RStudio®

Open source & enterprise-ready professional software for R

Using Python with RStudio Connect

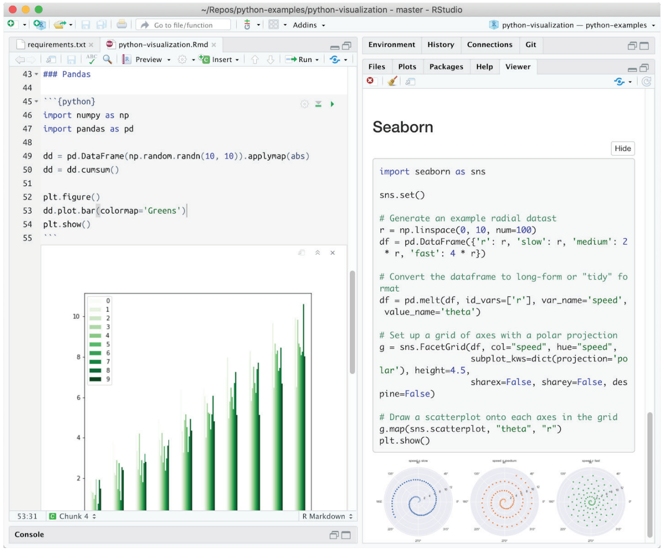
Python and R are popular data science languages for analyzing data. RStudio Connect (v1.7.0 and higher) provides connectivity to Python for teams and individuals who use both languages:

- Data scientists who use a combination of R and Python
- RStudio users working together with Jupyter Notebook users

Publish Projects that use Python and R from RStudio to RStudio Connect

Use the reticulate library in R scripts, Shiny apps, R Markdown, Plumber APIs to integrate existing Python code and libraries for interactive exploration (pandas), visualization (matplotlib, seaborn), and machine learning (PyTorch, scikit-learn, statsmodels) and publish them to RStudio Connect.

Configure your project to use a specific version of Python in the RStudio IDE or RStudio Server Pro using the **`RETICULATE_PYTHON`** environment variable. RStudio Connect supports multiple versions of Python that can be used by published applications.



~~~
### Pandas

```{python}
import numpy as np
import pandas as pd

dd = pd.DataFrame(np.random.randn(10, 10)).applymap(abs)
dd = dd.cumsum()

plt.figure()
dd.plot.bar(colormap='Greens')
plt.show()
```
~~~

Seaborn

```
import seaborn as sns

sns.set()

# Generate an example radial datast
r = np.linspace(0, 10, num=100)
df = pd.DataFrame({'r': r, 'slow': r, 'medium': 2
 * r, 'fast': 4 * r})

# Convert the dataframe to long-form or "tidy" fo
rmat
df = pd.melt(df, id_vars=['r'], var_name='speed',
 value_name='theta')

# Set up a grid of axes with a polar projection
g = sns.FacetGrid(df, col="speed", hue="speed",
                  subplot_kws=dict(projection='po
lar'), height=4.5,
                  sharex=False, sharey=False, des
pine=False)

# Draw a scatterplot onto each axes in the grid
g.map(sns.scatterplot, "theta", "r")
plt.show()
```

Publish Notebooks to RStudio Connect from Jupyter or JupyterHub

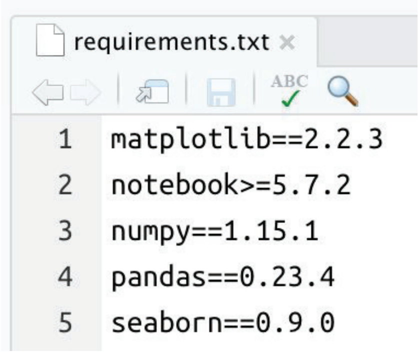
Publish notebooks to RStudio Connect with the **`rsconnect-jupyter`** notebook extension.

Configure Jupyter Notebooks to publish notebooks to RStudio Connect using the following steps from the **`rsconnect-jupyter`** documentation:

- Download and install the **`rsconnect-jupyter`** package in your Python environment
- Enable the rsconnect jupyter extension
- Generate an API token from RStudio Connect
- Use one-button publishing to send your Jupyter Notebooks to RStudio Connect
- Schedule email reports or control access to Jupyter Notebooks similar to other deployed applications

Resources:

- RStudio Connect with Python: http://docs.rstudio.com/connect/1.7.0/admin/python.html
- Python examples: https://github.com/sol-eng/python-examples
- Jupyter Notebook extension: http://docs.rstudio.com/rsconnect-jupyter/
- Reticulate library (R Interface to Python): https://rstudio.github.io/reticulate/

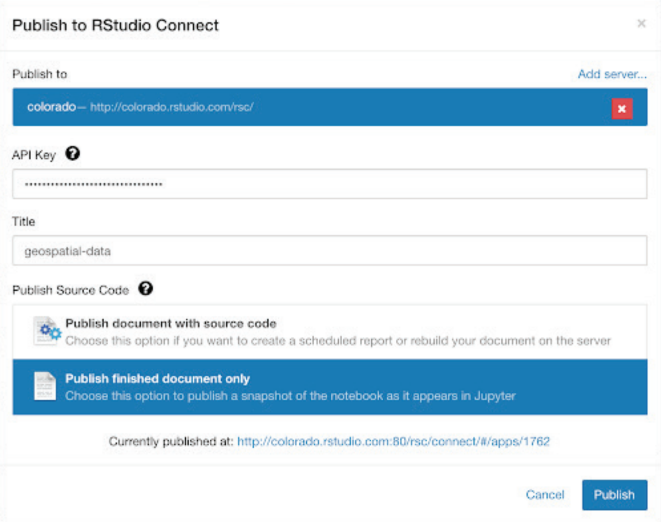


```
matplotlib==2.2.3
notebook>=5.7.2
numpy==1.15.1
pandas==0.23.4
seaborn==0.9.0
```

