CS230: Lecture 2
Deep Learning Intuition

Kian Katanforoosh
Recap
Learning Process

Model = Architecture + Parameters

Input  →  Model  →  Output

Loss  ←  Gradients

Things that can change
- Activation function
- Optimizer
- Hyperparameters
- ...
Logistic Regression as a Neural Network

\[
\begin{bmatrix}
255 \\
231 \\
94 \\
142
\end{bmatrix}
\rightarrow \frac{x_1^{(i)}}{255} \rightarrow \frac{x_2^{(i)}}{255} \rightarrow \frac{x_{n-1}^{(i)}}{255} \rightarrow \frac{x_n^{(i)}}{255}
\]

\[
w^T x^{(i)} + b \rightarrow \sigma \rightarrow 0.73 \rightarrow 0.73 > 0.5 \rightarrow \text{“it’s a cat”}
\]
Multi-class

\[
\begin{pmatrix}
255 \\
231 \\
\vdots \\
94 \\
142
\end{pmatrix}
\xrightarrow{\text{image2vector}}
\begin{pmatrix}
x_1^{(i)} \\
x_2^{(i)} \\
\vdots \\
x_{n-1}^{(i)} \\
x_n^{(i)}
\end{pmatrix}
\xrightarrow{\sigma}
\begin{pmatrix}
w^T x^{(i)} + b \\
w^T x^{(i)} + b \\
\vdots \\
w^T x^{(i)} + b \\
w^T x^{(i)} + b
\end{pmatrix}
\xrightarrow{\sigma}
\begin{pmatrix}
0.12 \\
0.73 \\
0.04
\end{pmatrix}
\xrightarrow{\text{class decision}}
\begin{pmatrix}
\text{Dog?} \\
\text{Cat?} \\
\text{Giraffe?}
\end{pmatrix}
\]

0.12 < 0.5
0.73 > 0.5
0.04 < 0.5
Neural Network (Multi-class)

\[
\begin{pmatrix}
255 \\
231 \\
\vdots \\
94 \\
142
\end{pmatrix}
\xrightarrow{\text{/255}}
\begin{align*}
x_1^{(i)} & \\
x_2^{(i)} & \\
\vdots & \\
x_{n-1}^{(i)} & \\
x_n^{(i)}
\end{align*}
\xrightarrow{\sigma}
\begin{align*}
w^T x^{(i)} + b & \\
w^T x^{(i)} + b & \\
\vdots & \\
w^T x^{(i)} + b & \\
w^T x^{(i)} + b &
\end{align*}
Neural Network (1 hidden layer)

\[
\begin{pmatrix}
255 \\
231 \\
\vdots \\
94 \\
142
\end{pmatrix}
\xrightarrow{/255}
\begin{pmatrix}
x^{(i)}_1 \\
x^{(i)}_2 \\
\vdots \\
x^{(i)}_{n-1} \\
x^{(i)}_n
\end{pmatrix}
\]

Hidden layer

\begin{pmatrix}
a_1^{[1]} \\
a_2^{[1]} \\
a_3^{[1]}
\end{pmatrix}

output layer

\begin{pmatrix}
a_1^{[2]}
\end{pmatrix}

0.73

0.73 > 0.5

Cat

Kian Katanforoosh, Andrew Ng, Younes Bensouda Mourri
Technique called “encoding”
Let’s build intuition on concrete applications
Today’s outline

We will learn how to:

- Analyze a problem from a deep learning approach
- Choose an architecture
- Choose a loss and a training strategy

I. Day’n’Night classification
II. Face Recognition
III. Art generation
IV. Keyword Spotting
V. Shipping model
Day’n’Night classification (warm-up)

Goal: Given an image, classify as taken “during the day” (0) or “during the night” (1)

1. Data? 10,000 images
2. Input? Resolution? (64, 64, 3)
3. Output? y = 0 or y = 1
4. Architecture? Shallow network should do the job pretty well
5. Loss? \( L = -[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})] \)
Goal: A school wants to use Face Verification for validating student IDs in facilities (dinning halls, gym, pool ...)

1. Data?
   Picture of every student labelled with their name
   Bertrand

2. Input?
   Resolution?
   (412, 412, 3)

3. Output?
   y = 1  (it’s you)
   or
   y = 0  (it’s not you)
**Goal**: A school wants to use Face Verification for validating student IDs in facilities (dinning halls, gym, pool …)

4. **What architecture?**

Simple solution:

- Compute distance pixel per pixel
- If less than threshold then $y=1$

Issues:

- Background lighting differences
- A person can wear make-up, grow a beard…
- ID photo can be outdated
**Goal:** A school wants to use Face Verification for validating student IDs in facilities (dinning halls, gym, pool ...)

