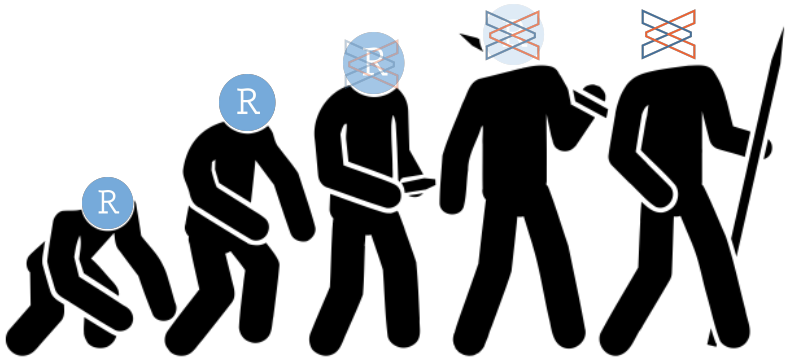
RStudio® / posit™

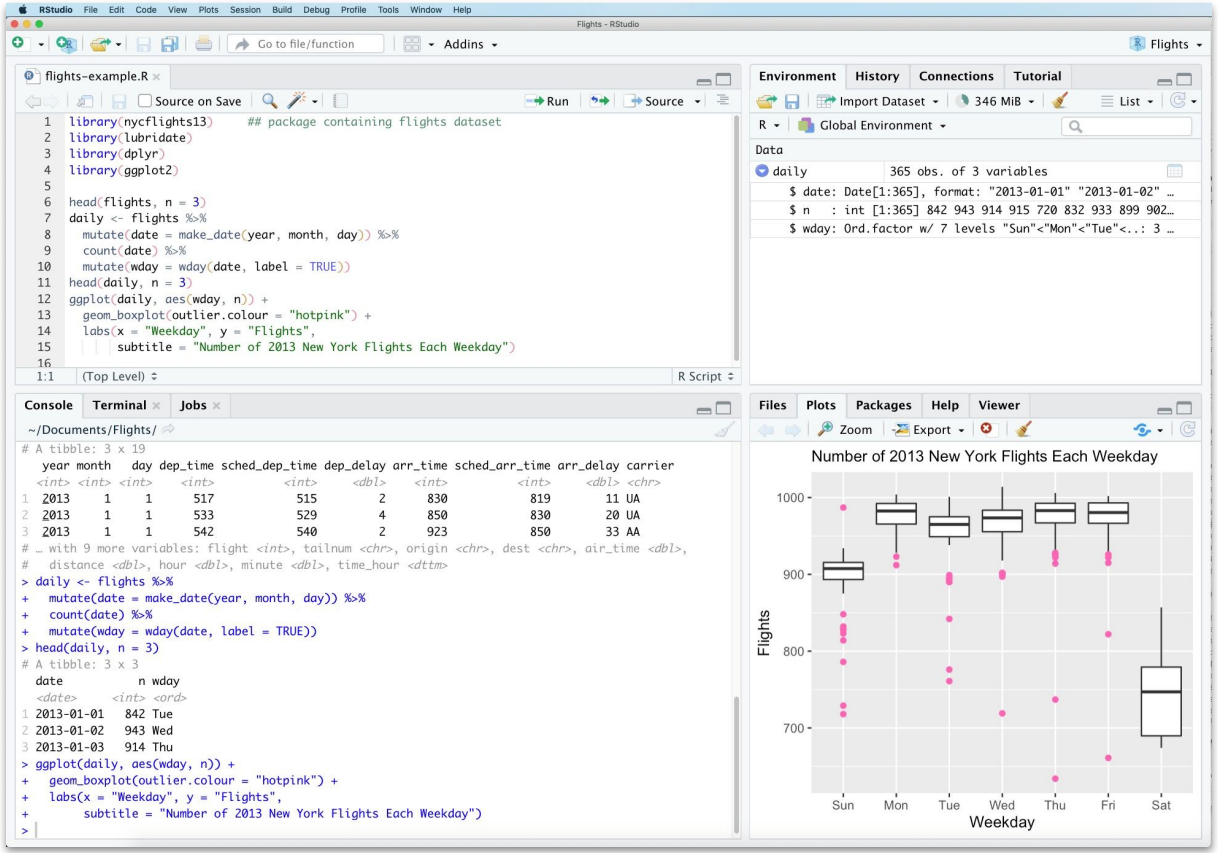
Product Updates



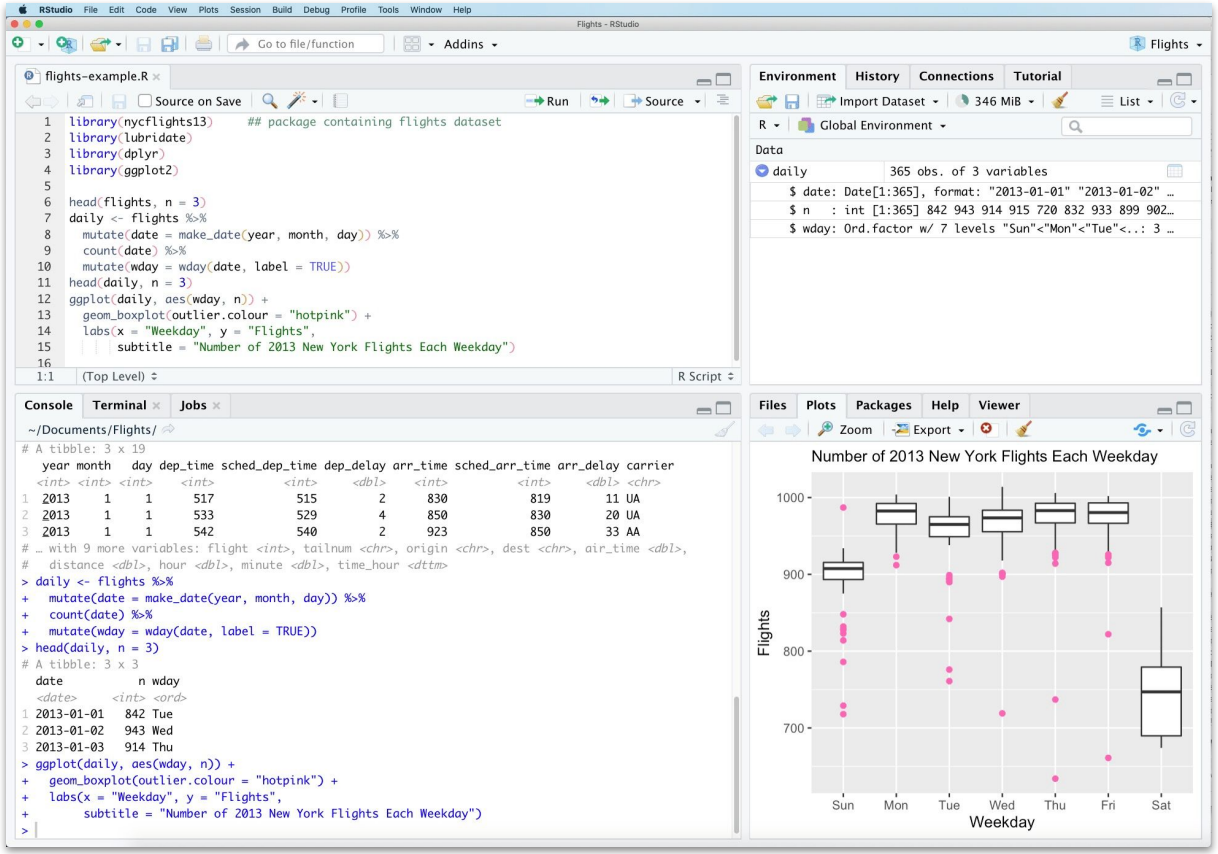
Ryan Johnson
Customer Success Manager

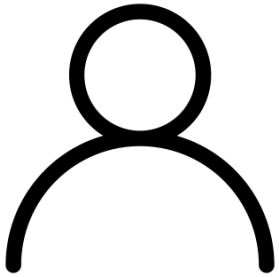
Image modified from “Evolution” by Jakob Vogel from the Noun Project

RStudio

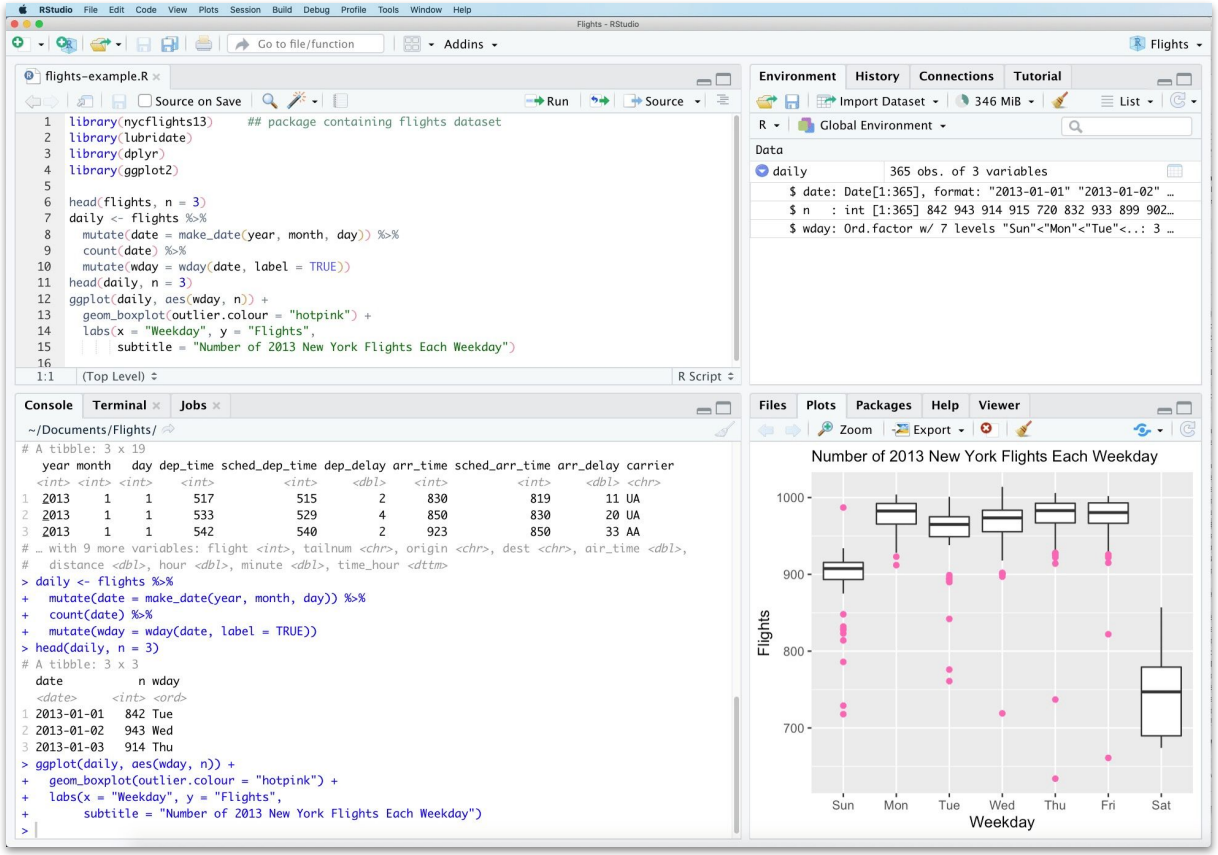

RStudio File Edit Code View Plots Session Build Debug Profile Tools Window Help
Flights - RStudio
Go to file/function
Addins
Flights
flights-example.R
Source on Save
Run
Source
1 library(nycflights13) ## package containing flights dataset
2 library(lubridate)
3 library(dplyr)
4 library(ggplot2)
5
6 head(flights, n = 3)
7 daily <- flights %>%
8 mutate(date = make_date(year, month, day)) %>%
9 count(date) %>%
10 mutate(wday = wday(date, label = TRUE))
11 head(daily, n = 3)
12 ggplot(daily, aes(wday, n)) +
13 geom_boxplot(outlier.colour = "hotpink") +
14 labs(x = "Weekday", y = "Flights",
15 subtitle = "Number of 2013 New York Flights Each Weekday")
16
1:1 (Top Level) R Script
Environment History Connections Tutorial
Import Dataset 346 MiB List
R Global Environment
Data
daily 365 obs. of 3 variables
