

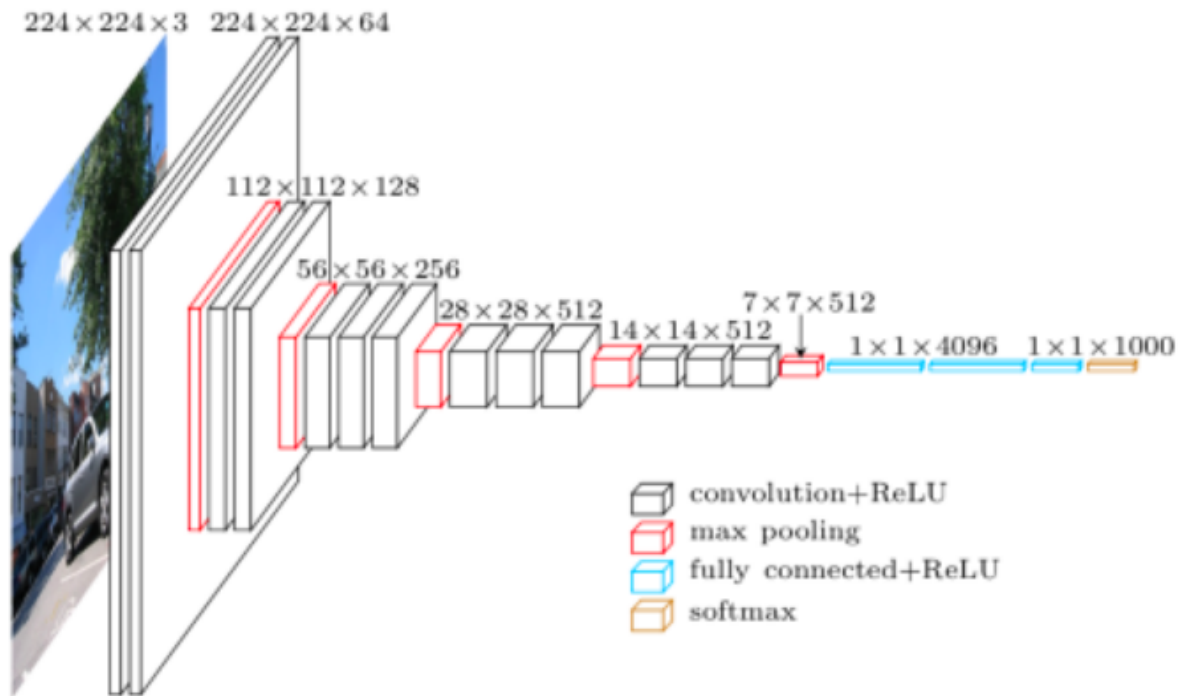
TTIC 31230, Fundamentals of Deep Learning

David McAllester, Autumn 2020

The Fundamental Equations of Deep Learning

What is a Deep Network?

VGG, Zisserman, 2014



Davi Frossard

138 Million Parameters

What is a Deep Network?

We assume some set \mathcal{X} of possible inputs, some set \mathcal{Y} of possible outputs, and a parameter vector $\Phi \in \mathbb{R}^d$.

For $\Phi \in \mathbb{R}^d$ and $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ a deep network computes a probability $P_\Phi(y|x)$.

The Fundamental Equation of Deep Learning

We assume a “population” probability distribution Pop on pairs (x, y) .

$$\Phi^* = \underset{\Phi}{\operatorname{argmin}} E_{(x,y) \sim \text{Pop}} - \ln P_{\Phi}(y|x)$$

This loss function $\mathcal{L}(x, y, \Phi) = -\ln P_{\Phi}(y|x)$ is called **cross entropy loss**.

