TQMP ¦ 2018 ■ Vol. 14 ■ no. 2

Welcome to Jupyter: Improving Collaboration and Reproduction in Psychological Research by Using a Notebook System

Philipp Sprengholz [a], ✉

[a] Friedrich-Schiller-Universität Jena, Department of Psychology

Abstract ■ The reproduction of findings from psychological research has been proven difficult. Abstract description of the data analysis steps performed by researchers is one of the main reasons why reproducing or even understanding published findings is so difficult. With the introduction of Jupyter, a new tool for the organization of both static and dynamic information became available. The software allows blending explanatory content like written text or images with code for preprocessing and analyzing scientific data. Thus, Jupyter helps documenting the whole research process from ideation over data analysis to the interpretation of results. This fosters both collaboration and scientific quality by helping researchers to organize their work. This tutorial is an introduction to Jupyter. It explains how to setup and use the notebook system. While introducing its key features, the advantages of using Jupyter notebooks for psychological research become obvious.

Keywords ■ Reproducible research, Interactive scientific computing, Collaboration, Notebook systems, Data management. **Tools** ■ Jupyter, R.

Acting Editor ■ Denis Cousineau (Université d'Ottawa)

Reviewers ■ Vincent LeBlanc (Université d'Ottawa)

✉ philipp.sprengholz@outlook.com

iD *PS*: 0000-0002-9925-1920

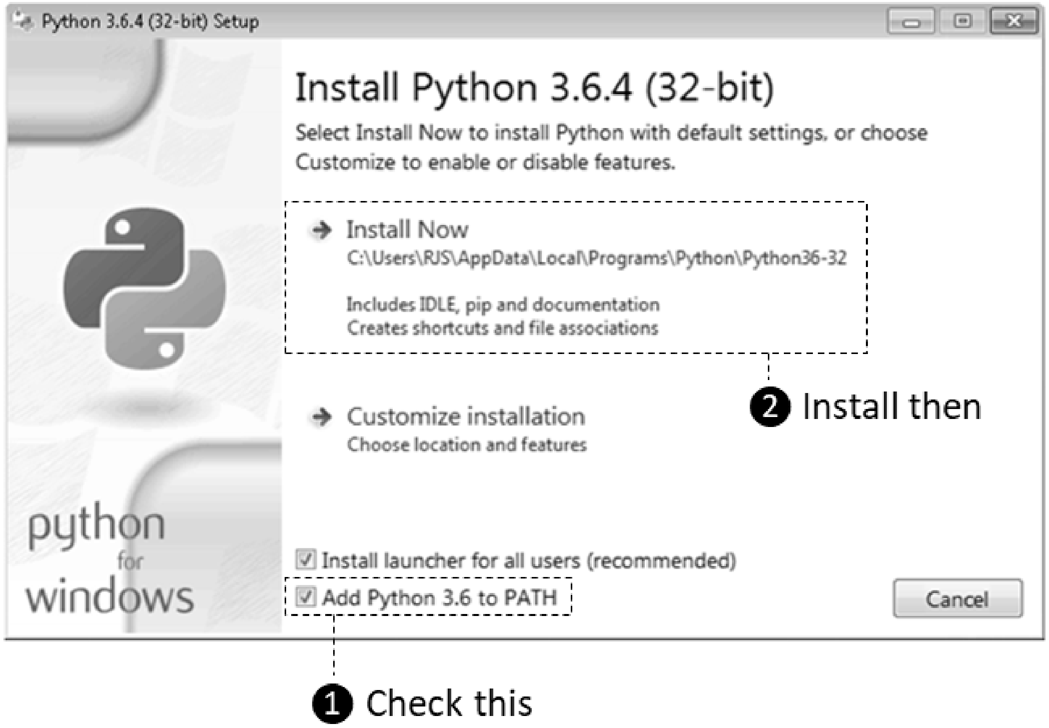
doi 10.20982/tqmp.14.2.p137

Introduction

The replicability of psychological research has been questioned increasingly (Klein et al., 2014; Pashler & Wagenmakers, 2012; Yong, 2012). Reproducing or even understanding research findings requires extensive knowledge about the experimental manipulations and methods used (Nosek, Spies, & Motyl, 2012). Unfortunately, many research publications fail in describing the research process in detail, are difficult to understand without background information or facilitate misinterpretation (Donoho, Maleki, Rahman, Shahram, & Stodden, 2009, 1). Most articles only include very abstract descriptions of data preparation and analysis steps, making it hard for the reader to follow up on. Consequently, reproducing results from psychological journals is merely possible (Shen, 2014). The scientific community tries to solve these problems by publishing supplemental information online. This includes raw data as well as detailed descriptions of data preprocessing and analysis steps. Unfortunately, this information is often organized in a confusing way. That's why a group of scientists developed Jupyter, a web application based on IPython (Perez & Granger, 2007). Jupyter enables users to create and share notebooks containing text, visualizations, equations, raw data and code for analyzing and transforming this data. By blending static content like explanatory text and images with dynamic output of calculations and data analysis procedures, the notebooks emphasize the prose first approach originally introduced by Mathematica Notebooks more than 20 years ago. The whole research process including ideation, data acquisition, analysis and interpretation of results can be documented in a linear, story-like way. Publishing these notebooks alongside or instead of read-only journal articles may enhance both replication of results and collaboration between researchers.

