

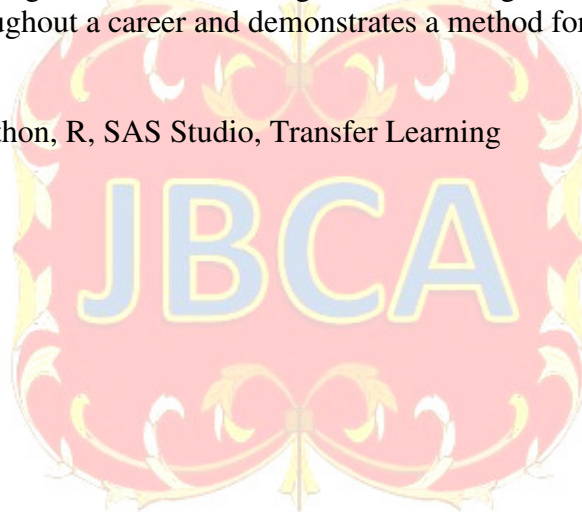
Analytics: Transitioning among software applications

Carolyn LaMacchia
Bloomsburg University of Pennsylvania

ABSTRACT

Data analysis techniques guide research, strategies, and decision making in a number of fields. Higher education curriculum responds to this need through course work which analyses real-world problems using analytic techniques and software applications including Python, R, and SAS Studio. This article presents an assignment which contributes to the preparation of students to actively participate in a variety of technology environments. The assignment includes the creation of a simple analytic process using a familiar software application then challenges students to recreate the equivalent analytic process with unfamiliar tools by drawing upon the power of the transfer learning technique. The learning process is accelerated as students recognize similarities among the tools. This assignment encourages the recognition of the need for lifelong learning throughout a career and demonstrates a method for supporting the learning process.

Keywords: Analytics, Python, R, SAS Studio, Transfer Learning



Copyright statement: Authors retain the copyright to the manuscripts published in AABRI journals. Please see the AABRI Copyright Policy at <http://www.aabri.com/copyright.html>

INTRODUCTION

Researchers, governments, nonprofit, and for-profit organizations depend upon data analysis techniques to guide strategies and decision making. Analytic skills are in high demand in research and a number of industries (Susnjak, Ramaswami, and Mathrani, 2022). There are multiple technology applications that provide a means to apply analytical techniques to the growing pools of data. As expected, higher education curriculum includes courses to prepare students to build data analysis and research skills (Susnjak, Ramaswami, and Mathrani, 2022). Concepts are applied through coursework including a number of possible technology applications. Python, R, and SAS Studio are analytic applications frequently mentioned in academic literature for their application in academic course work (Kanuru and P, 2020; Susnjak, Ramaswami, and Mathrani, 2022).

Higher education curriculums incorporate data science and analytics techniques through both content discussion and hands-on assignments requiring technology applications (Aliedeh, 2015). Research supports incorporating applied teaching techniques in delivering course content because students build an expertise through the examination of theory and the practical application of the theory to problems and situations (Riley, Eldridge, and Venkatesh, 2021). Python, R, and SAS Studio are applications widely used in hands-on assignments to demonstrate the application of the theory to real-world situations in a wide variety of disciplines (Lasser, Manik, Silbersdorff, Säfken, and Kneib, 2021).

There is extensive research that supports the viability of the transfer learning technique (Mayes, 2019). Transfer learning is used to apply knowledge and skills from one context to a different context. The application of transfer learning can be subconsciously or explicitly planned with both applications supporting a deeper understanding of the familiar and new situations (Raza and Williams, 2020). Transfer learning is a successful educational technique in multiple disciplines including information technology. As an example, it is considered quicker and easier to learn a new programming language after learning at least one other language (Mayes, 2019). This is explained through a number of reasons including students' skills and confidence after learning one programming language can accelerate the process for learning a second programming language because students can compare procedures and syntax (Mayes, 2019).

Upon graduation, students not only apply the knowledge and skills from their educational program but also adapt to challenges in new environments (Levin, 2015). Employment opportunities, especially in the Information Technology (IT) field, often specify a requirement for specific skills and even specific technology application(s) (Kanuru & P, 2020). Higher education programs need to prepare students to recognize similarities in situations and foster the ability to adapt to new environments (Levin, 2015). Post-graduation, students require the agility to respond to changing technology environments (Levin, 2015). The transfer learning technique can contribute to student agility in learning new skills (Mayes, 2019).

This article presents a hands-on assignment for applying three well-known analytic applications to an analytics problem. The software applications are Python, R, and SAS Studio. The design of the assignment is supported by the transfer learning technique. Students' familiarity with one of the three applications, from previous course work, supports the completion of the assignment using all three applications. This assignment contributes to the preparation of students to actively participate in a variety of technology environments. In

addition, it encourages lifelong learning throughout a career. The assignment challenges the student to replicate the same functions of this program using two additional analytics software applications.

Analytic Software Applications

Data repositories containing both structured and unstructured data are growing at an exponential rate. Organizations of all forms, in most industries, have implemented analytic software applications to support the decision-making process. Python, R, and SAS Studio are programming language platforms used to harness usefulness from data. All three applications have an excellent reputation and are widely used by organizations and in educational settings (Brittian, Cendon, Nizzi, and Pleis, 2018). With the explosion of data comes a growing job market with positions using all three tools (Brittian, et.al, 2018).