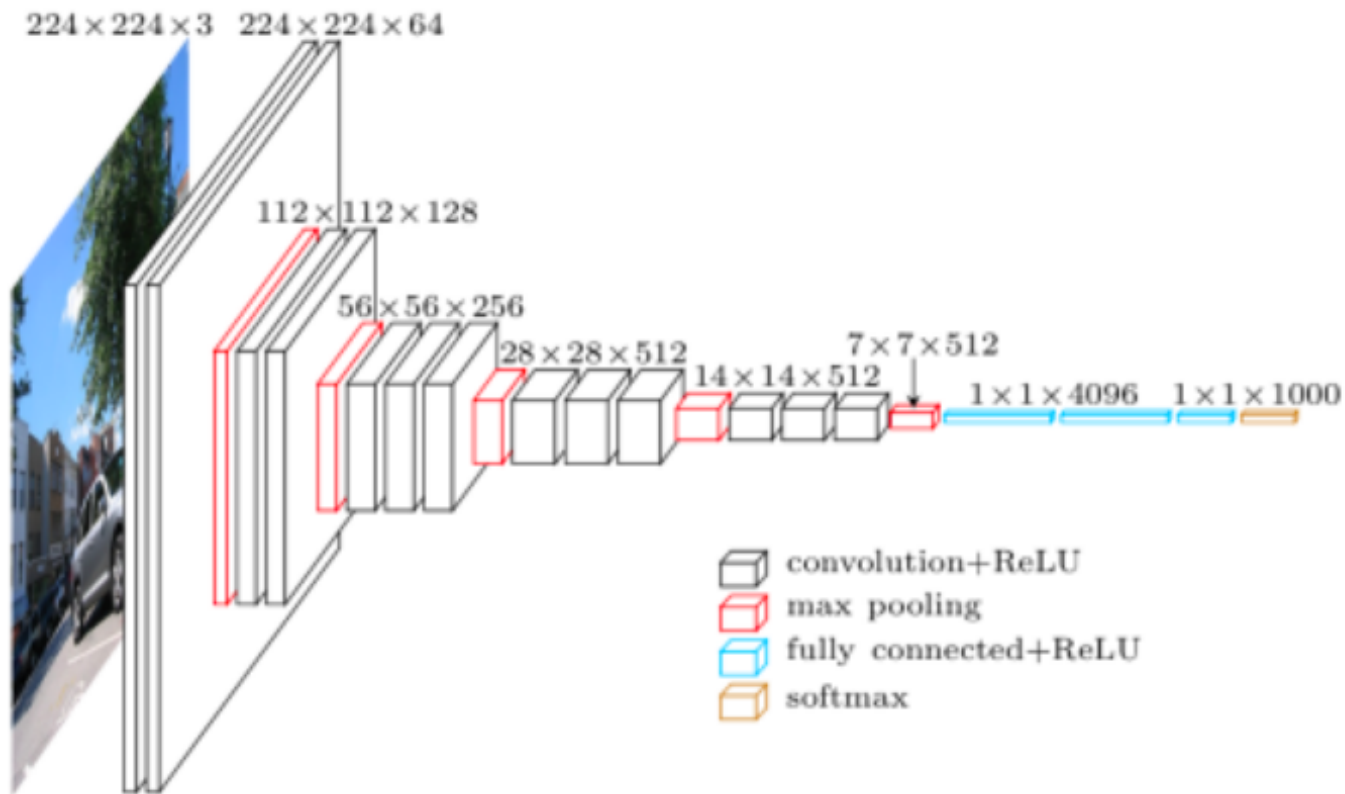
A Second Fundamental Equation

Softmax: Converting Scores to Probabilities

We start from a “score” function $s_{\Phi}(y|x) \in \mathbb{R}$.

$$\begin{aligned} P_{\Phi}(y|x) &= \frac{1}{Z} e^{s_{\Phi}(y|x)}; \quad Z = \sum_y e^{s_{\Phi}(y|x)} \\ &= \operatorname{softmax}_y s_{\Phi}(y|x) \end{aligned}$$

Note the Final Softmax Layer



Davi Frossard

How Many Possibilities

We have $y \in \mathcal{Y}$ where \mathcal{Y} is some set of “possibilities”.

Binary: $Y = \{-1, 1\}$

Multiclass: $Y = \{y_1, \dots, y_k\}$ k manageable.

Structured: y is a “structured object” like a sentence. Here $|Y|$ is unmanageable.

Binary Classification

We have a population distribution over (x, y) with $y \in \{-1, 1\}$.

