Codecademy on Data Science

David Mukajanga

10/9/2021

# Foreword

## Welcome to The Data Scientist Careet Path

The Data Scientist Career Path is designed for you to gain the technical skills, knowledge, resources, and confidence you need to get your start as a data scientist.

After this Path, you will be able to:

* Create programs using Python 3
* Move off of the Codecademy platform with the Command Line, Jupyter Notebook, Git, and GitHub Desktop
* Acquire and query data with SQL and BeautifulSoup
* Manipulate data using NumPy and pandas
* Wrangle and tidy data with pandas
* Summarize and analyze data with scikit-learn
* Implement many different hypothesis tests using Python
* Visualize data using Matplotlib and seaborn
* Communicate data science findings effectively
* Work with text data using natural language processing
* Build, train, and test a variety of supervised and unsupervised machine learning models
* Understand the fundamentals of deep learning
* Work with aggregate functions and joins in SQL  
  You will also complete several Challenge Projects along the way that will give you some guided practice of the material you’ve covered. Moreover, you will demonstrate your knowledge in several Portfolio Projects that exist throughout the Path. You can complete the Portfolio Projects either in parallel with or after taking the prerequisite content—it’s up to you!

Here’s a preview of the Portfolio Projects featured in this Path:

U.S. Medical Insurance Costs  
\* Use Python to analyze a dataset of U.S. medical insurance costs.

Life Expectancy and GDP  
\* Analyze, prepare, and plot data about GDP and life expectancy from the World Health Organization.

Biodiversity in National Parks  
\* Interpret data about endangered species from different parks and prepare your findings for presentation.

OKCupid Date-A-Scientist  
\* Use machine learning to build, train, and test predictions about data from the dating app OKCupid.

Data Scientist Final Portfolio Project \* Wow prospective employers with a portfolio project focused on a topic of your own choosing.

## Helpful Resources

Congratulations on enrolling in the Data Scientist Career Path. We’re excited that you’re here. To help you during your learning journey, we’re sharing with you a set of resources. These resources include popular data science resources on the web, as well as recommended, optional books that can be used to supplement our Codecademy content.

### Popular Data Science Resources

As you go through the path, we’ll suggest certain resources for you to use, like articles, videos, tutorials, and documentation. Here are some additional resources that are considered groundbreaking, significant, or classics in the industry and will help you throughout your Path:

* Documentation: [The Python Standard Library](https://docs.python.org/3/library/index.html)
* Documentation: [Jupyter Notebook](https://jupyter-notebook.readthedocs.io/en/stable/)
* Resource: [SQL Cheatsheet](https://www.codecademy.com/learn/learn-sql/modules/learn-sql-manipulation/cheatsheet)
* Resource: [Regex Cheatsheet](https://www.codecademy.com/learn/practical-data-cleaning/modules/data-cleaning-with-pandas/cheatsheet)
* Documentation: [pandas documentation](https://pandas.pydata.org/docs/index.html)
* Documentation: [NumPy Manual](https://numpy.org/doc/stable/)
* Resource: [Statistics for Data Analysis Cheatsheet](https://www.codecademy.com/learn/ida-6-statistics-for-data-analysis/modules/ida-6-1-variance-and-standard-deviation/cheatsheet)
* Resource: [Hypothesis Testing Cheatsheet](https://www.codecademy.com/learn/stats-hypothesis-testing/modules/hypothesis-testing-testing-a-sample-statistic/cheatsheet)  
  Documentation: [Matplotlib: Python plotting](https://matplotlib.org/)
* Documentation: [API reference — seaborn](https://seaborn.pydata.org/api.html)
* Resource: [The Data Visualization Catalogue](https://datavizcatalogue.com/)  
  Go ahead and bookmark these resources—you won’t need them now, but they’ll be useful to reference and use as you go through the Path.

### Recommended Books

Codecademy prides itself on its interactive lessons, but sometimes it’s useful to have a couple of books on hand. The books featured in this Path are considered classics in the industry and many Codecademy learners (and employees!) have found them useful.

* Book: A Whirlwind Tour of Python, Jake VanderPlas
* Book: Python for Data Analysis: Data Wrangling with Pandas, Numpy, and Ipython, Wes McKinney
* Book: Python Data Science Handbook, Jake VanderPlas, [also available for free here](https://jakevdp.github.io/PythonDataScienceHandbook/)
* Book: Algorithms of Oppression: How Search Engines Reinforce Racism, Safiya Umoja Noble
* Book: Introduction to Machine Learning with Python: A Guide for Data Scientists, Andreas C. Müller, Sarah Guido
* Book: Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy, Cathy O’Neil
* Book: Python Feature Engineering Cookbook, Soledad Galli
* Book: Deep Learning with Python, François Chollet

While the books listed are recommended and will help you on your coding journey, purchasing and reading them is optional. When possible, we’ve suggested a free alternative. The content covers the same topics that you’ll find in our own lessons, but may be useful for those who want another explanation, perspective, or other opportunities for practice.