Universities incorporate applications including Python, R, and SAS Studio to demonstrate data analysis techniques in multiple disciplines (Python, 2022; R, 2022; SAS, 2022). All three applications are platform independent; available to Windows, macOS, and Linux operating systems. The applications are incorporated in both online and face-to-face course work at both the undergraduate and graduate level (Brittian, et.al, 2018). Python, R, and SAS Studio are widely used in multiple disciplines including business, humanities, natural and applied sciences, and social sciences (Brittian, et.al, 2018).

Research reports that it is difficult to compare and rank the analytic software applications (Brittian, et.al, 2018). In industry, Python and R tend to be used by small to medium-sized organization while SAS is used by larger organizations. All three software applications provide an extensive collection of data manipulation and analysis techniques. While there are differences in processing techniques and performance, all three have multiple similarities. Through incorporating the transfer learning technique, the assignment supports portability in the use of the three applications. Because of the many similarities, becoming familiar with one application can lead to an easier transition in the use of another application.

Python

Python was released about three decades ago as an open source, free for commercial use, general purpose application. Python is used for web, interface, and general programming. Like SAS Studio, it is also a development environment for preparing data and performing analytical analysis. Python may be installed on multiple platforms including Windows, macOS, and Linux.

The selection of Python has grown in popularity as the introductory language in the curriculum for technology majors (Nelli, 2018). The growth is due to its simple syntax, versatile application, and free availability. The Python User Groups (PUGs) are large developer and user communities that contribute to its growth and evolution. Because it is an open source language, there is no warranty or official technical support but there are many online forums that support troubleshooting efforts.

R

R, named for its creators, Ross Ihalka and Robert Gentleman, has been available as an open-source project for about three decades, like Python (Crawley, 2022). Similar to Python and

SAS Studio, R has a development environment, called RStudio, for preparing data and performing analytical analysis. And like Python, R may be used in command mode outside of the development environment. R and RStudio may be installed on multiple platforms including Windows, macOS, and Linux.

R is considered among the best applications for researchers and analysis to prepare, examine and present information (Crawley, 2022). This is due to its simple syntax, versatile application, and free availability. At present, the R Development Core Team maintains the central repository for R source code (Crawley, 2022). The R Foundation and R Contributors are large developer and user communities that contribute to R's growth and evolution. Like Python, because it is an open source language, there is no warranty or official technical support. There are online forums that support troubleshooting activities.

SAS Studio

SAS Institute is the world's largest independent software organization, specializing in a suite of analytics software applications. The SAS analytics suite includes multiple applications for data preparation and analysis. SAS is planning to reorganize as a public enterprise through an Initial Program Offering (IPO) scheduled for 2024. SAS software is used by most Fortune 500 Companies. According to Gartner Peer insights, SAS Studio has a 4.3 out of 5 satisfactory rating (Gartner, 2022). SAS is labelled a "visionary-leader" and provides for a complete and intuitive user interface (Gartner, 2022). As expected with a proprietary tool, there is a warranty and there is official technical support.

Although SAS is proprietary, it does provide for the freely available SAS OnDemand for Academics platform for educators and students. This is cloud-based application, available through a web browser, which is used for learning statistical analysis, data mining, and forecasting (Code, 2021). SAS OnDemand for Academics includes multiple applications including SAS Studio, a development environment for preparing data and performing analytical analysis. This assignment works with the SAS Studio environment.

THE ASSIGNMENT

Assignment Learning Objectives

This assignment is designed for students who are familiar with at least one analytics software application either Python, R, and/or SAS Studio and want to deepen their understanding of the application and learn at least one other analytics application. This assignment provides students with a hands-on learning experience where they pivot from a familiar software application to a different software application. The assignment uses the transfer learning technique to accelerate the learning process for the new application. The ability to quickly learn new software applications contributes to a professional's career flexibility.

This assignment supports the process of data analytic application adaptability so that students are prepared to actively participate in multiple environments. The assignment highlights the many similarities among the applications. The assignment has the following learning objectives: (1) Analyze the installation or accessing of open-source data analytic applications; (2) Demonstrate the application of multiple data analytic applications to a data analytics exercise; and, (3) Discover a process for pivoting between Data Analytics applications.

Audience

The assignment is suitable for an undergraduate or graduate course with a focus on data analytics and the application of a data analytics software application, specifically Python, R, or SAS Studio. It is recommended that the assignment be delivered after the students developed familiarity with one data analytics application. Exhibit 1 (Appendix) includes a list of recommended topics which should indicate that students are sufficiently familiar with at least one of the three data analytics applications (Python, 2022; R, 2022; SAS, 2022). It is not necessary to be familiar with all three data analytics applications before beginning this assignment; just one application. The assignment promotes the understanding of an alternative data analytics application by drawing from familiarity with a different data analytics application.

The assignment may be administered in a face-to-face class scheduled in a computer laboratory or in an online class. Since Python and R are open-source and SAS Studio is available through a web browser, face-to-face classes may complete the exercise in a computer laboratory or on their own computer devices. Online classes will complete the exercise on their own computer devices. Instructors should consider whether that the application installation experience provides a learning experience for the students in their particular course.