This tutorial is written for readers with no previous experience using Jupyter. It explains how to setup and

Figure 1 ■ Python installer



Python 3.6.4 (32-bit) Setup

Install Python 3.6.4 (32-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

→ Install Now
C:\Users\RJS\AppData\Local\Programs\Python\Python36-32

Includes IDLE, pip and documentation
Creates shortcuts and file associations

→ Customize installation
Choose location and features

☑ Install launcher for all users (recommended)

☑ Add Python 3.6 to PATH

❷ Install then

❶ Check this

use notebooks for organizing, performing and documenting data analysis tasks common in psychological research. Jupyter supports more than 90 programming languages, thus enabling you to analyze data using scripts written in Python, R or virtually any other non-proprietary scripting language. However, this article will strictly focus on R. After setting up the system, an exemplary notebook will be created step by step.

Setting Up Jupyter

Setting up Jupyter on your local computer includes three steps. At first Python needs to be installed as it is required to run the notebook system. Afterwards Jupyter is downloaded. Finally, R is installed and configured to work with Jupyter. All three steps are detailed in the following. Since most readers are assumed to work on Microsoft Windows, the explanations are tailored to this operating system. However, Jupyter can also be setup on both Mac OS and Linux and the steps to perform are nearly identical.

Step 1: Installing Python

Download the latest Python 3 installer from Python.org (current version is 3.6.4). When starting the installer, use default settings but make sure Python is added to your systems path variable (see Figure 1).

Step 2: Installing Jupyter

After Python has been installed, a command window needs to be opened. Press the Win + R keys on your keyboard, type `cmd` and press Enter. Afterwards enter the following line into the command window and press Enter again: `pip install jupyter`

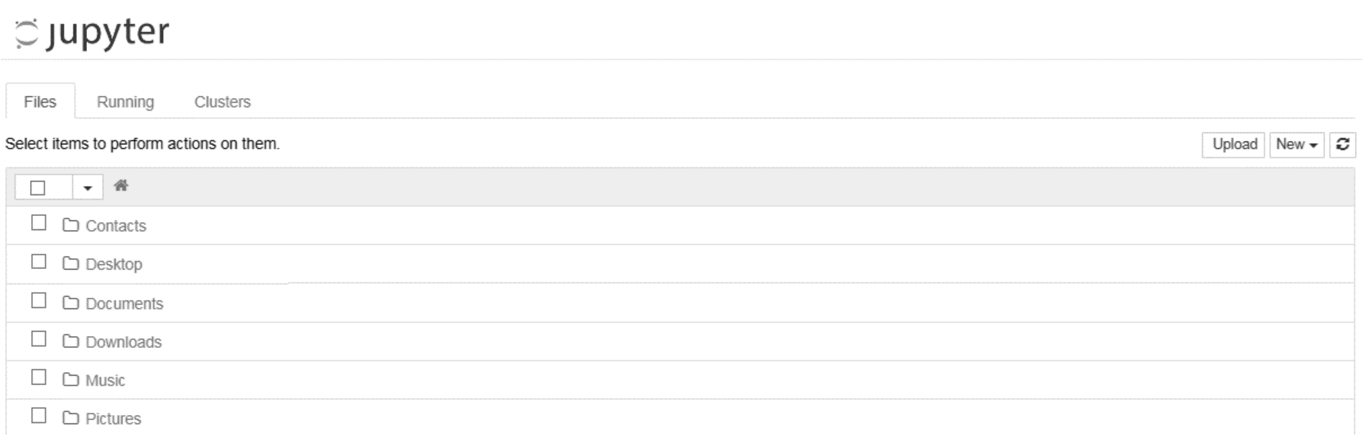
Step 3: Installing R and the R Kernel

Download the latest R installer from R-Project.org (current version is 3.4.4). Make sure to select the base installation for Windows. Run the installer using default settings afterwards.

Finally, Jupyter hast to be interconnected with R by installing the R kernel. Open the R console by starting R.exe (to be found under `C:\Program Files\R\R-3.4.3\bin`). Copy the following command into the console window and press enter:

```
install.packages(c('repr', 'IRdisplay',
  'evaluate', 'crayon', 'pbdZMQ',
  'devtools', 'uuid', 'digest'))
```

Figure 2 ■ Jupyter home screen



Select items to perform actions on them.

