

## Review

In this chapter, we've explained why deep learning is a subject of critical importance for the modern software engineer and taken a whirlwind tour of a number of deep architectures. In the next chapter, we will start exploring TensorFlow, Google's framework for constructing and training deep architectures. In the chapters after that, we will dive deep into a number of practical examples of deep architectures.

Machine learning (and deep learning in particular), like much of computer science, is a very empirical discipline. It's only really possible to understand deep learning through significant practical experience. For that reason, we've included a number of in-depth case studies throughout the remainder of this book. We encourage you to delve into these examples and to get your hands dirty experimenting with your own ideas using TensorFlow. It's never enough to understand algorithms only theoretically!



---

# Introduction to TensorFlow Primitives

This chapter will introduce you to fundamental aspects of TensorFlow. In particular, you will learn how to perform basic computation using TensorFlow. A large part of this chapter will be spent introducing the concept of tensors, and discussing how tensors are represented and manipulated within TensorFlow. This discussion will necessitate a brief overview of some of the mathematical concepts that underlie tensorial mathematics. In particular, we'll briefly review basic linear algebra and demonstrate how to perform basic linear algebraic operations with TensorFlow.

We'll follow this discussion of basic mathematics with a discussion of the differences between declarative and imperative programming styles. Unlike many programming languages, TensorFlow is largely declarative. Calling a TensorFlow operation adds a description of a computation to TensorFlow's "computation graph." In particular, TensorFlow code "describes" computations and doesn't actually perform them. In order to run TensorFlow code, users need to create `tf.Session` objects. We introduce the concept of sessions and describe how users perform computations with them in TensorFlow.

We end the chapter by discussing the notion of variables. Variables in TensorFlow hold tensors and allow for stateful computation that modifies variables to occur. We demonstrate how to create variables and update their values via TensorFlow.

## Introducing Tensors

Tensors are fundamental mathematical constructs in fields such as physics and engineering. Historically, however, tensors have made fewer inroads in computer science, which has traditionally been more associated with discrete mathematics and logic. This state of affairs has started to change significantly with the advent of machine