$ date: Date[1:365], format: "2013-01-01" "2013-01-02" …
$ n : int [1:365] 842 943 914 915 720 832 933 899 902…
$ wday: Ord.factor w/ 7 levels "Sun"<"Mon"<"Tue"<..: 3 …
Console Terminal Jobs
~/Documents/Flights/
A tibble: 3 x 19
year month day dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay carrier
<int> <int> <int> <int> <int> <dbl> <int> <int> <dbl> <chr>
1 2013 1 1 517 515 2 830 819 11 UA
2 2013 1 1 533 529 4 850 830 20 UA
3 2013 1 1 542 540 2 923 850 33 AA
… with 9 more variables: flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
> daily <- flights %>%
+ mutate(date = make_date(year, month, day)) %>%
+ count(date) %>%
+ mutate(wday = wday(date, label = TRUE))
> head(daily, n = 3)
A tibble: 3 x 3
date n wday
<date> <int> <ord>
1 2013-01-01 842 Tue
2 2013-01-02 943 Wed
3 2013-01-03 914 Thu
> ggplot(daily, aes(wday, n)) +
+ geom_boxplot(outlier.colour = "hotpink") +
+ labs(x = "Weekday", y = "Flights",
+ subtitle = "Number of 2013 New York Flights Each Weekday")
>
Files Plots Packages Help Viewer
Zoom Export
Number of 2013 New York Flights Each Weekday
1000
900
800
700
Flights
Sun Mon Tue Wed Thu Fri Sat
Weekday