□ Contacts
□ Desktop
□ Documents
□ Downloads
□ Music
□ Pictures

This downloads a set of packages required by the R kernel. You may be asked to create a personal library, respond with yes. If you are asked to select a CRAN mirror, select a mirror close to your current location as this accelerates the download. While retrieving the packages, several warnings may be printed in the console window. They can be ignored. After all packages have been downloaded, execute the following command in the R console:

```
devtools::install_github('IRkernel/
    IRkernel')
```

This installs the R kernel. Upcoming warning messages can be ignored again. Afterwards we need to make sure Jupyter identifies the newly installed kernel. Therefore, its spec must be registered by executing the following command in the R console:

```
IRkernel::installspec()
```

Now we are ready to start Jupyter. Close the R console and open a new Windows command window as explained in Step 2. Type

```
jupyter notebook
```

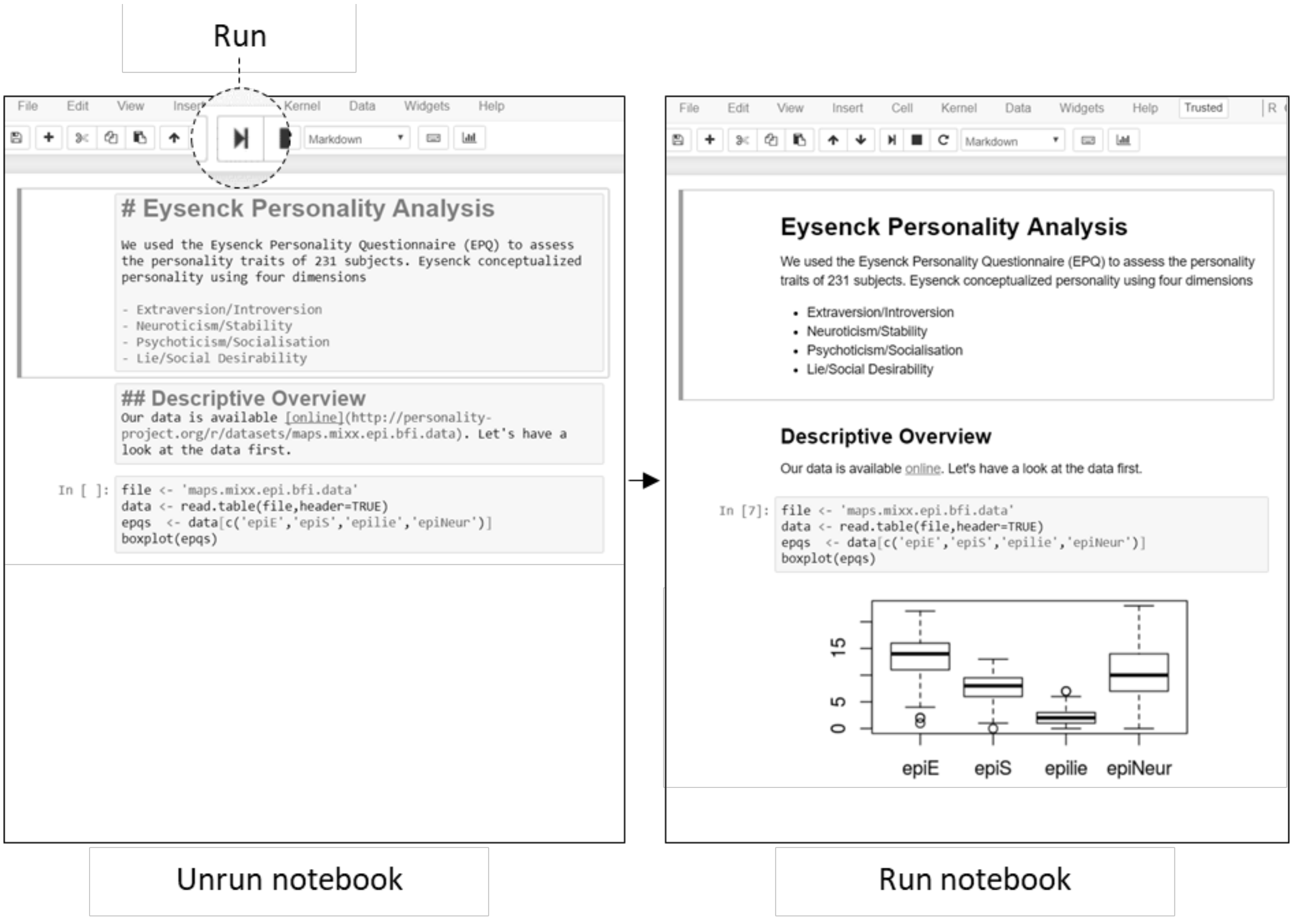
and press Enter. Starting the notebook system may take some seconds. Afterwards a browser window opens, showing Jupyter's homepage (see Figure 2). Congratulations, Jupyter has been set up successfully. In case you want to shut down the notebook system, simply close the command window. Whenever you want to start it up again, open a new command window and repeat executing the `jupyter notebook` command.

