Chpt4 Real-world Data Representations Using Tensors

Augest 9, 2021

Often, our raw data won't be perfectly formed for the problem we'd like to solve. Each section in this chapter will describe a data type, and each will come with its own dataset. We'll also cover tabular data, time series, and text. Starting with image data, we'll then demonstrate working with a three-dimensional array using medical data that represents patient anatomy as a volume.

In every section, we will stop where a deep learning researcher would start: right before feeding the data to model. We encourage you to keep these datasets; they will constitute excellent material for when we start learning how to train neural network models in next chapter.

4.1 Working with images

An image is represented as a collection of scalars arranged in a regular grid with a height and width. We might have a single scalar per gird point, which would be represented as a grayscale image; or multiple scalars per grid point, which would typically represent different colors, or different *features* like depth from a depth camera.

4.3 Representing tabular data

Continuous, ordinal, and categorical values:

We should be aware of three different kinds of numerical values as we attempt to make scence of our data. The first kind is *continuous* values. If you are counting or measuring something with units, it's probably a continuous value.

Next, we have ordinal data. The strict ordering we have with continuous values remains, but the fixed relationship between values no long applies.

Finally, categoraical values have neither ordering nor numerical meaning to their values. These are oftern just enumerations of possibilities assigned arbitrary numbers. Because the numerical values bear no meaning, they are said to be on a *nominal* scale.

4.3.5 When to categorize

We summarize our data mapping in a small flow chart in the following:

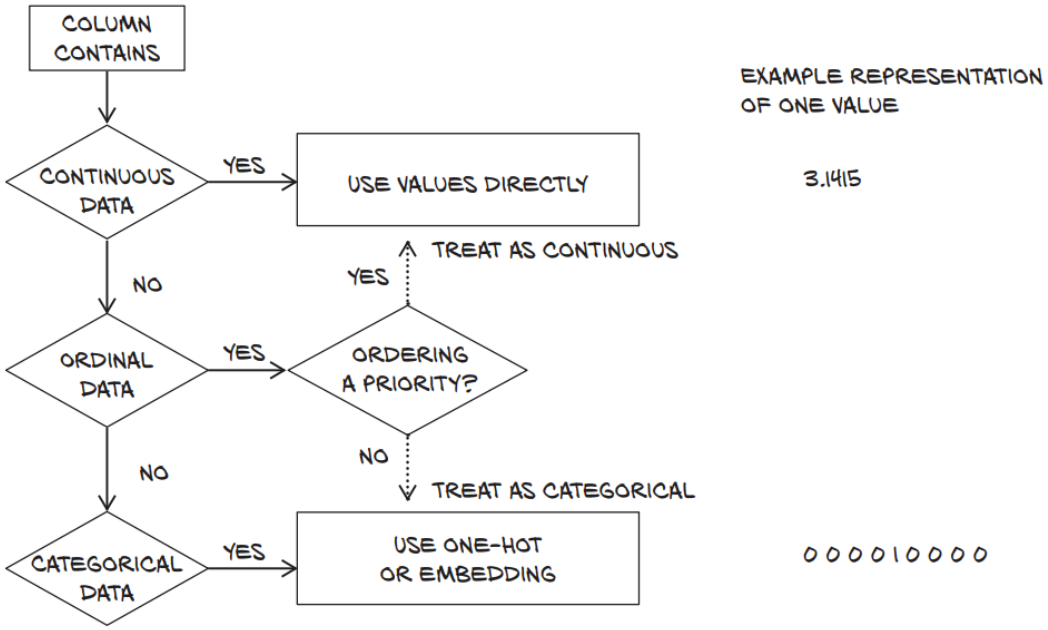

Figure 4.4 How to treat columns with continuous, ordinal, and categorical data

4.4 Working with timeseries

4.4.2 Shaping the data by time period

We might want to break up the two-year dataset into wider observation periods, like days. This way we'll have **N** *(for numbers of samples)* collections of **C** sequences of length **L**. The **C** would remain our 17 channels, while **L** would be 24: 1 per hour of the day. There's no particular reason why we must use chunks of 24 hours.

4.5 Representing text