Assignment Directions

The purpose of this assignment is to complete a simple data analytics process using a familiar data analytics software application then replicating the data analytics process using a different data analytics software application. The analytics process will read an input file then calculate and graph the correlation of the variables within the file. To challenge the understanding of the data analytics software applications, the student will modify the data analytics process to include additional analytic functions. To continue developing skills, students are encouraged to experiment with the data analytics software applications after completing this assignment.

The assignment includes five sequential steps (1) create a data file; (2) execute analytical processing using a familiar application; (3) install, or gain access to, unfamiliar applications; (4) execute analytical processing using unfamiliar applications; (5) explore additional techniques in both familiar and unfamiliar software applications.

Step 1 – Create data file

The data analytics exercise accesses a file which includes multiple records with each record containing eight variables. All variables are numeric and are represented as decimal, integer and binary [0,1] data. The variables for each record are separated by a comma. The value of each variable is randomly generated. There are no missing variables within the records. There are 1,000+ records in the sample exercise file.

Create a data file, formatted as comma separated values (CSV), that will be accessed by the data analytics application. The file may be created using either Microsoft Office Excel® and saved as a CSV file or through a text editor with each record beginning on a new line and a comma separating each variable. The first record in the file is the header which identifies the

variable name which, in the sample file, are X and Y1 through Y7. Exhibit 2 (Appendix) includes the first few records in the sample data file.

Step 2 – Process Familiar Software Analytics Application

Execute the data analytics process using the file created in Step 1 with the familiar data analytics application. The process reads the input file and calculates and graphs the correlation, using Pearson correlation, for every possible pair of variables within the file. The process then calculates and graphs Pearson correlation for a set of variables and graphs the data for two variable in the dataset. In the sample file, the variables X and Y3 were selected.

Students import the file created in Step 1 into the data analytics software environment. This is the data analytics software application for which they are already familiar. Students then process the program code for the data analytics application. Exhibit 3 (Appendix) includes the suggested program code for Python, R, and SAS Studio.

Step 3 – Install or Gain Access to Unfamiliar Software Analytics Application

Install an open-source data analytics applications, Python or R, or obtain access to SAS Studio. Python and R may be installed on computers in a university computer laboratory or students' personal devices. Regardless of the choice, it is recommended that students follow the process to verify installation of Python and R and verify access to SAS Studio. Exhibit 4 (Appendix) includes installation/access notes for all three data analytics applications and instructions to verify installation/access.

Step 4 – Process Unfamiliar Software Analytics Application

Execute the data analytics process using the file created in Step 1 and each data analytics applications installed/accessed in Step 3. These are applications that are new to the students. This data analytics process is equivalent to the process completed in Step 2 with the familiar application. The process reads the input file then calculates and graphs correlation. The only significant difference among the three processes is the user environment and the coding syntax. Exhibit 5 (Appendix) includes a suggested solution for all three applications: Python, R, and SAS Studio.

Step 5 – Explore additional techniques

Explore additional data analytics processes for all three applications by comparing and contrasting several functions/processes using the file created in Step 1. This step of the assignment requires investigation using the applications' Help features. Students learn the purpose of the function/process, investigate the formatting of the code, and decipher the output. Students also practice the process for installing and accessing the functions in Python and R. SAS Studio procedures (functions) are already available in the SAS Studio environment without an additional installation process.

Research and install Python “summary” and “missingno” libraries then create Python code to process each using the file created in Step 1. Research and install R packages “summaryapplications” and “visdat” then create R code to process each function using the file

created in Step 1. Research SAS Studio processes “summary” and “freq” then create SAS Studio code to process each proc using the file created in Step 1. Interpret the output of each application. Compare and contrast the three applications. Students are encouraged to continue working with at least one of the new applications. This supports independent learning which will likely be encountered during a career.

Teaching Notes

The assignment includes installation directions and coding syntax which may be revised over time. Fortunately, Python, R, and SAS Studio include detailed Help features that support installation, access, and use of the applications. There is flexibility in the implementation of the assignment in three areas: (1) the assignment may be completed independently or together as a class; (2) the assignment may be shorted by exploring only one new software application; and, (3) the assignment may be simplified if technology support installs Python and R software applications in the computer laboratory. Each of these options is discussed below.

Students may complete the assignment off-site, independently using their own devices. Or, students may complete the assignment collectively in class either using their own devices or computer laboratory devices. This choice is available to both face-to-face courses and online synchronous courses. Regardless of the option selected, class discussion during and after the assignment is encouraged.

The application can be shorted by completing the assignment with the familiar analytics software application and one, instead of two, new software applications. This choice requires a modification to Steps 3, 4, and 5. Modify Step 3 and install or gain access to only one unfamiliar analytics software application. In addition, modify Step 4 and Step 5 by performing analytic processing using only one unfamiliar analytics software application.

The application can also be simplified for face-to-face classes, with a modification to Step 3. During Step 3, students follow directions to install Python and R on their own device. An alternative is to arrange for installation of Python and R in the classroom computer laboratory. In this case, students do not experience the installation process but can, at minimum, verify that the software applications have been installed. Regardless, individual student registration is required to access SAS Studio through a web browser.