For those who choose to use them, relevant chapters and sections are woven throughout the syllabus. For those who are not using them, when you encounter a content item that is a reading, click the ‘Next’ button to move forward to the next content item. It will still count towards your overall progress, meaning that you do not need to purchase and read the books to complete these paths and receive a certificate.

Best of luck!

### Community Resources

Learning a programming language can be like learning a spoken language. It’s an incredible feeling when someone speaks to you and you understand what they’re saying! But it’s much more powerful when you are able to speak in response, using the new language with others. You can start practicing what you’ve learned by visiting the Codecademy Community forums or the [Codecademy Discord Server](https://discord.com/invite/QBn5MxG).

Codecademy Chapters are a great way to team up to reach your goals. Chapters meet virtually at a set time, and are peer-led by a Codecademy learner, just like you. Each Chapter will have a focus, ranging from web development projects, to mobile development, to technical interview preparedness. [Find a Chapter near you or start your own](https://community.codecademy.com/).

# INTRODUCTION TO DATA SCIENCE

## Introduction

Our world contains massive amounts of data. In fact, more data exists in the world from the last two years than in the last two hundred. But what does that mean and why should we care?

Data gives us information about the way the world works. And information can carry meaning - from a click telling us what someone likes, to toxins in the water signaling a health concern. But data is meaningless unless we do something with it. That’s where data science comes in.

Data science enables us to take data and transform it into meaningful information that can help us make decisions. Data science is interdisciplinary and combines other well-known fields such as probability, statistics, analytics, and computer science. The work that you might do could range from writing up reports to building machine learning models. No matter what your interests are, data science is applicable - because these days, we have data on everything!

In this lesson, we’ll introduce you to building blocks of data science so you can get a sense of the different skills you will need to pursue a career in the field.

## Statistics

A fundamental part of data science is statistics.

Some people say that data science is statistics dressed up for the 21st century - and there’s some truth in that. Statistics has been practiced for centuries but with the advent of computing prowess in the middle of the 20th century, statistics has taken on a new form.

As a refresher, statistics is the practice of applying mathematical calculations to sets of data to derive meaning. Statistics can give us a quick summary of a dataset, such as the average amount or how consistent a dataset is.

There are two types of statistics: descriptive statistics and inferential statistics. Descriptive statistics describe a dataset using mathematically calculated values, such as the mean and standard deviation. For instance, the graph below from FiveThirtyEight charts the wage gap between American men and women in 2014. An example of a descriptive statistic would be that at the 90th income percentile, women make 80.6% of what men make on average.

These values are useful when summarizing data collections.

On the other hand, inferential statistics are statistical calculations that enable us to draw conclusions about the larger population. For instance, from looking at the graph we can infer that at the 99th income percentile, women make less than 78% of what men make on average. We can also infer that the reason why the wage gap is smallest at the 10th income percentile is because the minimum wage for men and women is the same.

## Probability

Another field that informs the field of data science is probability.

Probability is the mathematical study of what could potentially happen. Fun fact: the study of probability came about as a method to decode secret messages.

In data science, probability calculations are used to build models. Models are able to help us understand data that has yet to exist - either data we hadn’t previously collected or data that has yet to be created. Data scientists create models to help calculate the probability of a certain action and then they use that probability to make informed decisions.

For instance, the social networking company Facebook collects data on the likes and dislikes of its users. Data scientists use that data to create models to calculate the probability of a user liking a certain advertisement. So, if you like several facebook posts about football, then a model may calculate a high probability of you positively responding to an advertisement selling football tickets at a stadium near you.

Calculating the probability of an event is sometimes dependent on other factors. For instance, in the birthday problem. “What is the probability that two people in a room have the same birthday?” the probability is dependent on the number of people in the room.

Other times, the probability of something is constant. For instance, the probability of flipping a coin and it landing heads will always be 50%.

## Programming

Given that we’re a “learn to code” website, this one may seem fairly obvious. But in all seriousness, programming is an essential part of data science. It’s also what sets data science apart from similar fields, like data analytics.

Programming is the practice of writing commands for a computer to execute. Computer Science is the discipline of interacting with computation systems.

A computer program is a series of instructions that tells the computer to perform a certain task. This could range from simply asking a computer to print, “Kirby has the best superpower!” to asking a computer to create a model that recommends movies based on your previous interests.