RStudio®

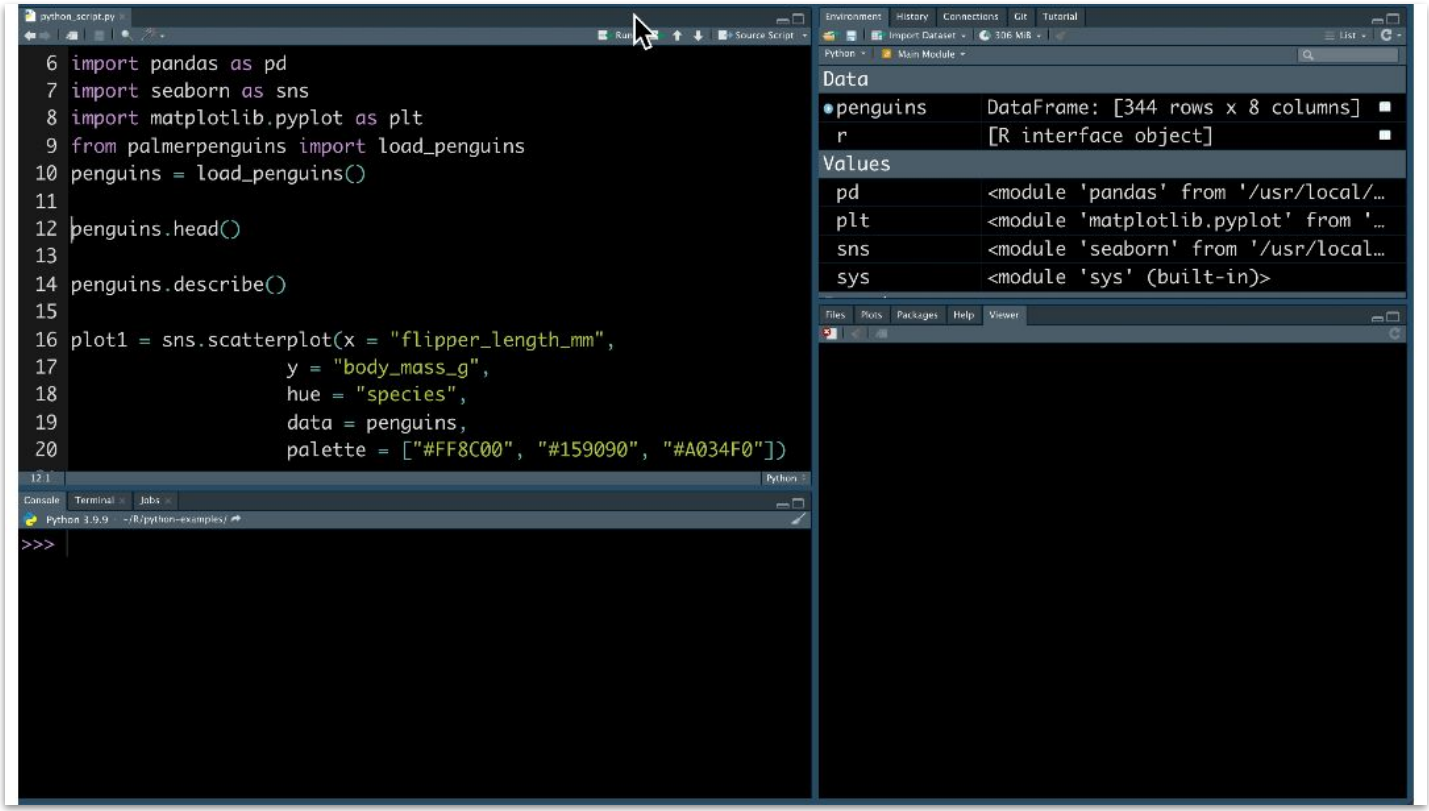

RStudio File Edit Code View Plots Session Build Debug Profile Tools Window Help
Flights - RStudio
Go to file/function
Addins
Flights
flights-example.R
Source on Save
Run
Source
1 library(nycflights13) ## package containing flights dataset
2 library(lubridate)
3 library(dplyr)
4 library(ggplot2)
5
6 head(flights, n = 3)
7 daily <- flights %>%
8 mutate(date = make_date(year, month, day)) %>%
9 count(date) %>%
10 mutate(wday = wday(date, label = TRUE))
11 head(daily, n = 3)
12 ggplot(daily, aes(wday, n)) +
13 geom_boxplot(outlier.colour = "hotpink") +
14 labs(x = "Weekday", y = "Flights",
15 subtitle = "Number of 2013 New York Flights Each Weekday")
16
1:1 (Top Level) R Script
Environment History Connections Tutorial
Import Dataset 346 MiB List
R Global Environment
Data
daily 365 obs. of 3 variables
$ date: Date[1:365], format: "2013-01-01" "2013-01-02" …
$ n : int [1:365] 842 943 914 915 720 832 933 899 902…
$ wday: Ord.factor w/ 7 levels "Sun"<"Mon"<"Tue"<..: 3 …
Console Terminal Jobs
~/Documents/Flights/
A tibble: 3 x 19
year month day dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay carrier
<int> <int> <int> <int> <int> <dbl> <int> <int> <dbl> <chr>
1 2013 1 1 517 515 2 830 819 11 UA
2 2013 1 1 533 529 4 850 830 20 UA
3 2013 1 1 542 540 2 923 850 33 AA
… with 9 more variables: flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
> daily <- flights %>%
+ mutate(date = make_date(year, month, day)) %>%
+ count(date) %>%
+ mutate(wday = wday(date, label = TRUE))
> head(daily, n = 3)
A tibble: 3 x 3
date n wday
<date> <int> <ord>
1 2013-01-01 842 Tue
2 2013-01-02 943 Wed
3 2013-01-03 914 Thu
> ggplot(daily, aes(wday, n)) +
+ geom_boxplot(outlier.colour = "hotpink") +
+ labs(x = "Weekday", y = "Flights",
+ subtitle = "Number of 2013 New York Flights Each Weekday")
>
Files Plots Packages Help Viewer
Zoom Export
Number of 2013 New York Flights Each Weekday
1000
900
800
700
Flights
Sun Mon Tue Wed Thu Fri Sat
Weekday

