Explanatory Notes for 6.390

Shaunticlair Ruiz (Current TA)

Fall 2022

Neural Network Perspectives: Classification and Regression

In this class, we **won't** focus on the brain analogy, though it did inspire the model.

Instead, we will mostly think of **neural networks** in terms of what they're able to do, and how they work.

One problem we have struggled with is certain tasks that can't be handled by **linear** models. We have used **feature representations** to work on this problem.

Simply, some problems are outside our **hypothesis** space. But, there's another way: this is where **neural networks** come in.

By combining lots of simple **units** ("neurons"), we can get a very **complex** model for solving our problems.

With such a **rich** hypothesis class, combined with the power of **gradient descent**, we can create a model that can do **classification** or **regression** for very difficult problems!

Reminder: "richness" or "expressiveness" of a hypothesis reflect how wide our options are. Neural networks give us many possibilities for models. With more options, we can handle more problems!

Concept 1

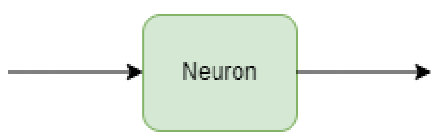
Neural Networks can create a very **rich hypothesis class** by combining many simple **units**.

With this **hypothesis class**, we can handle **regression** or **classification** for very challenging **problems**.

Building up a basic neural network

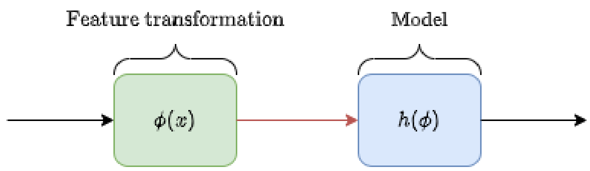
Let's make sense of what we said above, and **visualize** what a neural network might look like.

We start with one function: a **neuron**. This function could be, for example, one we've used before: our logistic **classifier**, or linear **regression**. We'll ignore the details for now.



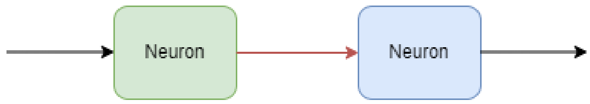
One neuron might not be very powerful, or **expressive**. It's useful, but limited. We've seen its weaknesses.

We could try to use **feature transformations** to help us. But, let's think in a more **general** way: a transformation is just another **function** we apply to our input!



This gives us an **idea**: rather than trying to think of a single, more **complex** model, we could **combine** multiple simple models!

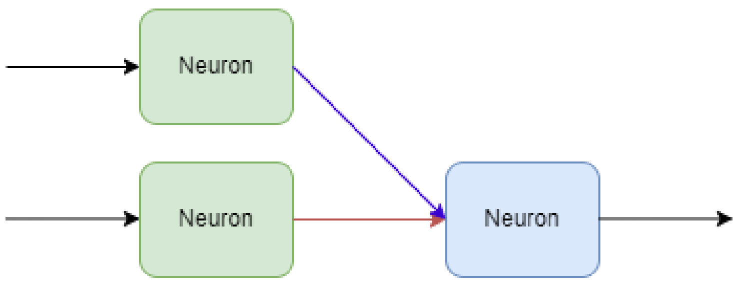
Note that feature transformations are a bit complex for what we'd usually put in a neuron. But, it gives us the right inspiration.



We could repeatedly add more neurons in **series**: each one being the input to another. And we'll do that later!

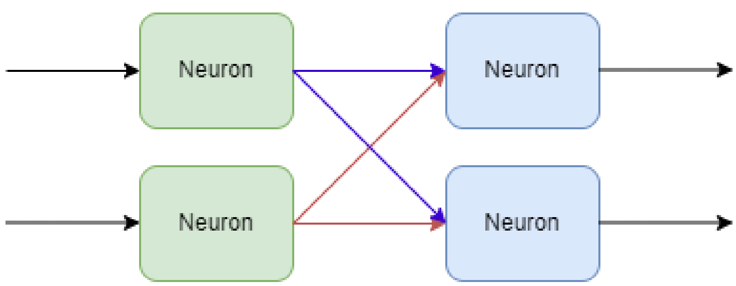
But, there's another type of **complexity** we haven't explored: we could have two neurons in **parallel**.

This parallel/series vocabulary is borrowed from circuits. We'll just use it for demonstration: you don't need to remember it.



Now, we have **two** neurons feeding into one output neuron! This already looks like a more **complicated** model.

We can go even further: what if we have two outputs as well?



Because we had two **inputs**, we had to add two new **links** when we added the output neuron. This is getting difficult to **view**!

We'll stop here for now, but you can imagine repeatedly **adding** more neurons in **parallel** (with the same inputs/outputs) or in **series** (as an input or output).

And we each addition, the function gets more and more **complex**: you can create a **richer** hypothesis class!

We'll explore how to do this **systematically** later in the chapter.

By "systematic", we just mean "in a way that's consistent and makes sense".

Definition 2

Neural Networks are a **class** of models that can be used to solve **classification**, **regression**, or other interesting problems.

They create very **rich** hypothesis classes by combining many **simple** models, called **neurons**, into a **complex** model.

We do this combination **systematically**, so that it is easy to **analyze** and work with our **model**.

This creates a very **flexible** hypothesis, which can be **broken down** into its **simple** parts and what **connects** them.