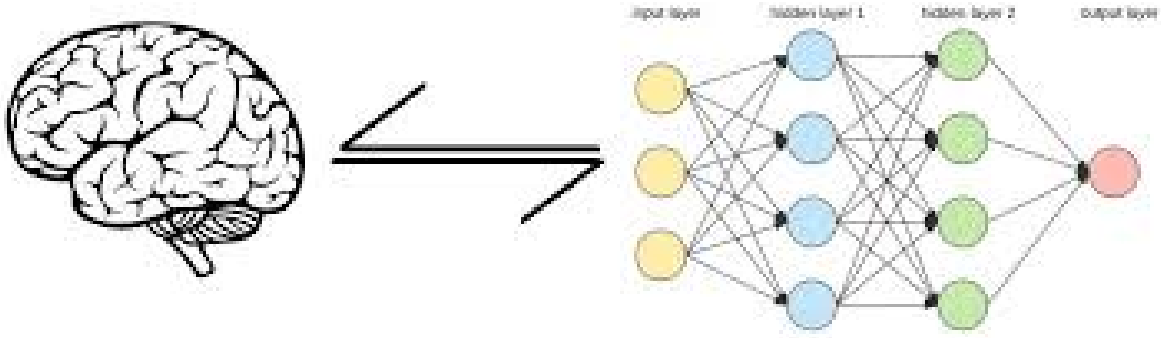
Learning Bijections

Can Duyar

Advisor: Dr. Zafeirakis ZAFEIRAKOPOULOS

10 November 2021

Project Definition



My project is related with training a neural network to act as a bijection. I am going to pick two sets of combinatorial objects that are known to have the same cardinality and train a network to act as a bijection. In this case, I am going to have different output for each input.

The goal is to investigate best practices to develop a general framework in the direction of learning bijections.

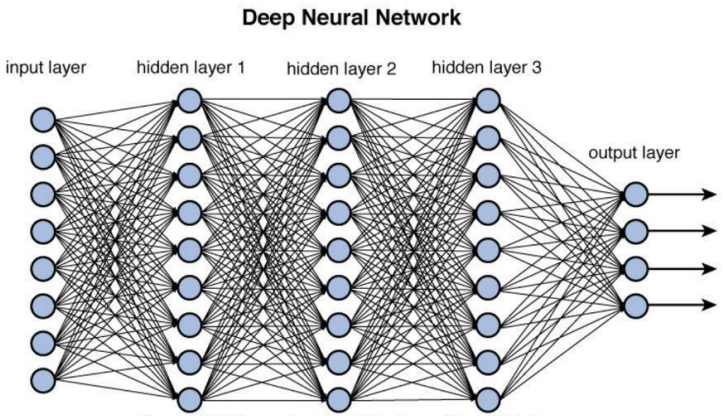
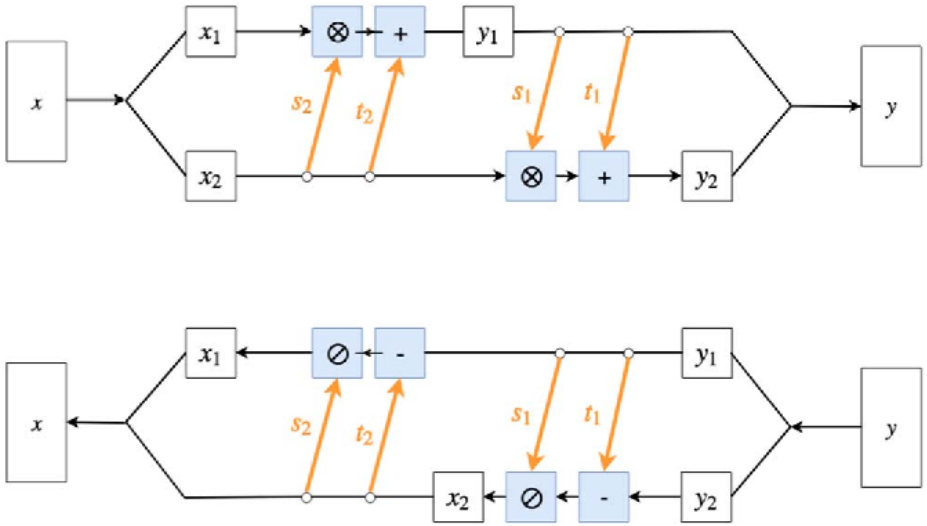


Figure 12.2 Deep network architecture with multiple layers.