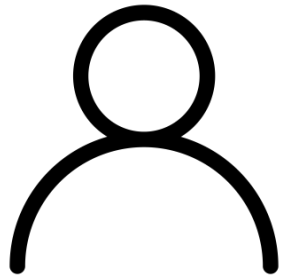


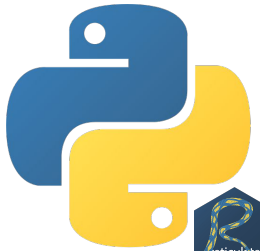
RStudio®


flights-example.R
library(nycflights13) ## package containing flights dataset
library(lubridate)
library(dplyr)
library(ggplot2)
head(flights, n = 3)
daily <- flights %>%
mutate(date = make_date(year, month, day)) %>%
count(date) %>%
mutate(wday = wday(date, label = TRUE))
head(daily, n = 3)
ggplot(daily, aes(wday, n)) +
geom_boxplot(outlier.colour = "hotpink") +
labs(x = "Weekday", y = "Flights",
subtitle = "Number of 2013 New York Flights Each Weekday")
Environment
History
Connections
Tutorial
daily
365 obs. of 3 variables
$ date: Date[1:365], format: "2013-01-01" "2013-01-02" …
$ n : int [1:365] 842 943 914 915 720 832 933 899 902…
$ wday: Ord.factor w/ 7 levels "Sun"<"Mon"<"Tue"<..: 3 …
Console
Terminal
Jobs
~/Documents/Flights/
A tibble: 3 x 19
year month day dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay carrier
2013 1 1 517 515 2 830 819 11 UA
2013 1 1 533 529 4 850 830 20 UA
2013 1 1 542 540 2 923 850 33 AA
… with 9 more variables: flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
> daily <- flights %>%
+ mutate(date = make_date(year, month, day)) %>%
+ count(date) %>%
+ mutate(wday = wday(date, label = TRUE))
> head(daily, n = 3)
A tibble: 3 x 3
date n wday
2013-01-01 842 Tue
2013-01-02 943 Wed
2013-01-03 914 Thu
> ggplot(daily, aes(wday, n)) +
+ geom_boxplot(outlier.colour = "hotpink") +
+ labs(x = "Weekday", y = "Flights",
+ subtitle = "Number of 2013 New York Flights Each Weekday")
Files
Plots
Packages
Help
Viewer
Number of 2013 New York Flights Each Weekday
1000
900
800
700
Flights
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Weekday

RStudio®

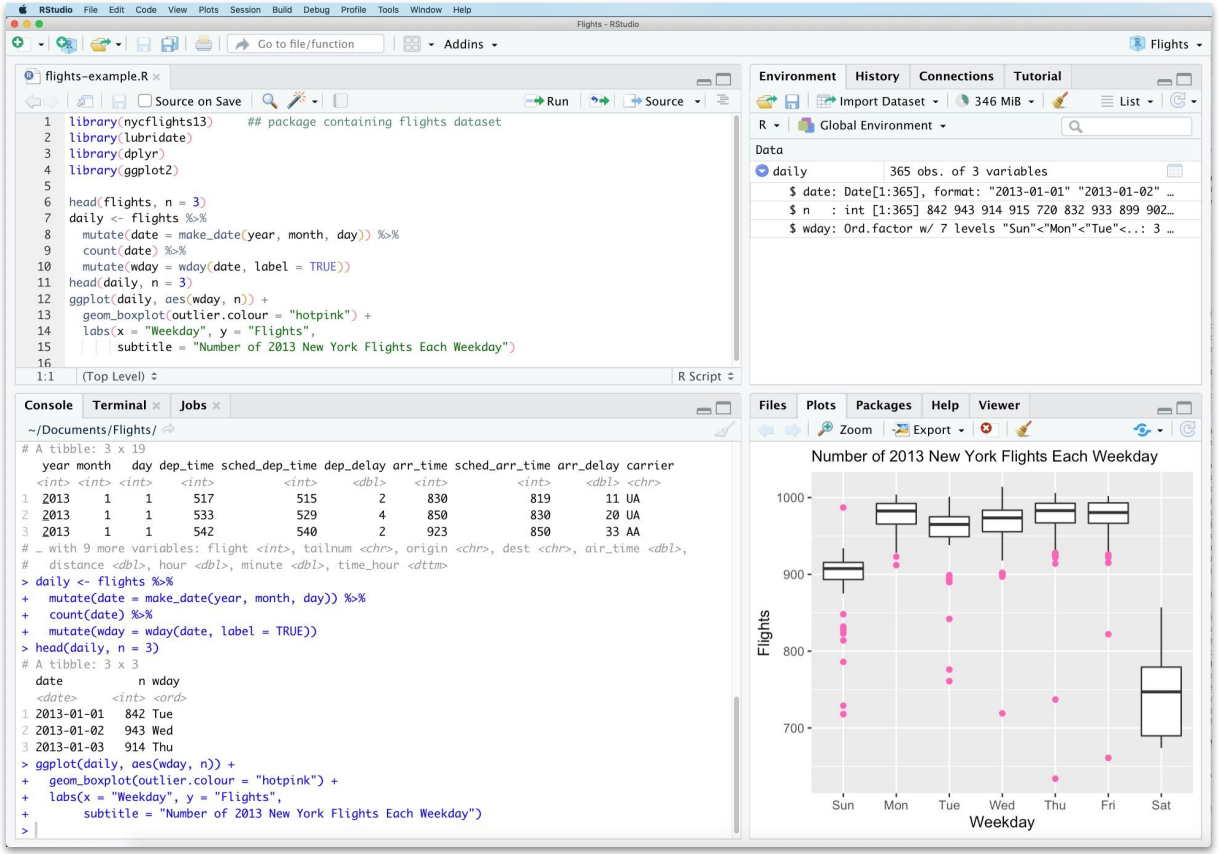

python_script.py
Run
Source Script
6 import pandas as pd
7 import seaborn as sns
8 import matplotlib.pyplot as plt
9 from palmerpenguins import load_penguins
10 penguins = load_penguins()
11
12 penguins.head()
13
14 penguins.describe()
15
16 plot1 = sns.scatterplot(x = "flipper_length_mm",
17 y = "body_mass_g",
18 hue = "species",
19 data = penguins,
20 palette = ["#FF8C00", "#159090", "#A034F0"])
Console
Terminal
Jobs
Python 3.9.9 ~/R/python-examples/
>>>
Environment
History
Connections
Git
Tutorial
Import Dataset
306 MiB
List
Python
Main Module
Data
penguins DataFrame: [344 rows x 8 columns]
r [R interface object]
Values
pd <module 'pandas' from '/usr/local/…
plt <module 'matplotlib.pyplot' from '…
sns <module 'seaborn' from '/usr/local…
sys <module 'sys' (built-in)>
Files
Plots
Packages
Help
Viewer