Creating and Editing a Notebook

When looking at Jupyter's home screen, you will see your computer's user directory. By default, the notebook app can only access files within this directory and any subfolders. Navigate to a place where you want to store your notebooks. You can create a new folder by clicking `New → Folder` and rename it afterwards by selecting it and clicking Rename. After choosing a folder, create a new notebook in there by clicking `New → Notebook: R`. A new browser window opens, showing the empty notebook you just created. Each notebook is made of vertically ordered cells holding either explanatory content or code. The input of each cell can be interpreted (a.k.a run) by Jupyter, leading to a well formatted output. Figure 3 shows an example. As we can see on the left side, this notebook contains multiple cells. When running them (by pressing the play button at the top of the page), they are rendered as shown on the right side.

In our empty notebook we can easily create new cells by clicking the plus button. Before filling the cells, we have to decide about the type of content. Each cell can be a Markdown or code cell. You can change the cell type by clicking `Cell → Cell Type` in the menu. To get a deeper understanding about the two types, we will use our recently created notebook for the analysis of exemplary Big Five personality data to be retrieved from the Personality Project. First, we will note down some conceptual basics using Markdown cells. Afterwards we will load the data and analyze it using code cells. As the algorithms may require some explanation, code cells should alternate with describing Markdown cells. The final result can be previewed and downloaded here. Before entering the first cell, let's change the name of our empty notebook. Click on

Figure 3 ■ Example notebook



```
# Eysenck Personality Analysis

We used the Eysenck Personality Questionnaire (EPQ) to assess
the personality traits of 231 subjects. Eysenck conceptualized
personality using four dimensions

- Extraversion/Introversion
- Neuroticism/Stability
- Psychoticism/Socialisation
- Lie/Social Desirability
```

```
## Descriptive Overview
Our data is available [online](http://personality-
project.org/r/datasets/maps.mixx.epi.bfi.data). Let's have a
look at the data first.
```

```
In [ ]: file <- 'maps.mixx.epi.bfi.data'
        data <- read.table(file,header=TRUE)
        epqs  <- data[c('epiE','epiS','epilie','epiNeur')]
        boxplot(epqs)
```

Eysenck Personality Analysis

We used the Eysenck Personality Questionnaire (EPQ) to assess the personality traits of 231 subjects. Eysenck conceptualized personality using four dimensions

- Extraversion/Introversion
- Neuroticism/Stability
- Psychoticism/Socialisation
- Lie/Social Desirability

Descriptive Overview

Our data is available online. Let's have a look at the data first.

```
In [7]: file <- 'maps.mixx.epi.bfi.data'
        data <- read.table(file,header=TRUE)
        epqs  <- data[c('epiE','epiS','epilie','epiNeur')]
        boxplot(epqs)
```

the title at the top of the page and change it to something like *Working with Personality Data*.

