# What is Plotnine?

The plotnine python library brings the power of R’s ggplot2 to Python. Gain access to functions like:

ggplot() – Make the plot canvas (layout).

aes() – Map pandas DataFrame columns to the plot aesthetics (x, y, color, fill, etc).

**Geometries** – Add geometry layers including geom\_point(), geom\_smooth().

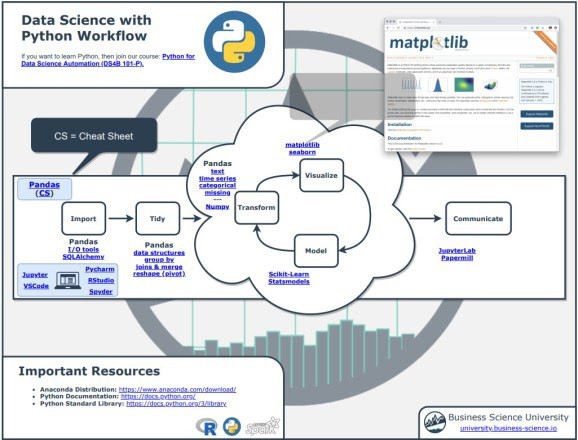
And more!

# Before we get started, get the Python Cheat Sheet

Plotnine is great for data visualization in Python if you are coming from an R background. But, you might want to explore documentation for the entire Python Ecosystem (pandas, plotnine, plotly, and many more libraries).

### Ultimate Python Cheat Sheet:

This gives you access to the entire Python Ecosystem at your fingertips via hyperlinked documenation and cheat sheets.

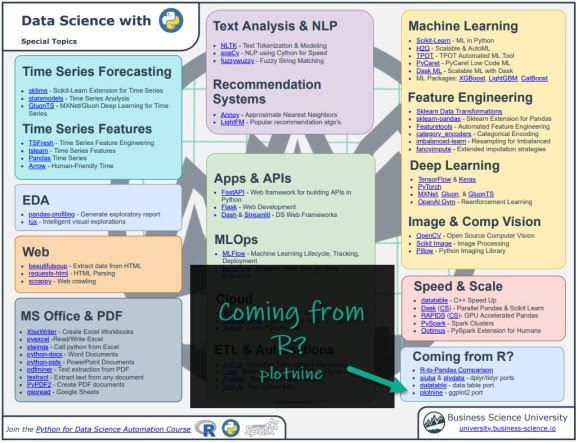


(Click image to download)

### If you’re coming from R, navigate to “Coming From R?” Section

Next, go to the section, “Coming from R?”. You can quickly get to the Plotnine

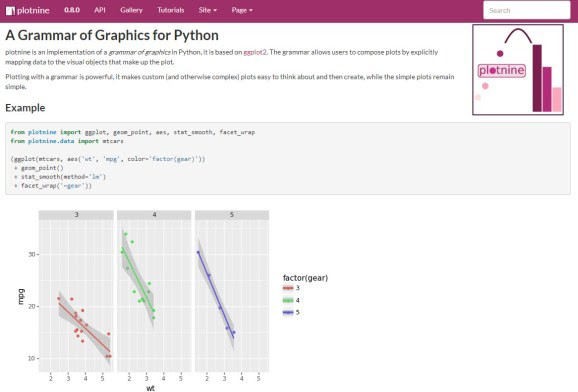
Documentation.



(Click image to download)

### Explore Plotnine

You have access to the Plotnine Documentation at your fingertips.



Onto the tutorial.

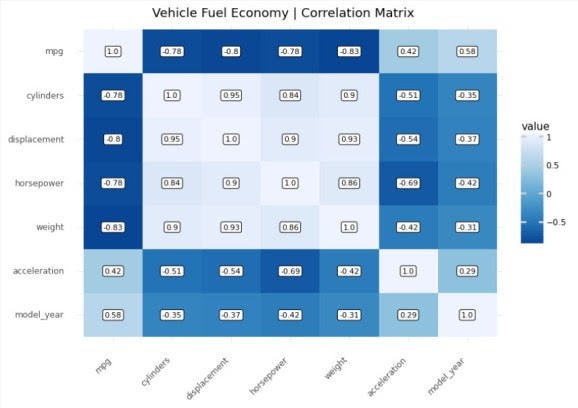
# How Plotnine Works

From the *Plotnine Documentation*, you can see that the grammar of graphics from ggplot is used to add layers that control geometries, facets, themes, and more.



# Making a Correlation Matrix Plot

Let’s check out how to make a professional **correlation matrix plot** with plotnine.



Get the code.

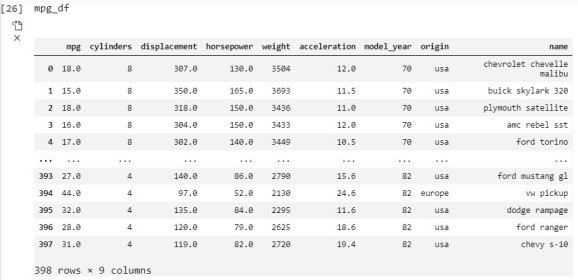
## Step 1: Load Libraries and Data

First, let’s load the libraries and data. From the libraries, we’ll import numpy and pandas to start out. We’ll also load the mpg dataset.



Get the code.

We’ll also load the mpg\_df data set.



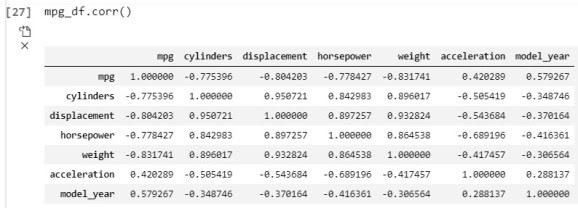
## Step 2: Expose Relationships with Correlation

Goal: Understand Relationships to Fuel Economy (mpg) versus vehicle attributes like weight, cylinders, and model year.

