

Working with Numerical Data

This unit focuses on **numerical data**, meaning integers or floating-point values that behave like numbers. That is, they are additive, countable, ordered, and so on.

Examples of numerical data include:

- Temperature
- Weight
- The number of deer wintering in a nature preserve

Numerical Data: How a model Ingests Data using Feature Vectors



The model ingests an array of floating-point values called a **feature vector**. You can think of a feature vector as the floating-point values comprising one example.

Feature vectors seldom use the dataset's *raw values*. Instead, you must typically process the dataset's values into representations that your model can better learn from.

You must determine the best way to represent raw dataset values as trainable values in the feature vector. This process is called **feature engineering**, and it is a vital part of machine learning. The most common feature engineering techniques are:

- **Normalisation**: Converting numerical values into a standard range.
- **Binning** (also referred to as **bucketing**): Converting numerical values into buckets of ranges.

Normalisation

After examining your data through statistical and visualization techniques, you should transform your data in ways that will help your model train more effectively. The goal of **normalization** is to transform features to be on a similar scale. Normalization might manipulate  and  so that they span a similar range, perhaps 0 to 1.

Binning

Binning (also called **bucketing**) is a **feature engineering** technique that groups different numerical subranges into *bins* or ***buckets***. In many cases, binning turns numerical data into categorical data. For example, consider a **feature** named **x** whose lowest value is 15 and highest value is 425. Using binning, you could represent **x** with the following five bins:

- Bin 1: 15 to 34
- Bin 2: 35 to 117
- Bin 3: 118 to 279
- Bin 4: 280 to 392
- Bin 5: 393 to 425

Scrubbing

Many examples in datasets are unreliable due to one or more of the following problems:

Problem category	Example
Omitted values	A census taker fails to record a resident's age.
Duplicate examples	A server uploads the same logs twice.
Out-of-range feature values.	A human accidentally types an extra digit.
Bad labels	A human evaluator mislabels a picture of an oak tree as a maple.

Time must be spent cleaning the data as it may throw off model predictions

Qualities of Good Numerical Features

- Clearly Named
- Checked or tested before training
- Sensible

Completion

You earned the **Machine Learning Crash Course: Numerical data** badge!

The badge has been added to your profile.

