Introduction

Whether you consider yourself a data scientist, an analyst, or a statistical hobbyist, you already now that the best part of any project is the modeling. No matter what part of the modeling world you find yourself living, models give us the ability to answer questions, make recommendations, and understand the world around us a little bit better. While we might be understanding the world a little better, do we always understand what our models are doing at a fundamental level? Let's look at the following equation for Pearson's correlation coefficient:

$$r = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\Sigma(x_i - \bar{x})^2 \Sigma(y_i - \bar{y})^2}}$$

You don't need to memorize this formula to interpret the correlation results, but knowing how those individual pieces work together only helps to deepen your understanding of the model. The goal of this book isn't to bore you with formulas, but to show you how those formulas work. Not only do we want to show you how those formulas work, but to show you how they work in a data-based context – we will eschew random numbers and use real data.

Code

You've probably noticed a trend in books, blogs, and courses of choosing R or Python. While practioners, authors, scholars, and professors alike often choose one over the other, we take a different approach – both languages are great and it is in your best interest to flip back and forth between the two. Throughout this book, we will be presenting ideas in both R and Python; you can use both or take your pick, but we want to give you both. Our goal isn't to obscure the ideas behind packages and specialty functions, but to show you the most basic functions behind big model ideas.

When presenting code, we will be specific and tell you which language we are using before we get to the coding. For example, we will use R to load the `ggplot2` package and produce a visualization:

```r
library(ggplot2)

ggplot(data = diamonds,
       mapping = aes(x = carat, y = price)) +
  geom_point(alpha = .5) +
  theme_minimal()
```
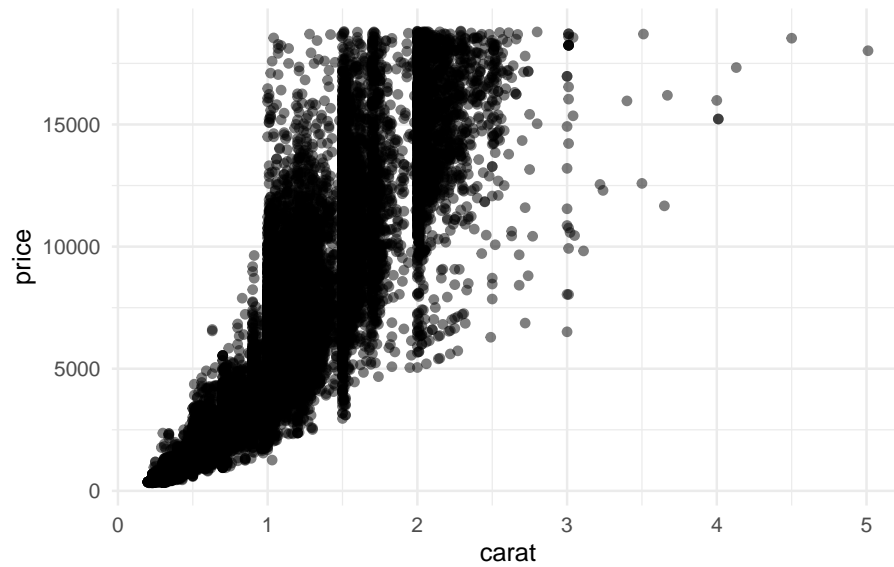
Now, we will use Python to import **seaborn** and create a visualization:

```python
import seaborn as sns
import matplotlib.pyplot as plt
sns.set_theme(style="whitegrid")

diamonds = sns.load_dataset("diamonds")

sns.scatterplot(data=diamonds,
                x="carat", y="price",
                alpha = .5)

plt.show()
```