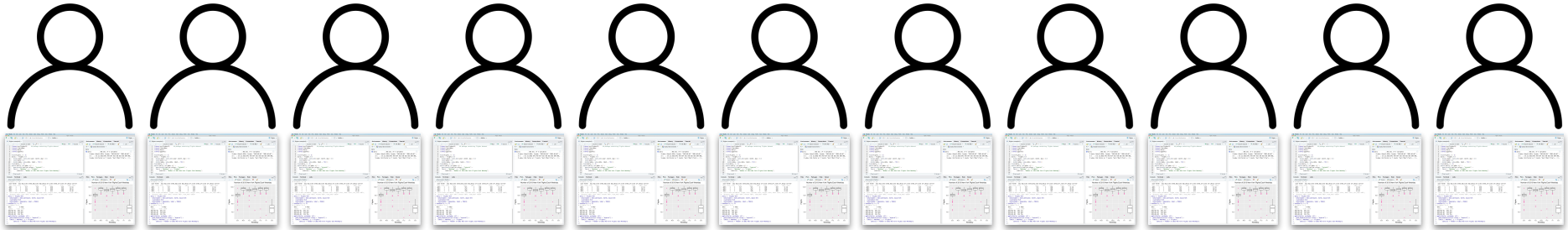


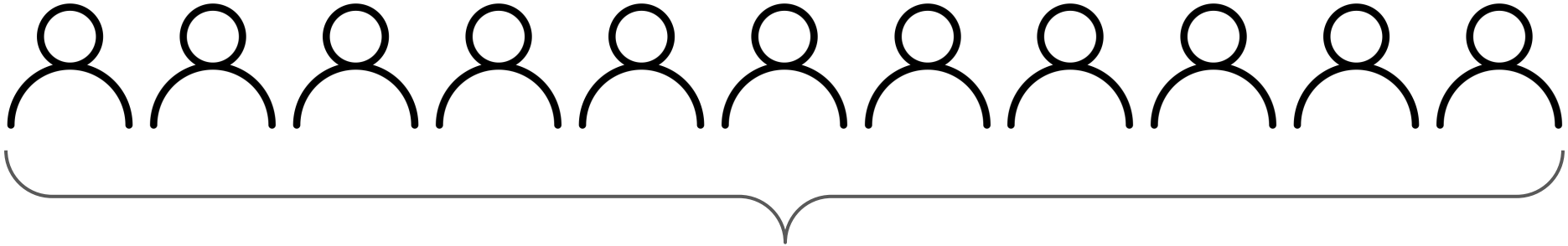


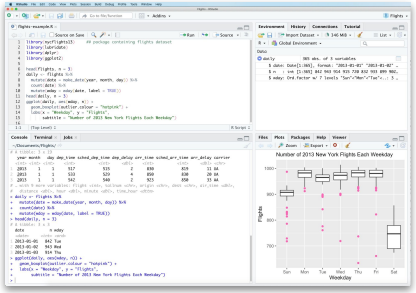
reticulate

RStudio®

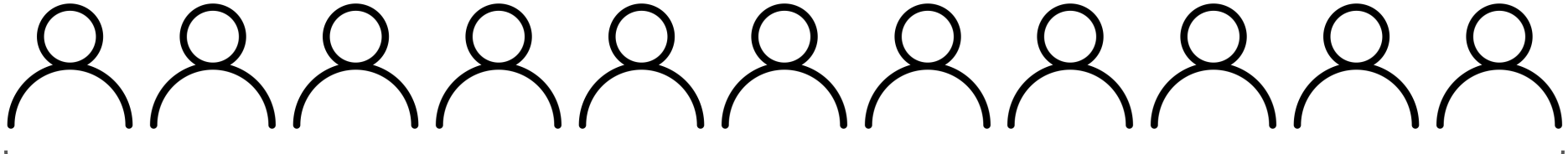

flights-example.R
library(nycflights13) ## package containing flights dataset
library(lubridate)
library(dplyr)
library(ggplot2)
head(flights, n = 3)
daily <- flights %>%
mutate(date = make_date(year, month, day)) %>%
count(date) %>%
mutate(wday = wday(date, label = TRUE))
head(daily, n = 3)
ggplot(daily, aes(wday, n)) +
geom_boxplot(outlier.colour = "hotpink") +
labs(x = "Weekday", y = "Flights",
subtitle = "Number of 2013 New York Flights Each Weekday")
Environment
History
Connections
Tutorial
daily
365 obs. of 3 variables
$ date: Date[1:365], format: "2013-01-01" "2013-01-02" …
$ n : int [1:365] 842 943 914 915 720 832 933 899 902…
$ wday: Ord.factor w/ 7 levels "Sun"<"Mon"<"Tue"<..: 3 …
Console
Terminal
Jobs
~/Documents/Flights/
A tibble: 3 x 19
year month day dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay carrier
2013 1 1 517 515 2 830 819 11 UA
2013 1 1 533 529 4 850 830 20 UA
2013 1 1 542 540 2 923 850 33 AA
… with 9 more variables: flight <int>, tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>,
distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
> daily <- flights %>%
+ mutate(date = make_date(year, month, day)) %>%
+ count(date) %>%
+ mutate(wday = wday(date, label = TRUE))
> head(daily, n = 3)
A tibble: 3 x 3
date n wday
2013-01-01 842 Tue
2013-01-02 943 Wed
2013-01-03 914 Thu
> ggplot(daily, aes(wday, n)) +
+ geom_boxplot(outlier.colour = "hotpink") +
+ labs(x = "Weekday", y = "Flights",
+ subtitle = "Number of 2013 New York Flights Each Weekday")
Files
Plots
Packages
Help
Viewer
Number of 2013 New York Flights Each Weekday
Flights
Weekday
Sun
Mon
Tue
Wed
Thu
Fri
Sat