LIMITATIONS AND CONCLUSION

The transfer learning technique is a recommended technique to support students in the learning of a software application or technology skill. The assignment launches the process for learning different, but related, software applications. The student is introduced to the software application and experiences success in completing an analytics exercise through a hands-on, guided approach and the transfer learning technique. Students are encouraged to continue working with one or more of the software applications to gain expertise. Completing this one assignment is not sufficient to master a software application. All three software applications have a vast and growing set of functions.

This assignment was developed and piloted in an undergraduate capstone course, Advanced Analytics, over several semesters. Students work extensively with SAS Studio during this course. The assignment was piloted using several methods. The first method was repeated for two semesters and was presented in online sections of Advanced Analytics. The second

method was repeated for three semesters in face-to-face sections of the Advanced Analytics course. In all cases, the assignment was well-received by the class. Students reported that they appreciated the guidance in learning to recognize similarities in software applications within a similar domain. Students were encouraged to continue working with new applications.

The intent of this assignment is to prepare students for life after graduation and the high probability that students may be learning new skills and using software applications different than those used in course work. Students are provided with the opportunity to practice self-development. Demonstrating the similarity among multiple software applications in the analytics domain may assist students in realizing the transfer learning technique supports and accelerates the learning process. This assignment may provide an additional example that reinforces the need for lifelong learning in multiple career avenues including the technology field.



REFERENCES

- Aliedeh, M. A. (2015). Call from the South for a Transparent Higher Education (THE) Part 3: Expanded TTA Conceptual Framework and Its Applications in Math, Science, and Engineering Education. *Journal of Higher Education Theory & Practice*, 15(7), 121–142.
- Brittain, J., Cendon, M., Nizzi, J., & Pleis, J. (2018). Data scientist's analysis toolbox: Comparison of Python, R, and SAS Performance. *SMU Data Science Review*, 1(2), 7.
- Code, R. (2021). *Getting Started with SAS Programming: Using SAS Studio in the Cloud*. Cary, NC, USA: SAS Institute Inc.
- Crawley, M. J. (2022). *The R Book*, 3rd edition. Hoboken, NJ, USA: John Wiley & Sons, Inc.,
- Gartner (2022). *Gartner Peer Insights*. Retrieved March 26, 2022, from <https://www.gartner.com/reviews/vendor/sas>
- Kanuru, S. L., & M, P. (2020). Lifelong Learning in higher education using Learning Analytics. *Procedia Computer Science*, 172, 848–852.
- Lasser, J., Manik, D., Silbersdorff, A., Säfken, B., & Kneib, T. (2021). Introductory data science across disciplines, using Python, case studies, and industry consulting projects. *Teaching Statistics*, 43, S190–S200.
- Levin, H. M. (2015). The importance of adaptability for the 21st century. *Society*, 52(2), 136–141.
- Mayes, T. (2019). *Learning theory and the new science of learning*. In *Rethinking Pedagogy for a Digital Age* (pp. 17-31). Routledge.
- Nelli, F. (2018). *Python Data Analytics: With Pandas, NumPy, and Matplotlib*, 2ed. NY, NY: Springer Science + Business Media.
- Python (2022). *Python: About*. Retrieved March 15, 2022, from <https://www.python.org/about/>
- R (2022). *The R Project for Statistical Computing*. Retrieved March 15, 2022, from <https://www.r-project.org/>
- Raza, S. A., & Williams, M.-A. (2020). Human Feedback as Action Assignment in Interactive Reinforcement Learning. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 14(4), 1–24.
- Riley, J., Eldridge, S., & Venkatesh, R. (2021). Engaging the Experts to Embrace the Future: An Interactive Professional Learning Experience. *Issues in Accounting Education*, 36(4), 41–58.
- SAS (2022). *SAS OnDemand for Academics*. Retrieved March 15, 2022, from <https://support.sas.com/en/software/ondemand-for-academics-support.html>
- Susnjak, T., Ramaswami, G. S., & Mathrani, A. (2022). Learning analytics dashboard: a tool for providing actionable insights to learners. *International Journal of Educational Technology in Higher Education*, 19(1), 1–23.

APPENDIX

Exhibit 1: Recommended topics that indicate familiarity with a data analytics application.

Python	R	SAS Studio
Installation	Installation	Registration
Launching Python	Launching R	Accessing SAS Studio
Coding environment	Coding environment	Coding interface
Package index	Package index	Function index
Package installation	Package installation	N/A
File access and handling	File access and handling	File access and handling
Variable Types	Variable Types	Variable Types
Coding Syntax	Coding Syntax	Coding Syntax
Report Creation	Report Creation	Report Creation
Graphics Creation	Graphics Creation	Graphics Creation
Accessing Help feature	Accessing Help feature	Accessing Help feature

Exhibit 2: Sample data file. Record one is the header which includes the names of the variables. In this sample, records two through seven are data records. The data file included about 1,300 records. Although the choice of variable types is not significant for this exercise, the variables are in the sample file are described as: X, Y2, and Y3 are real numbers with two decimal places; Y1 is an integer; Y4 and Y5 are category variables; and Y6 and Y7 are binary variables.