We compute a single score $s_\Phi(x)$ where

for $s_\Phi(x) \geq 0$ predict $y = 1$

for $s_\Phi(x) < 0$ predict $y = -1$

Binary Classification: Softmax Cross Entropy

$$P_{\Phi}(y|x) = \operatorname{softmax}_{y \in \{-1,1\}} y s_{\Phi}(x) = \frac{1}{Z} e^{ys(x)}$$

$$= \frac{e^{ys(x)}}{e^{ys(x)} + e^{-ys(x)}}$$

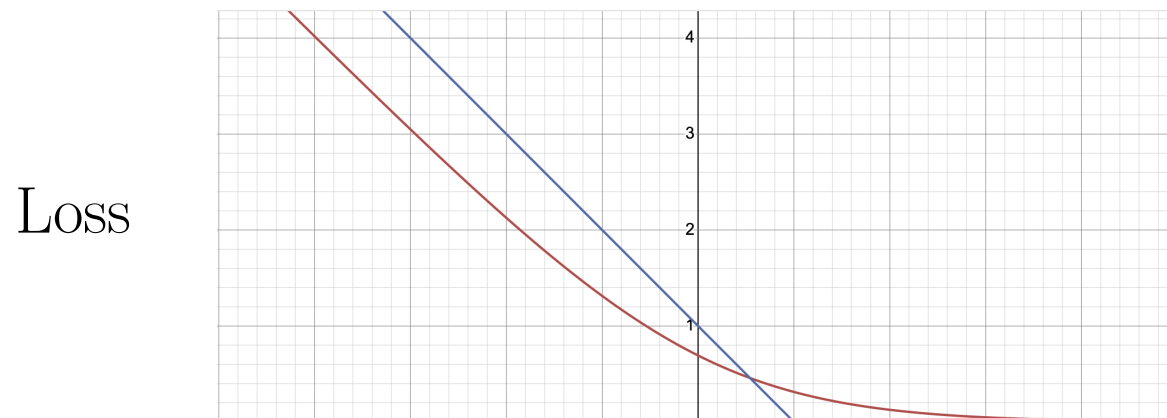
$$= \frac{1}{1 + e^{-2ys(x)}}$$

$$= \frac{1}{1 + e^{-m}} \quad m = 2ys(x) \text{ is the margin}$$

Binary Classification

Softmax Cross Entropy: $-\ln P_{\Phi}(y|x) = \ln(1 + e^{-m})$

SVM Hinge loss: $\max(0, 1 - m)$



margin $m = 2ys(x)$, $y \in \{-1, 1\}$

Image Classification (Multiclass Classification)

We have a population distribution over (x, y) with $y \in \{y_1, \dots, y_k\}$.

$$P_{\Phi}(y|x) = \underset{y}{\text{softmax}} \ s_{\Phi}(y|x)$$

$$\begin{aligned} \Phi^* &= \underset{\Phi}{\text{argmin}} \ E_{(x,y) \sim P_{\text{op}}} \mathcal{L}(x, y, \Phi) \\ &= \underset{\Phi}{\text{argmin}} \ E_{(x,y) \sim P_{\text{op}}} \textcolor{red}{- \ln P_{\Phi}(y|x)} \end{aligned}$$

Machine Translation (Structured Labeling)

We have a population of translation pairs (x, y) with $x \in V_x^*$ and $y \in V_y^*$ where V_x and V_y are source and target vocabularies respectively.

$$P_{\Phi}(w_{t+1}|x, w_1, \dots, w_t) = \underset{w \in V_y \cup \langle \text{EOS} \rangle}{\text{softmax}} s_{\Phi}(w \mid x, w_1, \dots, w_t)$$

$$P_{\Phi}(y|x) = \prod_{t=0}^{|y|} P_{\Phi}(y_{t+1} \mid x, y_1, \dots, y_t)$$

$$\begin{aligned} \Phi^* &= \underset{\Phi}{\operatorname{argmin}} E_{(x,y) \sim \text{Pop}} \mathcal{L}(x, y, \Phi) \\ &= \underset{\Phi}{\operatorname{argmin}} E_{(x,y) \sim \text{Pop}} -\ln P_{\Phi}(y|x) \end{aligned}$$

Fundamental Equation: Unconditional Form

$$\Phi^* = \operatorname{argmin}_{\Phi} E_{y \sim P_{\text{op}}} - \ln P_{\Phi}(y)$$

Summary

$$\Phi^* = \operatorname{argmin}_{\Phi} E_{(x,y) \sim \text{Pop}} - \ln P_{\Phi}(y|x)$$

$$\begin{aligned} P_{\Phi}(y|x) &= \frac{1}{Z} e^{s_{\Phi}(y|x)}; \quad Z = \sum_y e^{s_{\Phi}(y|x)} \\ &= \operatorname{softmax}_y s_{\Phi}(y|x) \end{aligned}$$

END