In data science, programming allows us to hand the processing power over to the computers. Given the right commands, computers can process millions of data points in a matter of seconds. In further Codecademy content, you will learn to write code that organizes and analyzes data. Furthermore, within data science, programs will allow you to reproduce experiments by simply running the program again.

You will also learn how to program models that can make predictions based on data points. These models are the basis of machine learning - a field of computer science that allows computers to make predictions based on data.

## Domain Expertise

One of the reasons that data science is one of the most interesting fields to go into is that it doesn’t limit you to a specific industry. Interested in airplanes? Work for a company on ticket pricing! Like fashion? Create a model that predicts the latest trends.

Domain Expertise refers to the particular set of knowledge that someone cultivates in order to understand their data. You may be able to crunch the numbers, but in order to understand their meaning, you’re going to need a lot of context. Sometimes this context comes from your research, your teammates, or the knowledge that you gain after working in a particular industry.

So take the time to think about where your passions lie (outside of data science, of course). Maybe you’re an avid gardener. How can data science help you learn more about your passion for orchids? Or perhaps you’re an all-star baseball player. Could you improve your stats if you crunch the numbers (remember that movie, Moneyball?).

When you are passionate about a particular field, your data science skills will make you extremely valuable in that field.

## Review

Review! You now know the basic parts that make up the field of data science.

* Data Science—the field of taking data and transforming it into meaningful information that can help us make decisions
* Descriptive Statistics—statistics that describe the data in objective terms
* Inferential Statistics—inferences for the overall population based on data
* Probability—the likelihood that an event will happen
* Programming—the act of giving the computer instructions to perform a task
* Domain Expertise—the particular set of knowledge that someone cultivates and brings with them in order to understand their data
* Data science is in every field. Take a look at these cool data science projects that range from politics to space travel.

In our next lesson, we will go over how the different aspects of data science make up the data science process. These building blocks will give you an overview of how exactly data science practitioners collect and use data.

### Instructions

What about data science excites you? Could you see career possibilities in data science?

It’s ok if you don’t know the answer yet! In our next lesson, we will delve a little further into data science.

# Introduction To Functions

## Defining a Function

A function consists of many parts, so let’s first get familiar with its core - a function definition.

Here’s an example of a function definition:

def function\_name():  
 # function's tasks go here

There are some key components we want to note here:

* The def keyword indicates the beginning of a function (also known as a function header). The function header is followed by a name in snake\_case format that describes the task the function performs. It’s best practice to give your functions a descriptive yet concise name.
* Following the function name is a pair of parenthesis ( ) that can hold input values known as parameters (more on parameters later in the lesson!). In this example function, we have no parameters.
* A colon : to mark the end of the function header.
* Lastly, we have one or more valid python statements that make up the function body (where we have our python comment).

Notice we’ve indented our # function tasks go here comment. Like loops and conditionals, code inside a function must be indented to show that they are part of the function.

Here is an example of a function that greets a user for our trip planning application:

def trip\_welcome():  
 print("Welcome to Tripcademy")  
 print("Let's get you to your destination)

## Calling a Function

Now that we’ve practiced defining a function, let’s learn about calling a function to execute the code within its body.

The process of executing the code inside the body of a function is known as calling it (This is also known as “executing a function”). To call a function in Python, type out its name followed by parentheses ( ).

Let’s revisit our directions\_to\_timesSq() function :

def directions\_to\_timesSq():  
 print("Walk 4 mins to 34th St Herald Square train station.")  
 print("Take the Northbound N, Q, R, or W train 1 stop.")  
 print("Get off the Times Square 42nd Street stop.")

To call our function, we must type out the function’s name followed by a pair of parentheses and no indentation:

directions\_to\_timesSq()

Calling the function will execute the print statements within the body (from the top statement to the bottom statement) and result in the following output:

Walk 4 mins to 34th St Herald Square train station.  
Take the Northbound N, Q, R, or W train 1 stop.  
Get off the Times Square 42nd Street stop.

Note that you can only call a function after it has been defined in your code.

Now it’s your turn to call a function!

### Instructions

#### 1.

Call the directions\_to\_timesSq() function.

Click Run to see it execute and print out.

#### 2.

Add an additional print statement to our directions\_to\_timesSq() function.

Have the statement print "Take lots of pictures!"

Run your code again and see how your output changes.

### The Code

def directions\_to\_timesSq():  
 print("Walk 4 mins to 34th St Herald Square train station.")  
 print("Take the Northbound N, Q, R, or W train 1 stop.")  
 print("Get off the Times Square 42nd Street stop.")  
 print("Take lots of pictures!")  
  
# Call your function here:  
  
directions\_to\_timesSq()