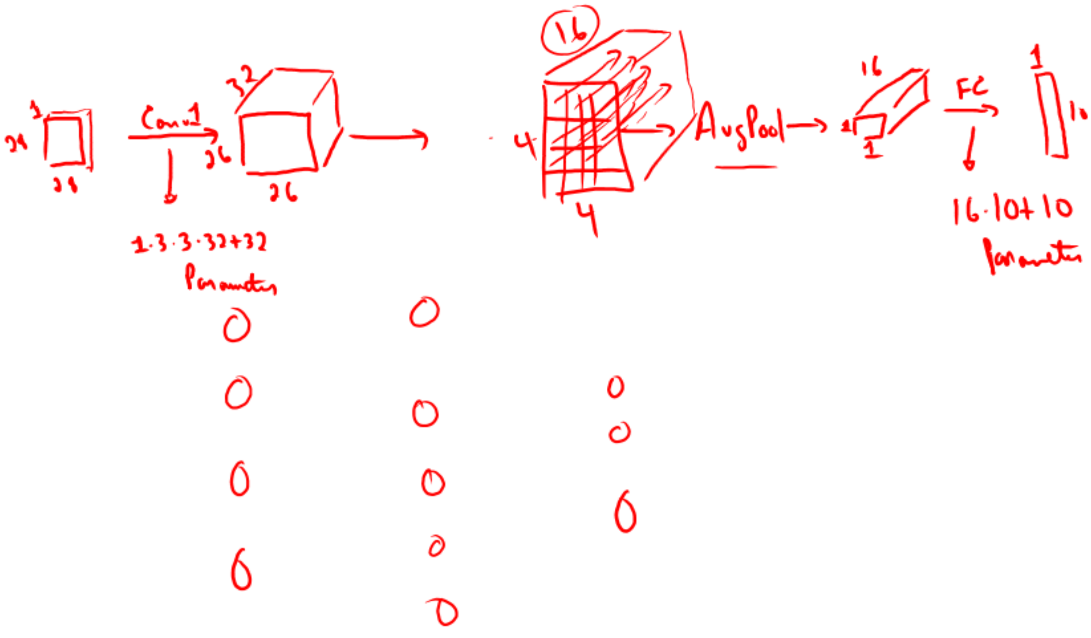
bias

$\underbrace{32 \cdot 3 \cdot 3}_{1 \text{ Conv}} \cdot 32 + 32$

$\underbrace{32) \cdot 3 \cdot 3}_{1 \text{ conv}} \cdot 16 + 16$

Avg Pool

AlexNet



3. Designing an MLP for classification.

(a) [5 points] Design a Multilayer Perceptron (MLP) with an input layer of size 3, two hidden layers with 5 units each, and a signle output unit. Indicate the parameters of each layer and how many parameters each layer has.

Input layer: 3 units

First hidden layer: 5 units

Second hidden layer: 5 units

Output layer: 1 unit

Parameters:

Input layer → first hidden layer: 3 x 5 weights + 5 biases = 20 parameters
First hidden layer → second hidden layer: 5 x 5 weights + 5 biases = 30 parameters
Second hidden layer → output layer: 5 x 1 weights + 1 bias = 6 parameters

for a total of 20 + 30 + 6 = 56 parameters.

(b) [5 points] Write the expression for the function this MLP computes using the parameters indicated in part (a).

Note: $f(x) = Wx + b$

$f(x) = \sigma_{out}(W_{out}\,\sigma(W_2\,\sigma(W_1 x + b_1) + b_2) + b_{out})$

(c) [5 points] What type of classification problems could you use this MLP for? What loss function should you use to train it?

Binary Classification, log loss;
Multiclass Classification, softmax loss;
Multilabel Classification, log loss.

[15 points] Assume you have a dataset for a 3 class classification problem and you want to train a classification model f(x,θ) parametrized by θ on it. Write the pseudocode for one iteration of batch gradient descent and pseudocode for one iteration of minibatch stochastic gradient descent (batch size B). For this model and dataset assume the negative log likelihood loss.

BGD:
1. Initialize the parameters θ.
2. Compute the predictions:

3. Compute the negative log likelihood loss:

4. Compute the gradients of the loss with respect to θ:

5. Average the gradients:

6. Update the parameters θ:

Mini-Batch SGD:
1. Initialize the parameters θ.
2. Shuffle the dataset D randomly.
3. For each mini-batch B_k of size B:
a.
!"
$
b. Compute the predictions:
"

c. Compute the negative log likelihood loss:

d. Compute the gradients of the loss with respect to θ:
"

e. Average the gradients over the mini-batch:
"

f. Update the parameters θ:

[10 points] Assume you have a lineraly separable dataset for binary classification problem. Show that any solution minimizing the hinge loss is guaranteed to satisfy the constraints of the hard-margin primal SVM objective.

Optimal hard-margin SVM problem:
subject to for .

Total hinge loss to minimize:
! " #

1. For a linearly separable dataset, we know there exists a hyperplane such that all points can be perfectly classified with some margin. That is, there exists , such that for all we have:

$ " % " will be minimized when is satisfied.

2. Now consider the minimization of the hinge loss. Since the hinge loss is minimized when for all , the optimization problem will try to push as far above 1 as possible. This corresponds to maximizing the margin, which is the goal of the hard margin SVM objective.

Thus, minimizing the hinge loss directly encourages a solution where the decision boundary (defined by and) separates the classes with a margin of at least 1, which satisfies the constraints of the hard margin SVM problem.