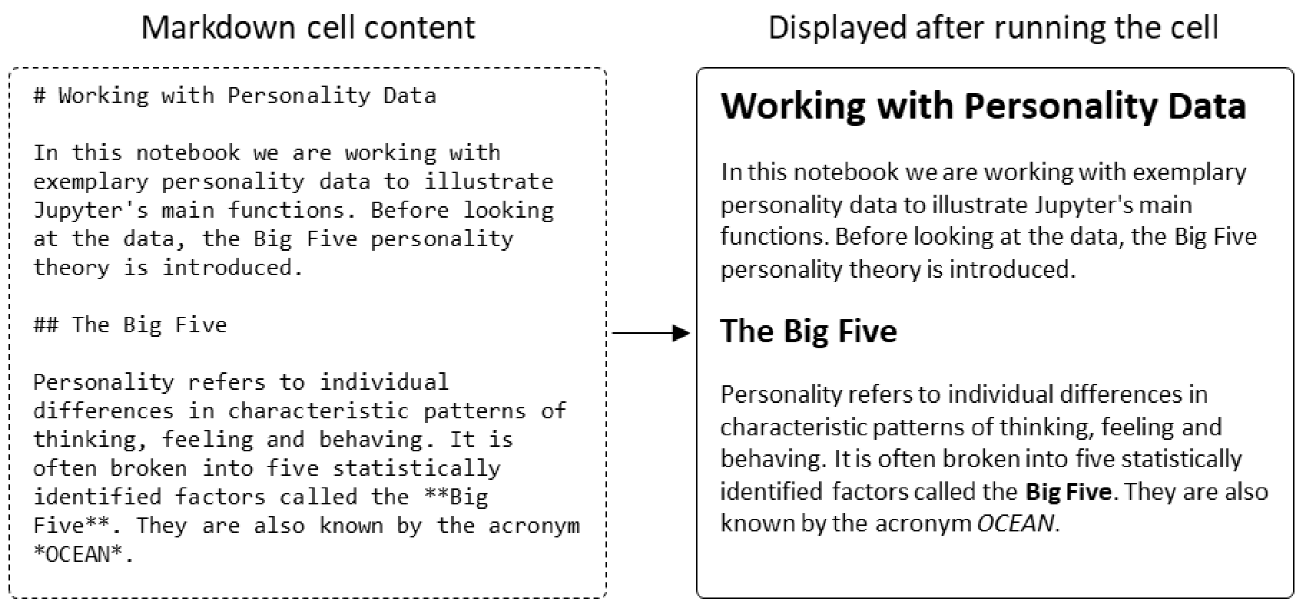
Markdown Cells

Markdown cells are used for explanatory static content like text, images and mathematical expressions. The content is styled and formatted by using the popular Markdown syntax. It is also possible to use HTML commands. Furthermore, mathematical expressions can be added to Markdown cells using LaTeX expressions. When Markdown cells are interpreted, their content is formatted by Jupyter and presented in a well readable way. In summary, Markdown cells can be used to achieve a presentation of static content comparable to cur-rent psychological journal publications. Let's have a closer look at some examples.

Headings, Bolds and Italics. Headings can be used to structure texts. In Markdown, a heading has to be in its own line and preceded by hashtags (#). The amount of hashtags defines the outline level of a heading. Text can be decorated using bold or italic words. Letters, words or groups of words surrounded by single star signs (*) are printed in italics whereas using two star signs (**) causes bold printing. Try to add the content shown on the left side of Figure 4 to your first Markdown cell. After running the cell, you should see a well formatted output containing headings, bold and italic words as shown on the right side of the figure.

Links and Images Links to websites or external data can be added to Markdown cells too. Simply surround the link's name in square brackets, followed by the target address in round brackets. You can also include images to your notebook. To do that, use an exclamation mark (!), followed by an image title in square brackets and the image's address in round brackets. If you want to show an image that is stored in the same location as your notebook, you do not need to provide its full address. Instead, you

Figure 4 ■ Formatting headings, bold and italic content



```
# Working with Personality Data

In this notebook we are working with
exemplary personality data to illustrate
Jupyter's main functions. Before looking
at the data, the Big Five personality
theory is introduced.

## The Big Five

Personality refers to individual
differences in characteristic patterns of
thinking, feeling and behaving. It is
often broken into five statistically
identified factors called the **Big
Five**. They are also known by the acronym
*OCEAN*.
```

Working with Personality Data

In this notebook we are working with exemplary personality data to illustrate Jupyter's main functions. Before looking at the data, the Big Five personality theory is introduced.

The Big Five

Personality refers to individual differences in characteristic patterns of thinking, feeling and behaving. It is often broken into five statistically identified factors called the **Big Five**. They are also known by the acronym *OCEAN*.

can just use its filename. Add another Markdown cell to your notebook containing the text from Figure 5. When running this cell, you should see both a link and an image of the big five retrieved from Wikimedia Commons.

Lists and Tables Markdown supports both numbered and unnumbered lists. Starting a new line of text with a number and a dot (1.) defines an item of a numbered list. Using a hyphen (-) instead defines an item of an unnumbered list. Tables can be rendered too, using a more complex syntax. Figure 6 shows an example. When you copy the text from the left side into a new Markdown cell, a table containing exemplary traits will be printed after running the cell. Furthermore, an unnumbered list of exemplary items will be rendered.

Mathematical Expressions Furthermore, Markdown can print mathematical expressions defined using LaTeX conventions. Simply surround the LaTeX formatted expression with single dollar signs ($) to print it in line with encompassing text or double dollar signs ($$) to render it in a separate paragraph. Try to run the example presented in Figure 7.

