Week 8 Reflections

Course	CS 598 - Deep Learning for Healthcare
Submitted By	Rohit Bhagwat (rohitb2@illinois.edu)

Questions

What are the main messages you learned from this chapter?

This week was focussed around discussing Autoencoders. Autoencoder is an unsupervised and nonlinear dimensionality reduction model. It is widely used in many healthcare applications.

The main idea of an autoencoder is -

- It first maps the input to an internal representation via an encoder
- It then maps the internal representation back to the input space through a decoder

The composition of encoder and decoder is called the reconstruction function. The autoencoder's objective tries to minimize the reconstruction loss, thus allowing the autoencoder to focus on essential properties of the data while reducing the dimensionality.

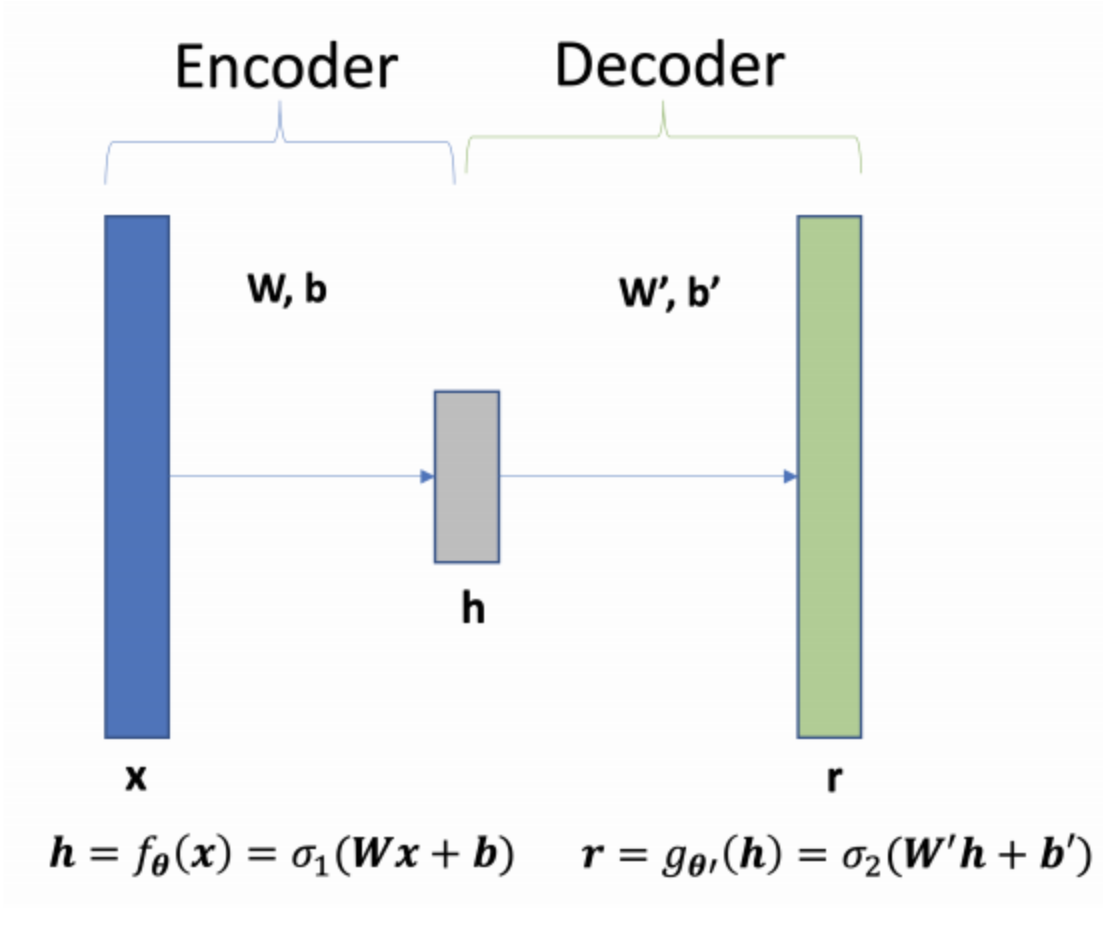


Fig 1: Autoencoder Model Illustration (From the chapters)

Variants of Autoencoder-

- **Sparse Autoencoders**: Aim at imposing a sparsity constraint on the hidden layer of the Autoencoders
- **Denoising Autoencoders**: Denoising autoencoders can learn more robust representation against noisy input. The idea is to add noises to the original input to obtain corrupted version and then try to reconstruct clean uncorrupted input from noisy input.
- **Stacked Autoencoders**: Multiple autoencoders are stacked together such that outputs of each layer are used as inputs of the next layer

What related resources (book, paper, blog, link) do you recommend your classmates to checkout?

