*Example 2-29. Assignment fails when shapes aren't equal*

```
>>> sess.run(a.assign(tf.zeros((3,3))))
ValueError: Dimension 0 in both shapes must be equal, but are 2 and 3 for 'Assign_3'
(op: 'Assign') with input shapes: [2,2], [3,3].
```

You can see that TensorFlow complains. The shape of the variable is fixed upon initialization and must be preserved with updates. As another interesting note, `tf.assign` is itself a part of the underlying global `tf.Graph` instance. This allows TensorFlow programs to update their internal state every time they are run. We will make heavy use of this feature in the chapters to come.

## Review

In this chapter, we've introduced the mathematical concept of tensors, and briefly reviewed a number of mathematical concepts associated with tensors. We then demonstrated how to create tensors in TensorFlow and perform these same mathematical operations within TensorFlow. We also briefly introduced some underlying TensorFlow structures like the computational graph, sessions, and variables. If you haven't completely grasped the concepts discussed in this chapter, don't worry much about it. We will repeatedly use these same concepts over the remainder of the book, so there will be plenty of chances to let the ideas sink in.

In the next chapter, we will teach you how to build simple learning models for linear and logistic regression using TensorFlow. Subsequent chapters will build on these foundations to teach you how to train more sophisticated models.

CHAPTER 3

# Linear and Logistic Regression with TensorFlow

This chapter will show you how to build simple, but nontrivial, examples of learning systems in TensorFlow. The first part of this chapter reviews the mathematical foundations for building learning systems and in particular will cover functions, continuity, and differentiability. We introduce the idea of loss functions, then discuss how machine learning boils down to the ability to find the minimal points of complicated loss functions. We then cover the notion of gradient descent, and explain how it can be used to minimize loss functions. We end the first section by briefly discussing the algorithmic idea of automatic differentiation. The second section focuses on introducing the TensorFlow concepts underpinned by these mathematical ideas. These concepts include placeholders, scopes, optimizers, and TensorBoard, and enable the practical construction and analysis of learning systems. The final section provides case studies of how to train linear and logistic regression models in TensorFlow.

This chapter is long and introduces many new ideas. It's OK if you don't grasp all the subtleties of these ideas in a first reading. We recommend moving forward and coming back to refer to the concepts here as needed later. We will repeatedly use these fundamentals in the remainder of the book in order to let these ideas sink in gradually.

## Mathematical Review

This first section reviews the mathematical tools needed to conceptually understand machine learning. We attempt to minimize the number of Greek symbols required, and focus instead on building conceptual understanding rather than technical manipulations.