X,	Y1,	Y2,	Y3,	Y4,	Y5,	Y6,	Y7
21275.00,	18,	58.59,	4680.50,	3,	0,	0,	1
2174.20,	17,	70.92,	478.32,	2,	1,	1,	0
5606.32,	27,	69.30,	1240.79,	2,	3,	1,	0
27700.43,	32,	47.68,	6109.33,	1,	0,	1,	0
4872.24,	31,	60.65,	1074.57,	1,	0,	1,	0
4733.34,	30,	54.05,	1043.94,	2,	0,	0,	0

Exhibit 3: Recommended program code for Python, R, and SAS Studio.

Python Code

Prerequisites:

1. Python must be installed on the local environment. If necessary, refer to Exhibit 4 for installation notes.
2. To process data analytics code, install libraries matplotlib, numpy, pandas, and seaborn. This is required before the initial run of the program so that this library is part of the local Python environment. Internet access is required to install Python libraries. Libraries may be installed through the command prompt. Commands follow:

```
py -m pip install matplotlib
py -m pip install numpy
py -m pip install pandas
```

- py -m pip install seaborn
- Place input file, created in Step 1, in default “working directory” using command:
Import os then os.getcwd() indicates working directory.
 - Launch IDLE Shell to enter the following program syntax:

```
# Comment lines begin with a #hashtag
# Program Name: Data Analytics – Correlation – Python
# Program Author: Your Name
#
# Import functions with import command
import csv
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
#
# Initialize plot area
plt.show()
#
# Load test file into dataset “MyData”
MyData = pd.read_csv('TestData8x1300.csv')
#
# Display the first six rows of the MyData dataframe;
print(MyData.head())
#
# Create a correlation matrix
sns.pairplot(MyData)
plt.show()
#
# Calculate the correlation of X and Y3 and plot
np.corrcoef(MyData['X'],MyData['Y3'])
sns.scatterplot(data=MyData, x='X', y='Y3')
#
# show plots
plt.show()
#
# End
```

R Code

Prerequisites:

- R must be installed on the local environment. If necessary, refer to Exhibit 4 for installation notes.
- To process data analytics code, install library “corrplot”. This is required before the initial run of the program so that this library is part of the local R environment. Internet access is required to install R libraries. Libraries may be installed through the R menu

Packages-> install package(s) or from the console with the install command. As an example: install(corrplot)

3. Place input file, created in Step 1, in “working directory”. Command: getwd() indicates the working directory.
4. Launch R to enter the following program syntax in the R console:

```
# Comment lines begin with a #
# Program Name: Data Analytics – Correlation – R
# Program Author: Your Name
#
# Install library with library()
# corrplot – provides graphical display of correlation matrix
library(corrplot)
#
# Load test file into dataframe "MyData"
MyData <- read.csv(file="TestData8x1300.csv");
#
# Display the first six rows of the MyData dataframe;
head(MyData)
#
# Create a correlation matrix and store in MyCorrelationMatrix dataframe.
MyCorrelationMatrix = cor(MyData)
#
# Display the contents of MyCorrelationMatrix dataframe.
MyCorrelationMatrix
#
# Calculate the correlation of X and Y3 and place output in MyCorr dataframe.
mycorr <-cor(MyData$X, MyData$Y3)
#
# Establish text contents in the title dataframe.
# Contents are “X by Y3” and a data field calculated during the cor function
title <-paste("X by Y3: Correlation = ", sprintf("%s",mycorr))
#
# Graph the data points from the input data imported into MyData dataframe
plot(MyData$X, MyData$Y3,main=title,xlab="X",ylab="Y3")
#
# End
```

SAS Studio Code

Prerequisites:

1. SAS Studio access must be established. If necessary, refer to Exhibit 4 for installation note.
2. Access SAS Studio through a web browser. Internet access is required to use SAS Studio.
3. Upload input file, created in Step 1, in “working directory” in SAS Studio.
4. Access SAS Studio to enter the following program syntax in the user interface:

```

* Comment lines begins with * and end with ;
* Program Name: Data Analytics- Correlation - SAS;
* Program Author: Your Name;
*;
* Load test files into dataset "MyData";
data MyData;
    infile "/home/clamacch0/TestData8x1300.csv" DELIMITER=',' DSD;
    input X y1 Y2 Y3 Y4 Y5 Y6 Y7;
run;
*;
* Display the first six rows of the data file MyData;
* Create a report with a title";
title "List first six rows of data from the cvs file";
proc print data=MyData (obs=6) Label;
run;
*;
* Create a correlation matrix and print;
Title 'Exploring Correlations';
proc corr data=MyData plots(MAXPOINTS=NONE)=matrix(histogram);
run;
*;
* Calculate the correlation of X and Y;
*;
Title 'Exploring Correlations - X and Y3';
proc corr data=MyData plots(MAXPOINTS=NONE)=matrix(histogram);
    VAR X;
    WITH Y3;
run;
*;
* End;

```

Exhibit 4: Installation notes.

Python Installation Notes

Python is open source and is freely available through its website. Download a recent version. The Python version used to create code in this article is 3.10.5. Installation should include Pip, Python's package management system. There is an option provided during the installation process for selecting PiP and IDLE. Pip is necessary so that users have access to the many software packages developed for Python. These packages provide for simplistic access to powerful calculations. Installation should also include IDLE, Python's coding interface.