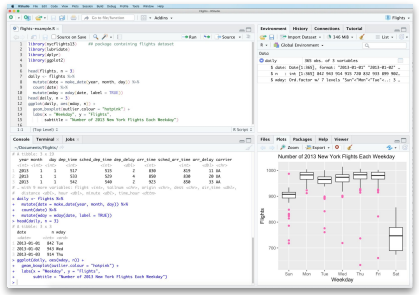




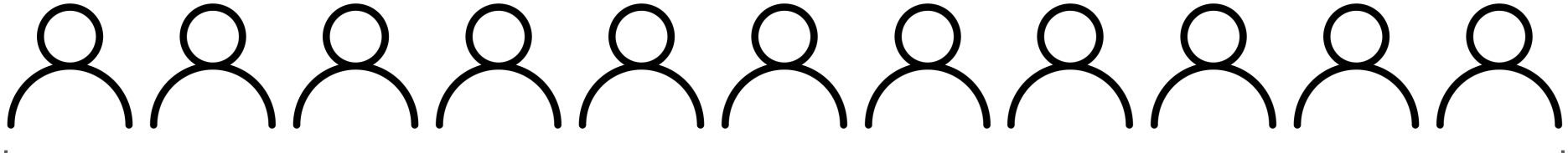


posit™ Workbench

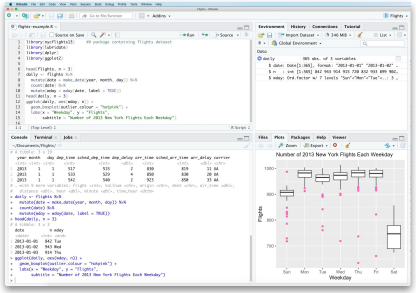




posit™ Workbench



jupyter



posit™ Workbench

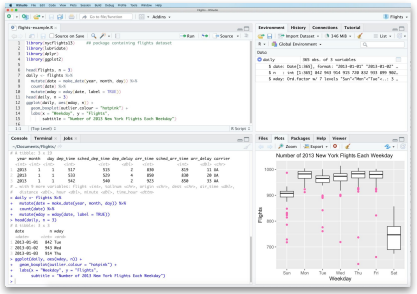
On-Prem

AWS/GCP/Azure

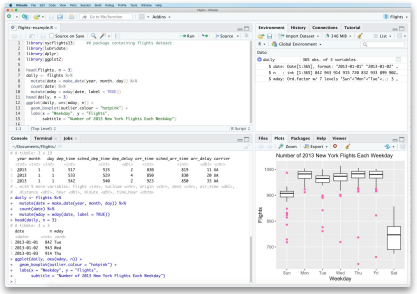
Amazon SageMaker

Azure Machine Learning

posit™ Workbench



posit™ Workbench



Static Content

quarto®

rmarkdown

pins

jupyter

Welcome to Quarto

Quarto® is an open-source scientific and technical publishing system built on Pandoc

- Create dynamic content with Python, R, Julia, and Observable.
- Author documents as plain text markdown or Jupyter notebooks.
- Publish high-quality articles, reports, presentations, websites, blogs, and books in HTML, PDF, MS Word, ePub, and more.
- Author with scientific markdown, including equations, citations, crossrefs, figure panels, callouts, advanced layout, and more.

Get Started | Guide

Hello, Quarto

Python | R | Julia | Observable

Weave together narrative text and code to produce elegantly formatted output. Quarto documents are fully reproducible. Use markdown with code cells executed via Jupyter (shown below) or render existing Jupyter notebooks.

```
---
title: "matplotlib demo"
format:
  html:
    code-fold: true
jupyter: python3
---

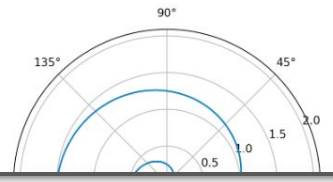
For a demonstration of a line plot on a polar axis,
see @fig-polar.

```{python}
#| label: fig-polar
#| fig-cap: "A line plot on a polar axis"
```

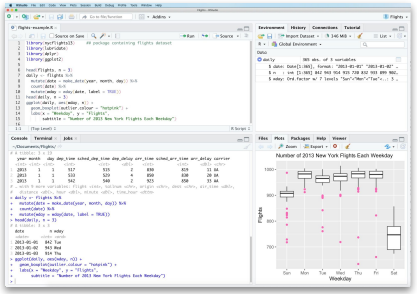
matplotlib demo

For a demonstration of a line plot on a polar axis, see Figure 1.

▶ Code


```

posit™ Workbench



Static Content

quarto

rmarkdown

pins

jupyter

Applications

Shiny

Streamlit

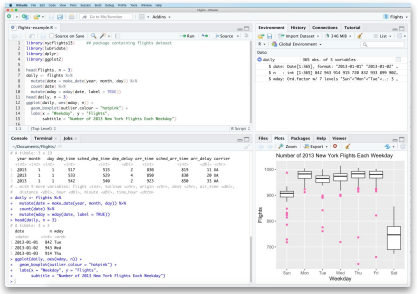
bokeh

Dash

Flask

jupyter

posit™ Workbench



Static Content

quarto

rmarkdown

pins

jupyter

Applications

Shiny

R

Streamlit

bokeh

Dash

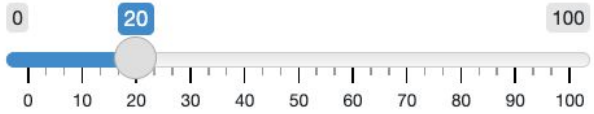
Flask

for

app.py

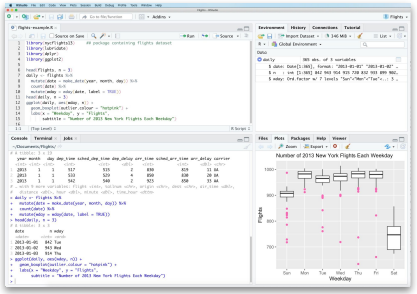
```python
from shiny import App, render, ui

app_ui = ui.page_fluid(
    ui.input_slider("n", "N", 0, 100, 20),
    ui.output_text_verbatim("txt"),
)


def server(input, output, session):
    @output
    @render.text
    def txt():
        return f"n*2 is {input.n() * 2}"


app = App(app_ui, server)