**4. What architecture?**

Our solution: encode information about a picture in a vector in a 128-d Deep Network.

![Diagram](image)

We gather all student faces encoding in a database. Given a new picture, we compute its distance with the encoding of card holder. If the distance is less than the threshold (0.4), then **y = 1**.

Kian Katanforoosh, Andrew Ng, Younes Bensouda Mourri
**Goal**: A school wants to use Face Verification for validating student IDs in facilities (dining hall, gym, pool …)

### 4. Loss? Training?

We need more data so that our model understands how to encode:

- Use public face datasets

**What we really want:**

- similar encoding
- different encoding

**So let's generate triplets:**

- anchor
- positive
- negative

- minimize encoding distance
- maximize encoding distance

Kian Katanforoosh, Andrew Ng, Younes Bensouda Mourri
Recap: Learning Process

Model = Architecture + Parameters

Input

Output

\[
\begin{align*}
L &= ||Enc(A) - Enc(P)||_2^2 \\
&\quad - ||Enc(A) - Enc(N)||_2^2 \\
&\quad + \alpha 
\end{align*}
\]

Loss

Gradients

Enc(A)  Enc(P)  Enc(N)

Kian Katanforoosh, Andrew Ng, Younes Bensouda Mourri
Face Recognition

**Goal**: A school wants to use Face Identification for recognize students in facilities (dinning hall, gym, pool …)

K-Nearest Neighbors

**Goal**: You want to use Face Clustering to group pictures of the same people on your smartphone

K-Means Algorithm

Maybe we need to detect the faces first?
Art generation (Neural Style Transfer)

**Goal**: Given a picture, make it look beautiful

1. **Data?**
   - Let’s say we have any data

2. **Input?**
   - content image
   - style image

3. **Output?**
   - generated image

---

Leon A. Gatys, Alexander S. Ecker, Matthias Bethge: A Neural Algorithm of Artistic Style, 2015

Kian Katanforoosh, Andrew Ng, Younes Bensouda Mourri
4. Architecture?

We want a model that understands images very well.

We load an existing model trained on ImageNet for example.

When this image is forward propagated, we can get information about its content & its style by inspecting the layers.

5. Loss?

Content$_C$

Style$_S$
Art generation (Neural Style Transfer)

Correct Approach

\[ L = \|\text{Content}_c - \text{Content}_G\|_2^2 + \|\text{Style}_s - \text{Style}_G\|_2^2 \]

We are not learning parameters by minimizing \( L \). We are learning an image!
Goal: Given an audio speech, detect the word “lion”.

1. **Input?**

![Audio Signal]

2. **Output?**

   \[ y = 0 \text{ (there is “lion”) } \quad \text{or} \quad y = 1 \text{ (there isn’t “lion”) } \]

   \[ y = (0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0) \]

3. **Data?**

   Many audio recordings (“words”)
Speech recognition: Keyword Spotting

**Goal:** Given an audio speech, detect the word “lion”.

4. **What architecture?**

\[ y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \ldots, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, \ldots, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \]

\[
L = \|\text{Enc}(A) - \text{Enc}(P)\|_2^2 - \|\text{Enc}(A) - \text{Enc}(N)\|_2^2 + \alpha
\]

\[ y = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \ldots, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \]

Threshold: 0.6

Kian Katanforoosh, Andrew Ng, Younes Bensouda Mourri
App implementation
Server-based or on-device?

Server-based

On-device

Model Architecture + Learned Parameters

y = 0
Server-based or on-device?

**Server-based**
- App is light-weight
- App is easy to update

**On-device**
- Faster predictions
- Works offline
Duties for next week

For Tuesday 04/17, 9am:

C1M3
• Quiz: Shallow Neural Networks
• Programming Assignment: Planar data classification with one-hidden layer

C1M4
• Quiz: Deep Neural Networks
• Programming Assignment: Building a deep neural network - Step by Step
• Programming Assignment: Deep Neural Network Application

Project
• For this Friday (04/13): find teammate and submit the Team-members form with your project category
• Fill-in AWS Form to get GPU credits

This Friday (04/13):
• (optional) Project section: How to get started with your projects?