Some great references, that also proved useful when working on the Homeworks -

Understanding Autoencoders. (Part I)

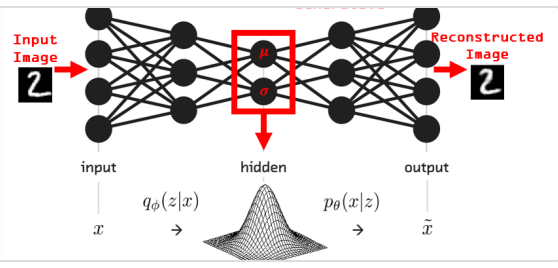
These days I've made research on autoencoders explanation and almost all of those articles are not trying to explain why we should use autoencoders, what they used for and all those articles were about only describing what autoencoders are. In my opinion, this is not a good kickstart for

https://medium.com/ai%C2%B3-theory-practice-business/understanding-autoencoders-part-i-116ed2272d35

Introduction to Autoencoders

In today's article, we are going to discuss a neural network architecture called autoencoders. This article is aimed at Machine Learning and Deep Learning beginners who are interested in getting a brief understanding of the underlying concepts behind autoencoders. So let's dive in

https://medium.com/swlh/introduction-to-autoencoders-56e5d60dad7f



Autoencoder Explained

How does an autoencoder work? Autoencoders are a type of neural network that reconstructs the input data its given. But we don't care about the output, we ca...

https://www.youtube.com/watch?v=H1AllrJ-_30

Which part do you want to improve in this chapter?

In the lectures as well as in the chapter, we are introduced to the concepts very well. The author and the professor walked through the idea of autoencoders.

I did struggle with completing assignments, perhaps a walkthrough of an example of PyTorch would prove to be useful for students.

Although I think the exercises will help students get that understanding, it just needs more research.

What are the main difference between autoencoder and principal component analysis?

Both Principal Component Analysis (PCA) and Autoencoders are both algorithms used for dimensionality reduction.

An autoencoder is a neural network that aims to reconstruct the input data
via a nonlinear dimensionality reduction.

PCA is essentially a linear transformation. PCA features are totally linearly uncorrelated with each other since features are projections onto the orthogonal basis. But autoencoded features might have correlations since they are just trained for accurate reconstruction.

A single layer auto encoder with linear transfer function is nearly equivalent to PCA!

> 💡 Reference: https://towardsdatascience.com/autoencoders-vs-pca-when-to-use-which-73de063f5d7

What is the main difference between autoencoder and denoising autoencoder?

Denoising autoencoder is a variant of Autoencoders- the idea for a denoising autoencoder is to learn a robust representation against noise.

The way it is done is by introducing some noise into the input- the autoencoder can overcome the noise and still reconstruct something close to the original input without denoise. The figure below represents the steps involved.

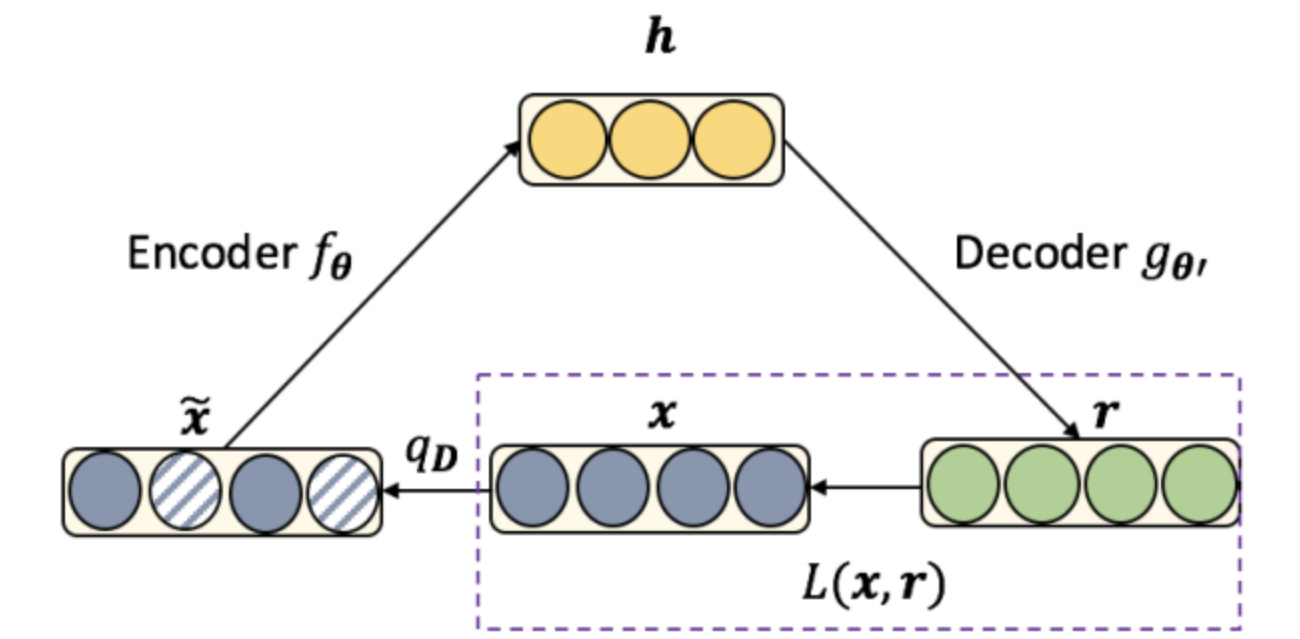


Fig 2: Denoising Autoencoder example