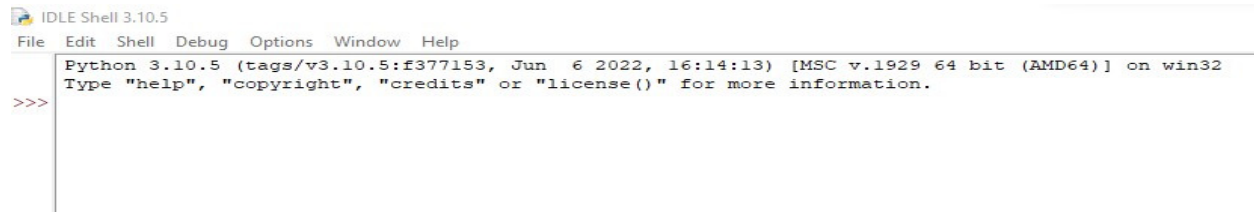
Verify the installation of Python in the computer laboratory or on student's personal device through the following steps:

1. Access command or terminal screen: for Linux use Ctrl+Alt+T; for Windows use Command Prompt; and, for MacOS use Finder>Applications>Utilities>Terminal.

2. Enter command `python --version` for Linux and Windows; `python -version` on MacOS.
3. Enter command `pip --version` for Linux and Windows; `pip -version` on MacOS.

The computer device must have Internet access to install packages in Python environment. Packages are required by the code used in this assignment. After installing packages, Python may be used without being connected to the Internet.

1. Access command of terminal screen (refer to installation, number 1).
2. Enter command: `pip install [package name]`. For example: `pip install numpy`. Repeat Install for packages: `numpy`, `matplotlib`, `pandas`, and `seaborn`.
3. Launch IDLE Python's coding environment, to confirm installation. IDLE coding environment:

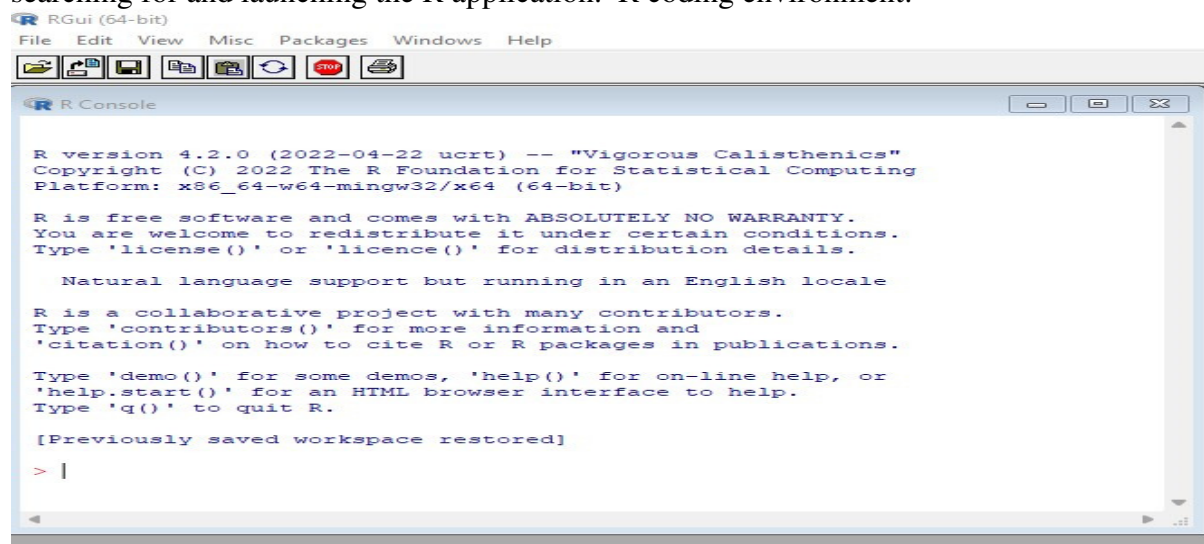


```
Python 3.10.5 (tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

R Installation Notes

R is open source and is freely available through its website. Download a recent version. The R version used to create code in this article is 4.2.0. Installation requires the selection of a CRAN (network file transfer process) and operating system. The computer device must have Internet access to install packages in R environment. One package is required by the code used in this assignment. There are two methods to install packages; through a command entered while on the R console and through the R console menu Packages. Command is `install.packages("corrplot")`, or follow prompts through `R>Packages>Install Package(s)`. After installing packages, R may be used without being connected to the Internet.

Verify the installation of R in the computer laboratory or on student's personal device by searching for and launching the R application. R coding environment:



```
R version 4.2.0 (2022-04-22 ucrt) -- "Vigorous Calisthenics"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> |
```

SAS Access Notes

SAS is a proprietary, commercially available application that provides for free academic use through a web browser. The faculty member creates their account and course through the SAS OnDemand for Academics. SAS assigns a course code to the faculty’s course. Once complete, each student creates an account. Email verification is required for both the faculty and student accounts. After students’ account is established, students enroll in the faculty’s course, using the course code, through the Control Center. Students are ready to access SAS Studio. Internet access is required. SAS Studio Coding Environment:

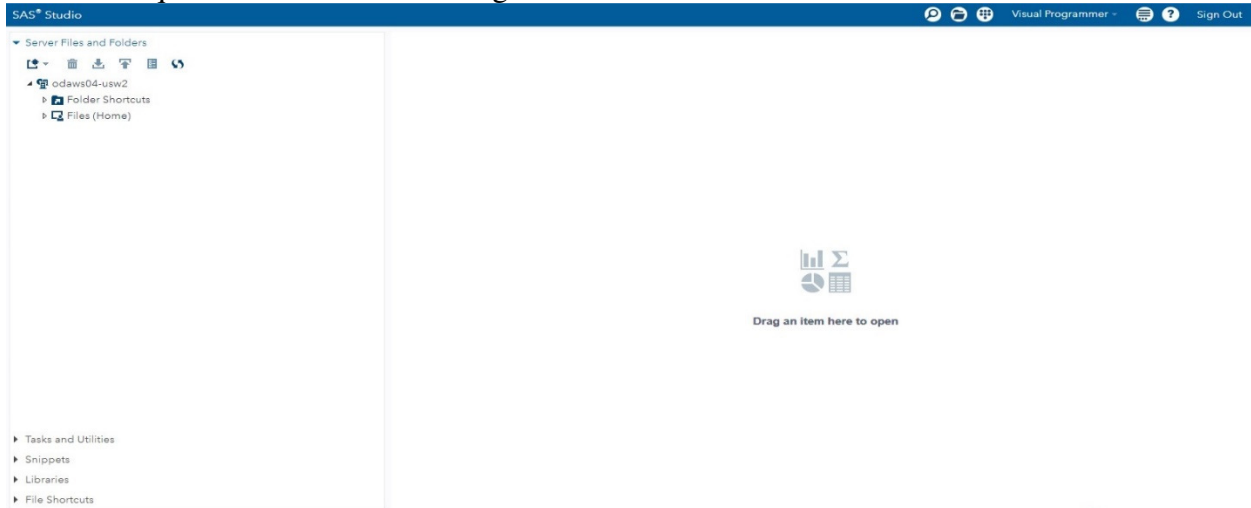
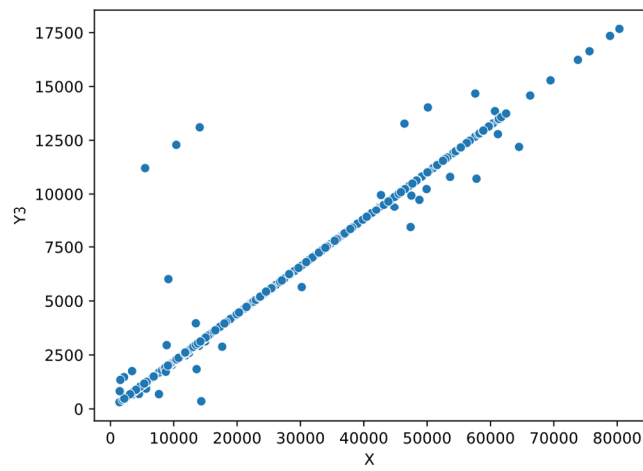
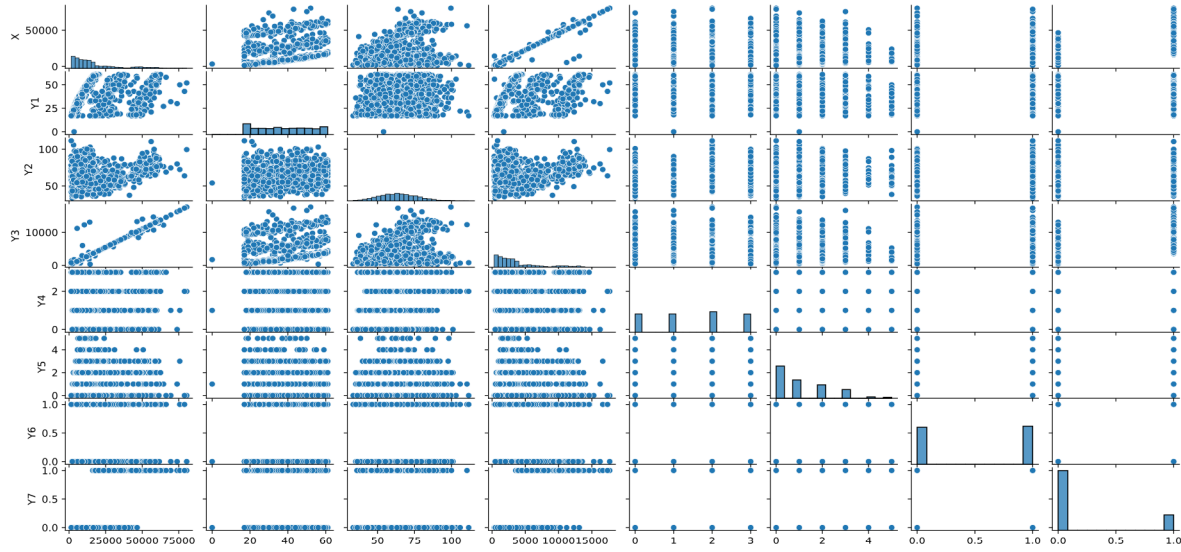


Exhibit 5: Suggested Solutions for all three applications: Python, R, and SAS Studio.

Python Suggested Solution:

X	Y1	Y2	Y3	Y4	Y5	Y6	Y7	
0	21275.00	18	58.59	4680.50	3	0	0	1
1	2174.20	17	70.92	478.32	2	1	1	0
2	5606.32	27	69.30	1240.79	2	3	1	0
3	27700.43	32	47.68	6109.33	1	0	1	0
4	4872.24	31	60.65	1074.57	1	0	1	0

(Continued)



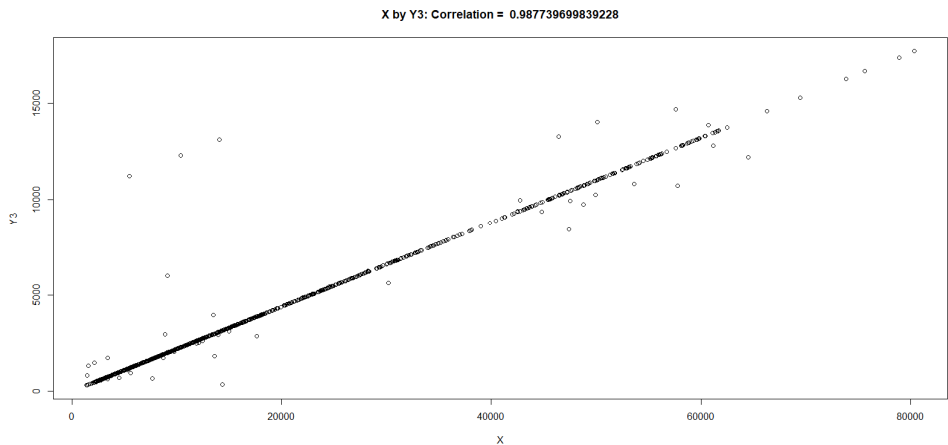
R Suggested Solution:

	X	Y1	Y2	Y3	Y4	Y5	Y6	Y7
1	21275.00	18	58.59	4680.50	3	0	0	1
2	2174.20	17	70.92	478.32	2	1	1	0
3	5606.32	27	69.30	1240.79	2	3	1	0
4	27700.43	32	47.68	6109.33	1	0	1	0
5	4872.24	31	60.65	1074.57	1	0	1	0
6	4733.34	30	54.05	1043.94	2	0	0	0

	X	Y1	Y2	Y3	Y4
X	1.000000000	0.297310612	0.20053617	0.98773970	-0.0049155937
Y1	0.297310612	1.000000000	0.11060207	0.29372202	0.0024207481
Y2	0.200536170	0.110602074	1.000000000	0.19433162	0.1554897640
Y3	0.987739700	0.293722025	0.19433162	1.000000000	-0.0089155301
Y4	-0.004915594	0.002420748	0.15548976	-0.00891553	1.0000000000