Publish Jupyter Notebooks

Use an extension for push-button publishing of Jupyter Notebooks to RStudio Connect

- Ability to publish rendered notebook
- Or publish notebook with source code (Python only)
- Handle Python dependencies at publish time



Publish to RStudio Connect ×

Publish to — Add server...

colorado — http://colorado.rstudio.com/rsc/

API Key

••••••••••••••••••••••••••••••••

Title

geospatial-data

Publish Source Code

Publish document with source code
Choose this option if you want to create a scheduled report or rebuild your document on the server

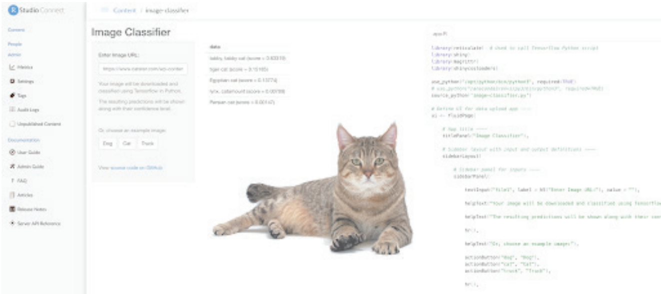
Publish finished document only
Choose this option to publish a snapshot of the notebook as it appears in Jupyter

Currently published at: http://colorado.rstudio.com:80/rsc/connect/#/apps/1762

Build Shiny apps that use Python

Build interactive applications and dashboards on top of existing Python code and libraries

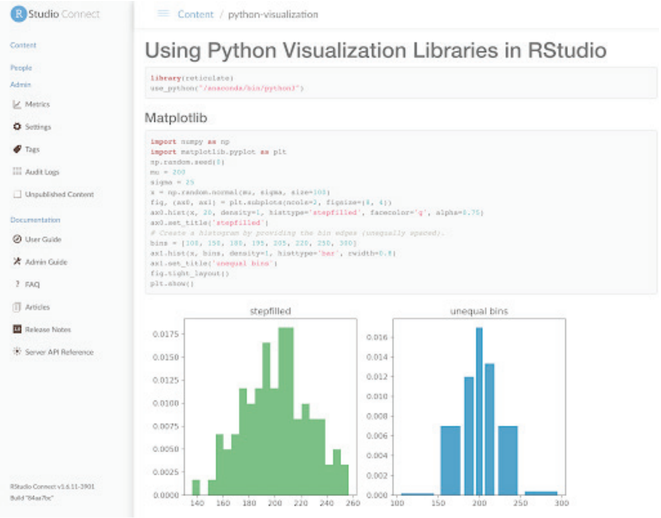
- Call functions in existing Python scripts directly from Shiny apps
- Execute Python scripts as a Shiny app is being interacted with



Use Python in R Markdown Documents

Use Python with R for interactive, exploratory analyses in notebooks or mixed Python and R content in documents and reports

- Import Python libraries to use in R Markdown documents
- Call functions in existing Python scripts



Using Python Visualization Libraries in RStudio

```
library(reticulate)
use_python("/anaconda/bin/python3")
```

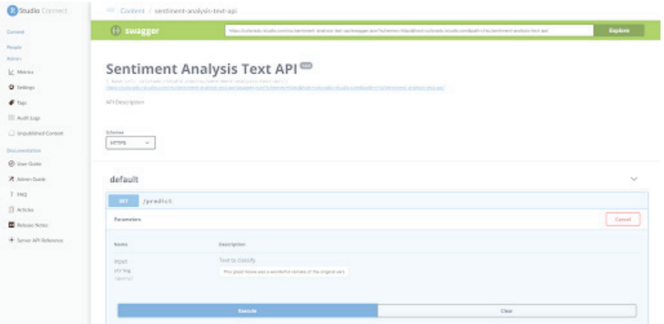
Matplotlib

```
import numpy as np
import matplotlib.pyplot as plt
np.random.seed(0)
mu = 200
sigma = 25
x = np.random.normal(mu, sigma, size=100)
fig, (ax0, ax1) = plt.subplots(ncols=2, figsize=(8, 4))
ax0.hist(x, 20, density=1, histtype='stepfilled', facecolor='g', alpha=0.75)
ax0.set_title('stepfilled')
# Create a histogram by providing the bin edges (unequally spaced).
bins = [100, 150, 180, 195, 205, 220, 250, 300]
ax1.hist(x, bins, density=1, histtype='bar', rwidth=0.8)
ax1.set_title('unequal bins')
fig.tight_layout()
plt.show()
```

Deploy Plumber APIs that use Python

Deploy APIs that combine Python and R code that is executed behind an API endpoint

- Execute Python scripts or use Python libraries when an API is queried



About RStudio

RStudio® makes data analysis with R easier and provides powerful tools for sharing reports, dashboards, and interactive Shiny® applications with your entire enterprise.

| rstudio.com | 844-448-1212 | info@rstudio.com