Code Cells

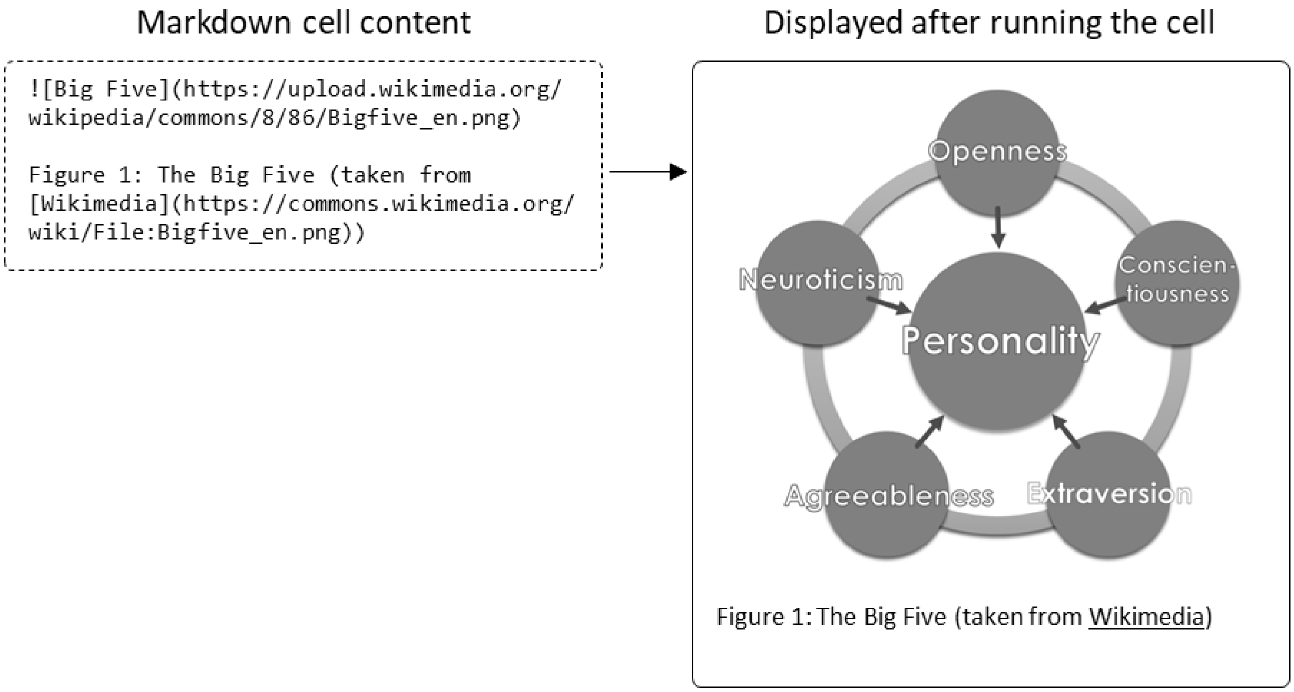
Code cells contain scripts written in a programming language like Python or (in our case) R. When interpreted by Jupyter, their output is presented below the respective cell. Depending on the languages and libraries used, outputs typically include tables, graphs (e.g. function plots, maps and rendered images) or even interactive elements like buttons and sliders. The latter can be used to alter variables within the code and visualize their effects on the output. In psychological research this can be used to investigate on specific parameters of data preparation and analysis. Typical applications include the exploration of cut-off values and outlier limits, the visualization of different statistical methods and their effects as well as the presentation of results. In comparison to Markdown cells, code cells are marked by the preceding keyword In. Note that all code used within the same notebook has to be of the same language.

Let's continue working on our personality data notebook. We already used Markdown cells to introduce the basic concept of the Big Five. Now we want to add code cells for loading real data and working with it. R provides several options for accessing both local and remote files. First, we can access data stored on our computer. For example, if a file data.csv containing personality data is stored in the exact same folder as our notebook, we can easily load its content into a new variable called person.data using the following two lines of code:

```
filename <- 'data.csv'
person.data <- read.table(filename,
    header=TRUE)
```

Second, we can access any file being publicly available on the web. For example, we can use the following lines to

Figure 5 ■ Formatting links and images



```
![Big Five](https://upload.wikimedia.org/
wikipedia/commons/8/86/Bigfive_en.png)

Figure 1: The Big Five (taken from
[Wikimedia](https://commons.wikimedia.org/
wiki/File:Bigfive_en.png))
```

Figure 1: The Big Five (taken from Wikimedia)

retrieve data from Northwestern University's Personality Project and store it in a new variable called person.data. Create a new code cell in your notebook containing these lines. Take care when copying the web address, it must not contain any spaces or line breaks.

```
filename <- 'http://
    personality-project.org/r/datasets/
    maps.mixx.epi.bfi.data'
person.data <- read.table(datafilename,
    header=TRUE)
```

To check if loading the data works as expected, also add the following line at the end of your code cell:

```
person.data
```

Run the cell (by clicking the Run button as shown in Figure 3, or by pressing Ctrl-Enter). The interpreter will load the data, create two new variables `filename` and `person.data` and finally print parts of the content of person.data in a table below the code cell (see Figure 8).