```

N




```
n*2 is 40
```

posit Workbench



Static Content

quarto

rmarkdown

pins

jupyter

Applications

Shiny

R

Streamlit

bokeh

Dash

Flask

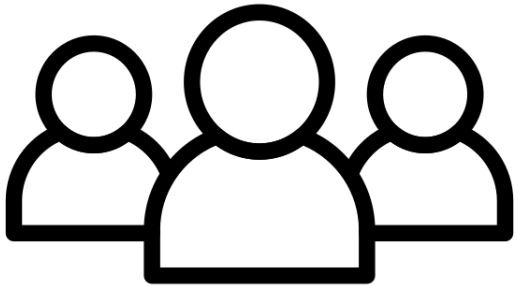
APIs

plumber

Flask

FastAPI

Data Scientists



bokeh

Streamlit

jupyter

quarto

Dash

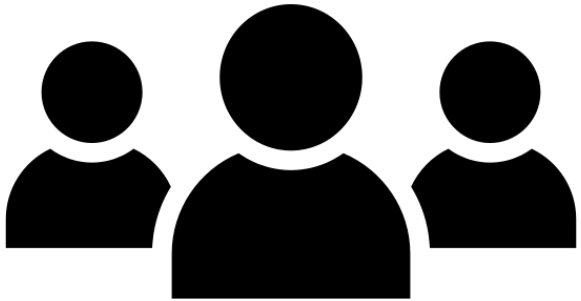
Flask

Flask

FastAPI

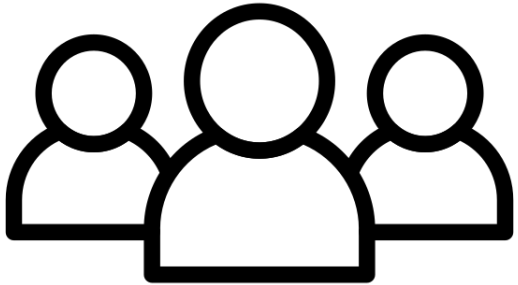
?

End Users

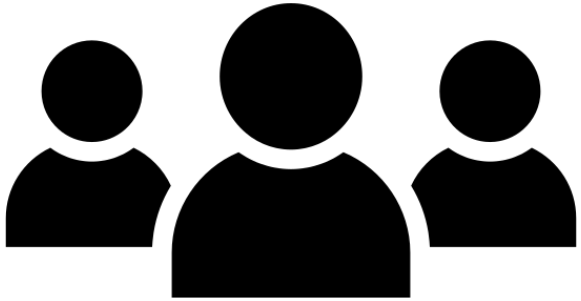


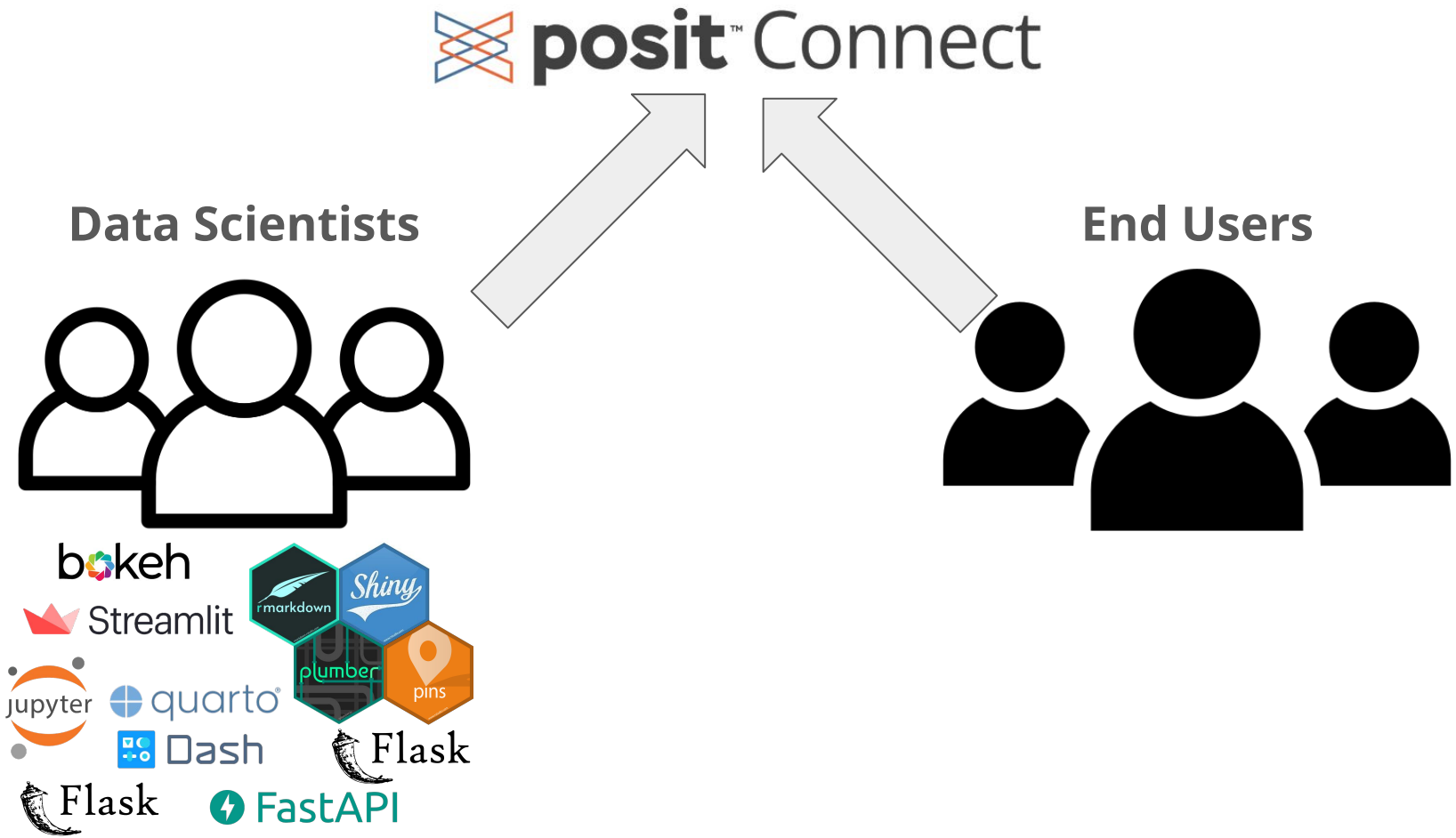
posit™ Connect

Data Scientists



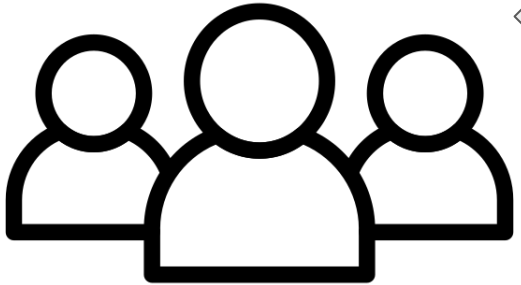
End Users




posit Connect
Data Scientists
End Users
bokeh
Streamlit
jupyter
quarto
Dash
Flask
Flask
FastAPI
rmarkdown
Shiny
plumber
pins