```

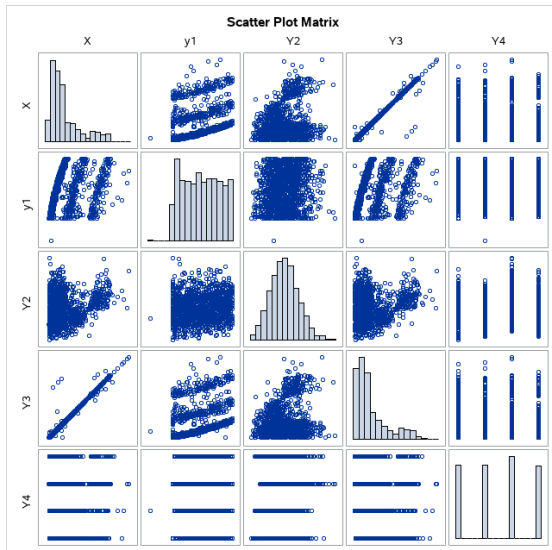
Y5 0.067827510 0.036678502 0.01018634 0.07338979 0.0155235960
Y6 0.056477149 -0.022474181 0.04730333 0.05028700 0.0063142528
Y7 0.787001456 -0.027764366 0.00559273 0.77685570 0.0001865855
  Y5      Y6      Y7
X 0.067827510 0.056477149 0.7870014560
Y1 0.036678502 -0.022474181 -0.0277643659
Y2 0.010186341 0.047303329 0.0055927295
Y3 0.073389794 0.050287003 0.7768557036
Y4 0.015523596 0.006314253 0.0001865855
Y5 1.000000000 0.013869115 0.0094592163
Y6 0.013869115 1.000000000 0.0765494221
Y7 0.009459216 0.076549422 1.0000000000
    
```



SAS Suggested Solution:

Obs	X	y1	Y2	Y3	Y4	Y5	Y6	Y7
1	21275.00	18	58.59	4680.50	3	0	0	1
2	2174.20	17	70.92	478.32	2	1	1	0
3	5606.32	27	69.30	1240.79	2	3	1	0
4	27700.43	32	47.68	6109.33	1	0	1	0
5	4872.24	31	60.65	1074.57	1	0	1	0
6	4733.34	30	54.05	1043.94	2	0	0	0

(Continued)



Pearson Correlation Coefficients X, Y3: .098774