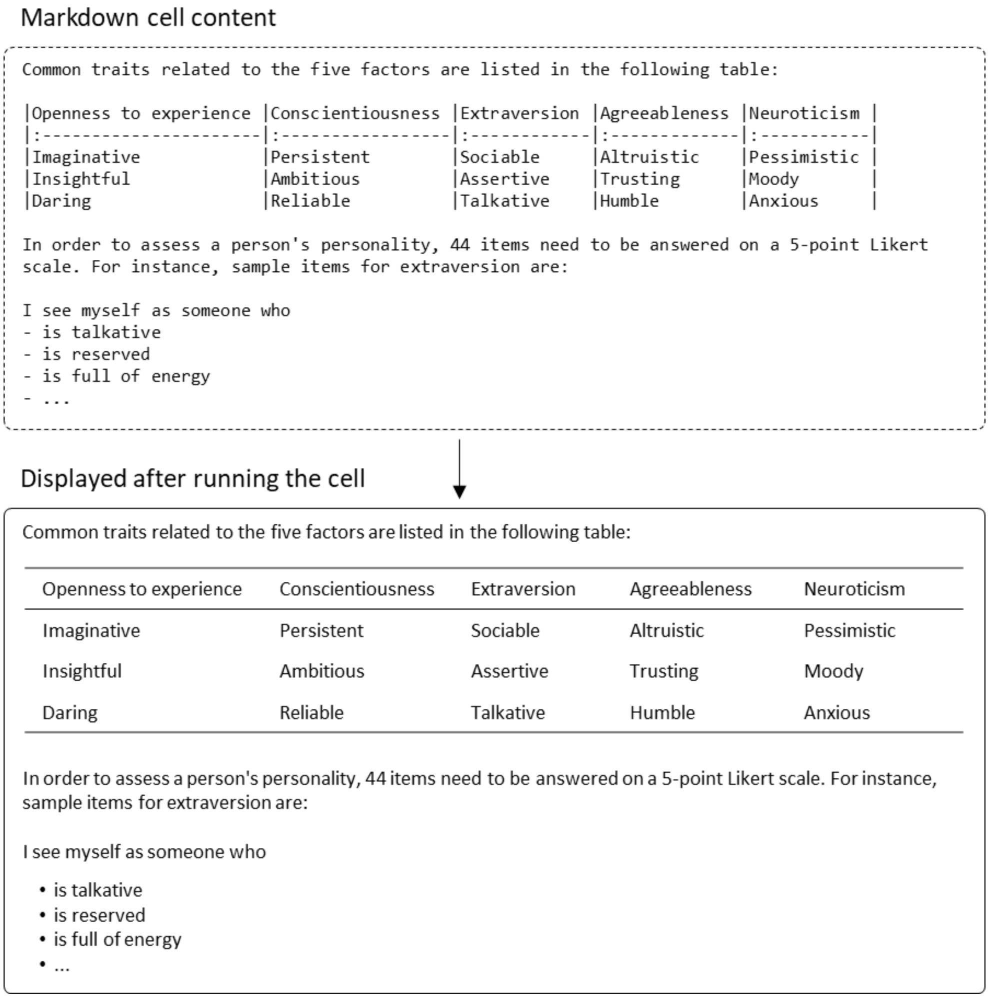
As we can see, the loaded file contains different personality scales for a lot of subjects. Since we are interested in the Big Five, we should only use a subset of columns (those starting with `bf`) for a subsequent analysis. Let's create a new variable containing these columns by adding another code cell at the end of our notebook. Enter the following:

```
bigfive <- person.data[c('bfagree','
    bfcon','bfext','bfneur','bfopen')]
boxplot(bigfive,las=2)
```

After running the cell, a boxplot will pop up below the code, showing descriptive details of the five factors (see Figure 9). Just like when working with plain R, data can be analyzed and manipulated in endless ways. You may have noticed that variables created after running a code cell, are usable within the other code cells too. That's why we could use the variable `person.data` to extract the Big Five.

Since Jupyter has no variable viewer, we need to use another code cell running the `ls()` command for getting an overview about the variables currently in use. By doing so, you will see that three variables have been created while we have been working with our notebook (`filename`, `person.data` and `bigfive`). If you want to delete all variables, you have to restart the R kernel from the menu by clicking `Kernel → Restart`. This is especially useful to clean up the notebook's memory after experimenting with a lot of variables. Always remember to restart the kernel if you want to rerun the whole notebook from a defined starting point.

Figure 6 ■ Formatting lists and tables



```
Common traits related to the five factors are listed in the following table:

|Openness to experience |Conscientiousness |Extraversion |Agreeableness |Neuroticism |
|:----------------------|:-----------------|:------------|:-------------|:-----------|
|Imaginative            |Persistent        |Sociable     |Altruistic    |Pessimistic |
|Insightful             |Ambitious         |Assertive    |Trusting      |Moody       |
|Daring                 |Reliable          |Talkative    |Humble        |Anxious     |

In order to assess a person's personality, 44 items need to be answered on a 5-point Likert
scale. For instance, sample items for extraversion are:

I see myself as someone who
- is talkative
- is reserved
- is full of energy
- ...
```

Common traits related to the five factors are listed in the following table:

Openness to experience	Conscientiousness	Extraversion	Agreeableness	Neuroticism
Imaginative	Persistent	Sociable	Altruistic	Pessimistic
Insightful	Ambitious	Assertive	Trusting	Moody
Daring	Reliable	Talkative	Humble	Anxious

In order to assess a person's personality, 44 items need to be answered on a 5-point Likert scale. For instance, sample items for extraversion are:

I see myself as someone who

- is talkative
- is reserved
- is full of energy
- ...

Advanced Features

Jupyter provides useful tools we cannot cover in detail here. Cells can be splitted and merged, deleted, moved and converted from one type to the other. They can be interpreted one by one or all at once. Notebooks can be exported into common formats including LaTeX, PDF and HTML. Depending on the target format, this may require a working internet connection since conversion services from the web are used. All features are accessible over the extensive menu at the top of the notebook.