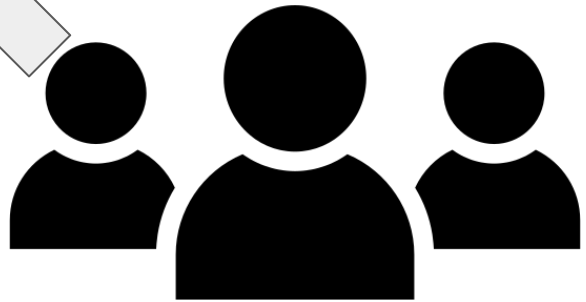
posit™ Connect

Data Scientists



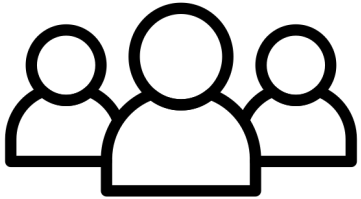
bokeh
Streamlit
jupyter
quarto
Dash
Flask
Flask
FastAPI

End Users



- Colleagues
- Business Leaders
- Customers
- General Public

Data Scientists

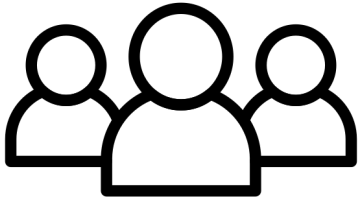


posit™ Workbench

posit™ Connect

Packages!

Data Scientists



posit™ Workbench

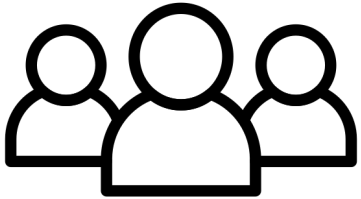
posit™ Connect

NumPy

matplotlib

seaborn

Data Scientists



posit™ Workbench

posit™ Connect

CRAN

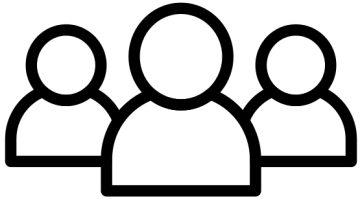
The Comprehensive R Archive Network

Bioconductor

OPEN SOURCE SOFTWARE FOR BIOINFORMATICS

python™ Package Index

Data Scientists



posit™ Workbench

posit™ Connect

CRAN

The Comprehensive R Archive Network

Bioconductor

OPEN SOURCE SOFTWARE FOR BIOINFORMATICS

python™ Package Index

posit™ Package Manager


posit™ Workbench


posit™ Connect


posit™ Package Manager


posit™ Team
posit™ Workbench
posit™ Connect
posit™ Package Manager

Want to learn more about:

posit™ Team?

Lauren Chadwick
Senior Account Executive

lauren@posit.co

Ryan Johnson
Customer Success Manager

ryan@posit.co

Want to learn more about:

posit™ Team?

Lauren Chadwick
Senior Account Executive

lauren@posit.co

Ryan Johnson
Customer Success Manager

ryan@posit.co

Data Science?

posit™ Academy

posit™ Cloud

posit™ Academy

Lessons
Learn the fundamentals for the project with interactive lessons.

Project milestones
Each week, recreate a progressive piece of the project and expand on it.

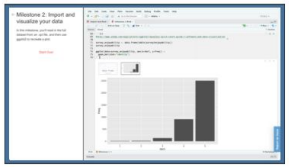
Group sessions
Share your work and ideas with your group at weekly sessions.

Mentor meetings
Meet regularly with a mentor to get personalized guidance.

Daily practice
Build long-lasting habits with spaced-repetition drills.

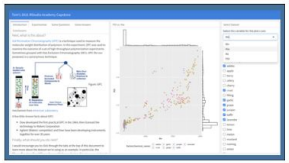
Tailored curriculum

Traditional data science learning is generic and broken. With Academy, content is immersive and tailored to fit your team's exact needs. Learn R & Python with specific use cases, data, and best practices, while completing projects relevant to your organization's daily work.



Practice-based learning

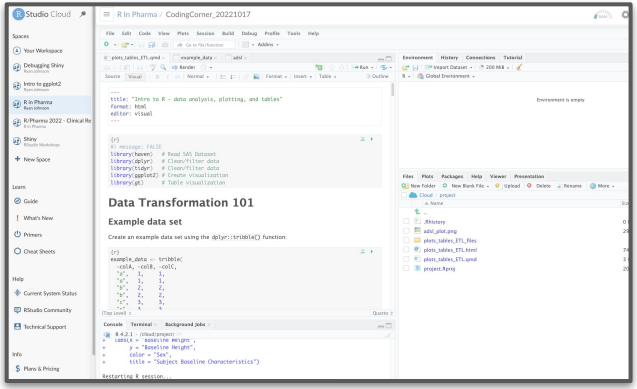
Academy focuses on adaptive practice-based learning and spaced repetition so that the individual learner can master the relevant skills. Learners work on collaborative projects and apply facts to solve real-world problems from the very start.

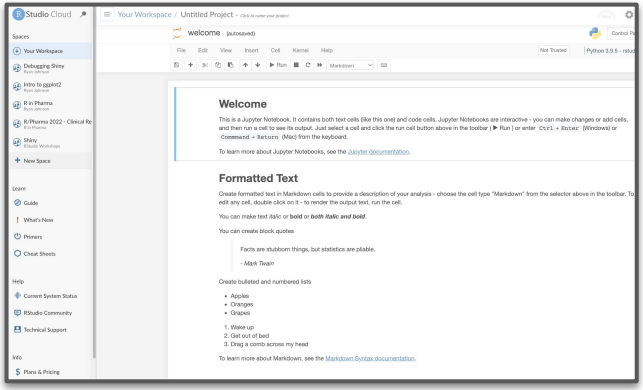


Familiar Tools

Academy leverages the same tools that so many enterprises are already familiar with for scaling open source data science, all hosted and managed on the platform. Learners will have the confidence to deliver immediate insights with those tools after the Academy experience.

posit™ Cloud





RStudio®

jupyter

Data science without the hardware hassles!

- Lightweight, cloud-based solution that allows anyone to **do, share, teach and learn data science online!**
- Nothing to configure
- No dedicated hardware
- No installation
- No annual purchase contract required

Want to learn more about:

posit™ Team?

Lauren Chadwick
Senior Account Executive

lauren@posit.co

Ryan Johnson
Customer Success Manager

ryan@posit.co

Data Science?

posit™ Academy

posit™ Cloud