The correlation matrix is a square (n-by-n) matrix that shows the relationships between each feature. The correlation values range from -1 to +1 indicating both the strength (magnitude) and direction (positive/negative) of the relationship.

### Code

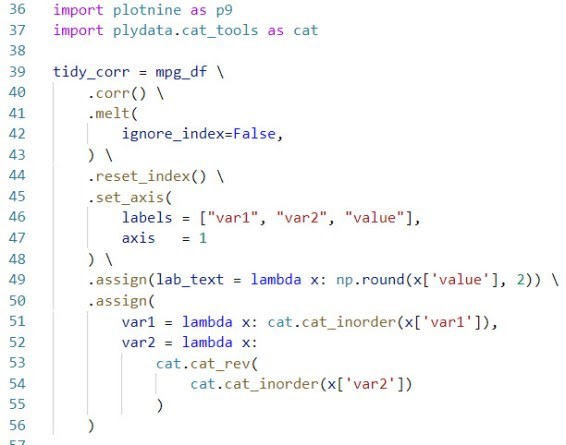
We’ll use the corr() method from Pandas to make a **correlation matrix** as a Pandas DataFrame.



## Step 3: Wrangle the Data into Tidy Format

Goal: Prepare the data for visualization with plotnine by formatting in “long” (“tidy”) format

The plotnine data visualization API requires data to be in the “tidy” or long format where each row is an observation. In this case, we need each row to contain the first variable, the second variable, and the value of the correlation. We can do this with pandas. Pandas can be a challenge for beginners.



Get the code.

The trick here is to use:

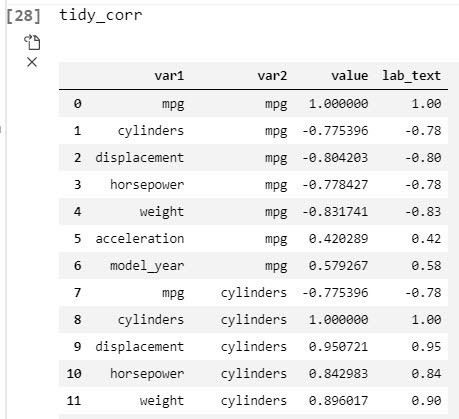
Import plotnine and plydate.cat\_tools to use ggplot functionality next and to more easily work with categorical data

melt() to pivot the data longer

assign() to add label text columns for the heatmap labels

assign() and cat\_inorder() to organize the categorical columns as categories in the correct order.

This outputs the data in Tidy format.

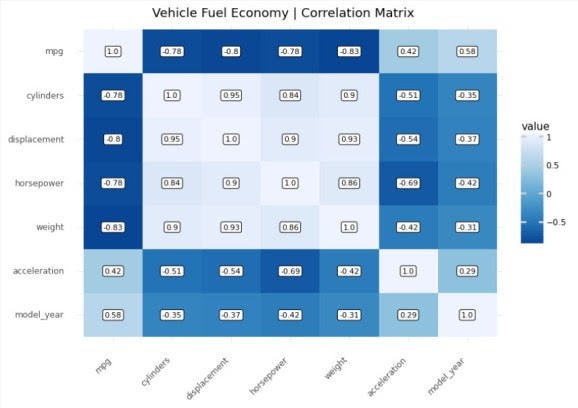


## Step 4: Make the correlation visualization with plotnine

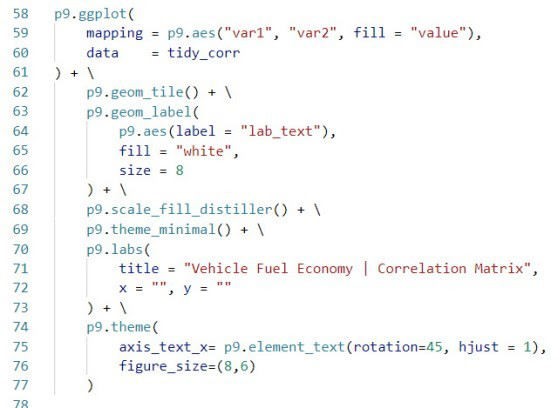
Goal: Make a professional-looking correlation plot that could be used in a business report to highlight key relationships to management.

Correlation visualizations are very powerful for business reporting as they can highlight key relationships for management. The problem is that many data scientists don’t know how to make them look professional, which can detract from your message to business stakeholders. Thankfully, plotnine solves this challenge.

First, here’s the correlation matrix heatmap visualization. We can clearly see that as cylinders increase (bigger engine) and weight increases (larger vehicles), fuel economy (mpg) tends to decrease. Conversely, as acceleration increases (possibly due to lower weight) and model year increases (newer vehicles), fuel economy tends to increase.



Next, here’s the code used to generate the visual.



The trick here is to use:

geom\_tile() to make the heat map.

geom\_label() to add label text for the correlation values.

scale\_fill\_distiller() to add a nice fill to the tile to give a professional

appearance.

# Summary

This was a short introduction to plotnine, which brings ggplot2 to python. If you’re coming from R, plotnine is a great package to make professional plots in Python.

With that said, you’re eventually going to want to learn pandas, the most widely used data wrangling tool in Python. Why?

Our data wrangling code was written in Pandas

Most data science teams use Pandas

Pandas plays nicely with Plotnine

So, it makes sense to eventually learn Pandas and Plotnine to help with communication and working on R/Python teams.

If you’d like to learn data science for business with Python, Pandas, and Plotnine from an R-programmers guidance, then read on.…