Brendan Neal 1001160236|nealbre1 2018-10-31

CSC411 Homework 4

Question 1: AlexNet Analysis

1a: Counting Parameters

Formulas for Analysis

Convolutional (Conv) Layers

$$(\#\ Conv\ Units) = Product(Output\ Tensor\ Dims) \times (\#\ Copies)$$

$$(\#\ Conv\ Weights) = Product(Kernel\ Dims) \times (\#\ Kernels)$$

$$(\#\ Conv\ Connections) = Product(Kernel\ Dims) \times (\#Units)$$

Dense Layers

$$(\#Dense\ Units) = (\#\ Output\ Dimensions)$$

$$(\#Dense\ Weights) = (\#Dense\ Connections) = (\#\ Input\ Units) \times (\#\ Output\ Units)$$

Layer	Units	Weights	Connections
Conv 1	$(55 \times 55 \times 48) \times 2 = 290{,}400$	$(11 \times 11 \times 3) \times 96 = 34{,}848$	$(11 \times 11 \times 3) \times 145{,}200 = 105{,}415{,}200$
Conv 2	$(27 \times 27 \times 128) \times 2 = 186{,}624$	$(5 \times 5 \times 48) \times 256 = 307{,}200$	$(5 \times 5 \times 48) \times 186{,}624 = 223{,}948{,}800$
Conv 3	$(13 \times 13 \times 192) \times 2 = 64{,}896$	$(3 \times 3 \times 256) \times 384 = 884{,}736$	$(3 \times 3 \times 256) \times 64{,}896 = 149{,}520{,}384$
Conv 4	$(13 \times 13 \times 192) \times 2 = 64{,}896$	$(3 \times 3 \times 192) \times 384 = 663{,}552$	$(3 \times 3 \times 192) \times 64{,}896 = 112{,}140{,}288$
Conv 5	$(13 \times 13 \times 128) \times 2 = 43{,}264$	$(3 \times 3 \times 192) \times 256 = 442{,}368$	$(3 \times 3 \times 192) \times 43{,}264 = 74{,}760{,}192$
Full 1 (Dense)	4,096	$43{,}264 \times 4{,}096 = 177{,}209{,}344$*	
Full 2 (Dense)	4,096	$4{,}096 \times 4{,}096 = 16{,}777{,}216$	
Output (Dense)	1,000	$4{,}096 \times 1{,}000 = 4{,}096{,}000$	

*I'm ignoring max pooling that should take place between Conv 5 and Full 1: https://piazza.com/class/jlp72odwmqo2v2?cid=606

1b: Case Study

i. Cell Phone: Parameter / Weight Reduction
   a. Decrease number of dense units. For example, reducing the number of dense units to 2048 in Full 1 would cut the number of parameters by $2048 \times 43{,}264 \approx 88M$. This would reduce the representational capacity of the dense layers but drastically decrease the amount of weights.

ii. Rapid Predictions: Connection Reduction
   a. Decrease dimensionality of convolution layer outputs. Can make smaller output tensors by:
      i. Reducing number of kernels (means less output channels)
      ii. Increasing stride of kernels (means smaller output images, and less dot-products per image)
      iii. Increasing max-pooling patch size (means less output pixels)
   b. Decrease number of dense units (parameter reduction mentioned above also carries over to number of computations / connections)

Question 2: Naïve Bayes Analysis

Denote for brevity: $p(y = k) = p(y^k)$

Formula for Reference

$$p(y^k) = \alpha_k$$

$$p(x|y^k, \mu, \sigma) = \left(\prod_{i=1}^{D} 2\pi\sigma_i^2\right)^{-\frac{1}{2}} \left(\exp\left(\sum_{i=1}^{D} \frac{1}{2\sigma_i^2}(x_i - \mu_{ki})^2\right)\right)$$

Part A

Applying Bayes Theorem:

$$p(y^k|x, \mu, \sigma) = \frac{p(x|y^k, \mu, \sigma)p(y^k|\mu, \sigma)}{p(x|\mu, \sigma)}$$