What is a "Neural Network" ?

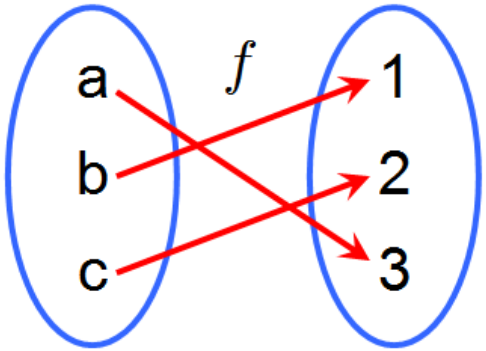
One of the main approaches to powering current ML systems is deep learning. Deep learning algorithms use the structure of the neural network to find connections between inputs and outputs to achieve a specific purpose. The reason why it is called deep learning is that the structure of deep neural network consists of a large number of layers compared to early neural networks, more weights and biases to adjust the outputs to improve the capabilities to approximate vast amounts of complex data.

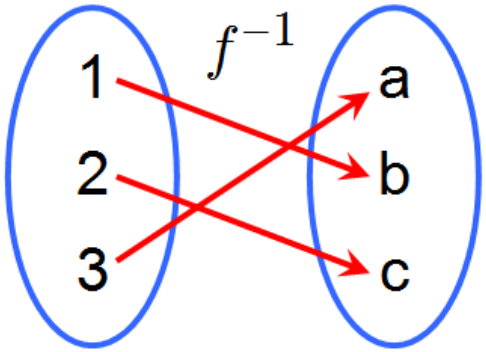
My Solution: Using an Invertible Neural Network Architecture



Invertible Neural Networks (INNs) are bijective function approximators which have a **forward mapping** $F_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^d \; x \mapsto y$ and an **inverse mapping** $F_\theta^{-1} : \mathbb{R}^d \rightarrow \mathbb{R}^d y \mapsto x$

Why "Invertible Neural Network Architecture" is appropriate for the project?





All bijective functions can be inverted and if we train a neural network act as a bijection then this network should be also inverted. Invertible network is able to accurately reconstruct the inputs to a layer from its outputs so it is an appropriate architecture for solving this problem.

Completed Parts

learning_bijections_dataset.csv

```
Set1,Set2,Bijection
1,2,(1 ,  2)
3,4,(3 ,  4)
5,6,(5 ,  6)
7,8,(7 ,  8)
9,10,(9 ,  10)
11,12,(11 ,  12)
13,14,(13 ,  14)
15,16,(15 ,  16)
17,18,(17 ,  18)
19,20,(19 ,  20)
21,22,(21 ,  22)
23,24,(23 ,  24)
25,26,(25 ,  26)
27,28,(27 ,  28)
29,30,(29 ,  30)
31,32,(31 ,  32)
33,34,(33 ,  34)
35,36,(35 ,  36)
37,38,(37 ,  38)
39,40,(39 ,  40)
41,42,(41 ,  42)
43,44,(43 ,  44)
45,46,(45 ,  46)
47,48,(47 ,  48)
49,50,(49 ,  50)
```

- I wrote a script that generates a data set that contains two sets with 100.000 elements and one of the possible bijections between them. (I used Python to create the data set)
- I researched possible architectures(like i-RevNet and i-ResNet) to construct an invertible network.
- I learn Julia to implement the model because the implementation of neural network will be in Julia language.

References I

- Lynton Ardizzone, Jakob Kruse, Sebastian Wirkert, Daniel Rahner, Eric W. Pellegrini, Ralf S. Klessen, Lena Maier-Hein, Carsten Rother, and Ullrich Köthe, *Analyzing inverse problems with invertible neural networks*, 2019.
- Jens Behrmann, Will Grathwohl, Ricky T. Q. Chen, David Duvenaud, and Jörn-Henrik Jacobsen, *Invertible residual networks*, 2019.
- Jörn-Henrik Jacobsen, Arnold Smeulders, and Edouard Oyallon, *i-revnet: Deep invertible networks*, 2018.
- Han Zhang, *Invariance and invertibility in deep neural networks*, 2020.