Total probability can be applied to yield $p(x|\mu, \sigma)$

$$p(x|\mu, \sigma) = \sum_{j=1}^{k} p(y^j)p(x|y^j, \mu, \sigma) = \sum_{j=1}^{k} \alpha_j p(x|y^j, \mu, \sigma)$$

y_k is a prior, can simplify conditionals involving it

$$p(y^k|\mu, \sigma) = p(y^k) = \alpha_k$$

Rephrasing equation

$$p(y^k|x, \mu, \sigma) = \frac{\alpha_k p(x|y^k, \mu, \sigma)}{\sum_{j=1}^{k} \alpha_j p(x|y^j, \mu, \sigma)}$$

Part B

Note: M refers to number of dimensions as D now refers to data

$$\ell(\theta;\ D) = -(N\log\alpha_k) + \left(N\sum_{j=1}^{M}\log\sigma_j\right) - \left(\sum_{j=1}^{M}\frac{1}{2}\sigma_j^{-2}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2\right) + \left(\frac{NM}{2}\log 2\pi\right)$$

Part C

Derivative WRT σ_j (second and third terms are only ones that depend on σ_j)

$$\ell_{\sigma_j}(\theta; D) = N\sigma_j^{-1} - \sigma_j^{-3}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2$$

MLE for σ_j^2: biased variance of attribute j over ALL samples. Proof:

$$0 = N\sigma_j^{-1} - \sigma_j^{-3}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2$$

$$\Rightarrow N\sigma_j^{-1} = \sigma_j^{-3}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2$$

$$\Rightarrow N = \sigma_j^{-2}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2$$

$$\Rightarrow \sigma_j^2 = \frac{1}{N}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2$$

Derivative WRT μ_{kj} (only term that needs to be considered is the third one)

$$\ell_{\mu_{kj}}(\theta; D) = \frac{d}{d\mu_{kj}}\left(\sum_{j=1}^{M}\frac{1}{2}\sigma_j^{-2}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2\right)$$

j (feature) is fixed if we are computing in terms of μ_{kj} – can drop the outer sum for simplicity

$$\Rightarrow \ell_{\mu_{kj}}(\theta; D) = \frac{d}{d\mu_{kj}}\left(\frac{1}{2}\sigma_j^{-2}\sum_{i=1}^{N}\left(x_j^{(i)} - \mu_{kj}\right)^2\right)$$

Use chain rule to compute derivative, and binary equality function to "turn off" terms whose class isn't k (their means are different):

$$\Rightarrow \ell_{\mu_{kj}}(\theta; D) = \frac{1}{2}\sigma_j^{-2}\sum_{i=1}^{N}1\{y^{(i)} = k\}\left(-2\left(x_j^{(i)} - \mu_{kj}\right)\right)$$

$$\Rightarrow \ell_{\mu_{kj}}(\theta; D) = \sigma_j^{-2}\sum_{i=1}^{N}1\{y^{(i)} = k\}\left(\mu_{kj} - x_j^{(i)}\right)$$

MLE for μ_{kj} (mean for j-th attribute for class k) – average of attribute j, for all samples from class k. Proof:

$$0 = \sigma_j^{-2}\sum_{i=1}^{N}1\{y^{(i)} = k\}\left(x_j^{(i)} - \mu_{kj}\right)$$

$$\Rightarrow 0 = \sum_{i=1}^{N}1\{y^{(i)} = k\}\left(x_j^{(i)}\right) - \sum_{i=1}^{N}1\{y^{(i)} = k\}\left(\mu_{kj}\right)$$

$$\Rightarrow \sum_{i=1}^{N} 1\{y^{(i)} = k\}(\mu_{kj}) = \sum_{i=1}^{N} 1\{y^{(i)} = k\}(x_j^{(i)})$$

$$\Rightarrow \mu_{kj} \sum_{i=1}^{N} 1\{y^{(i)} = k\} = \sum_{i=1}^{N} 1\{y^{(i)} = k\}(x_j^{(i)})$$

$$\Rightarrow \mu_{kj} = \frac{\sum_{i=1}^{N} 1\{y^{(i)} = k\}(x_j^{(i)})}{\sum_{i=1}^{N} 1\{y^{(i)} = k\}}$$

Part D

I was unable to complete this question.