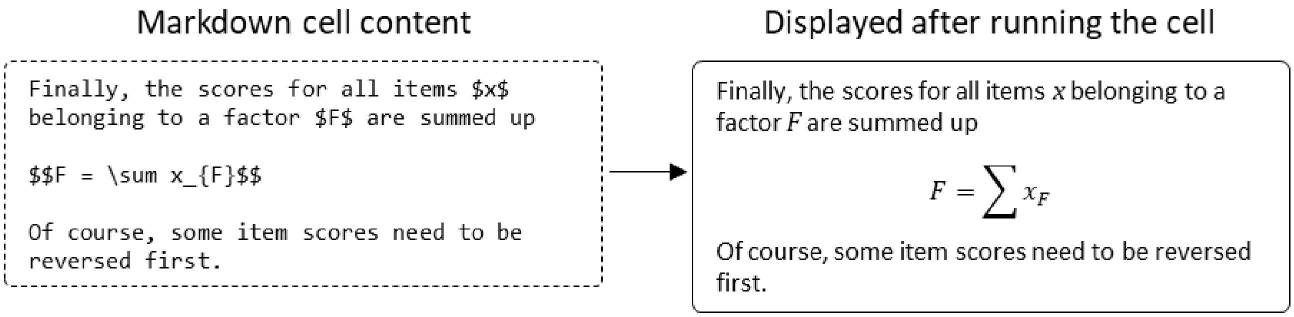
As of today, lots of plugins are available to extend the functionality of Jupyter. This includes additions for the management of references as well as plugins enabling others to comment on notebook content. Please consult the official Jupyter documentation about installing and using these plugins, available from http://jupyter-notebook.readthedocs.io/.

Versioning and Sharing Notebooks

Jupyter makes it easy to keep track of our changes made to a notebook. It automatically saves an opened notebook from time to time and we can force it to do so by clicking `File → Save and Checkpoint`. Jupyter allows us to restore a saved checkpoint by choosing `File → Revert to Checkpoint`. That means we can easily roll back to an older version after experi-menting around a lot.

In many cases, you want to provide a notebook to other people. There are several options to share it. First, you can send the notebook file via email. The receiving person can simply load the notebook in his or her own Jupyter installa-

Figure 7 ■ Formatting mathematical expressions



```
Finally, the scores for all items $x$
belonging to a factor $F$ are summed up

$$F = \sum x_{F}$$

Of course, some item scores need to be
reversed first.
```

Finally, the scores for all items x belonging to a factor F are summed up

$$F = \sum x_{F}$$

Of course, some item scores need to be reversed first.

tion by choosing `File → Open` from the menu. Second, you can host your notebook online and provide its link to others, so they can continue where you finished working. You can either bring your own Jupyter installation online (this requires setting up a server machine and cannot be detailed here) or use an installation set up by one of several specialized cloud hosting providers (e.g. Microsoft Azure Notebooks). However, it is not possible to bring your notebooks online by using file hosting platforms like Dropbox or Google Drive. Since Jupyter is still in it's infancy, alternatives for sharing notebooks are expected to increase.

Conclusion

Jupyter is designed to solve some of the main problems in psychological research. First, it helps scientists keeping track of their work. Since both static and dynamic assets of a research project can be included in Jupyter notebooks, they help organizing ideas, information acquired from lab or field environments, statistical methods and scripts as well as results and interpretations. Everything is kept in one place. Changes are documented and can be reverted at any time. Second, Jupyter promotes sharing work. It enables others to explore and understand the research undertaken. Thus, Jupyter may help increasing the reproducibility of results and fostering good academic practice. However, there are other notebook systems as well. This includes Apache Zepplin and the R Notebooks feature of RStudio. Apache Zeppelin is a web-based notebook system like Jupyter. It was designed for data analysis using Python and Spark, but can be used with R too. Apache Zeppelin even supports combining several languages within the same notebook, a feature missing in Jupyter. Unfortunately, setting up Apache Zeppelin on a Windows machine requires a lot of effort. The system is pretty new and not as well established as Jupyter. Thus, Jupyter may be the better choice. However, if you are familiar with RStudio and use R for all your data analysis, you won't need Jupyter at all. RStudio supports writing R Notebooks containing both markup and R code. The notebooks can be shared more easily compared to Jupyter because they are stored as plain text. Unfortunately, R Notebooks do not support coding in other languages. In conclusion, for Windows users with the need for multiple languages, Jupyter notebooks may be the best choice. In any case, they are a great addition to the often short and abstract journal publications.

References

Donoho, D. L., Maleki, A., Rahman, I. U., Shahram, M., & Stodden, V. (2009). Reproducible research in computational harmonic analysis. *Computing in Science & Engineering*, *11*, 8–18. doi:10.1109/mcse.2009.15

Klein, R. A., Ratliff, K. A., Vianello, M., Adams, R. B., Bahnìk, S., Bernstein, M. J., & Nosek, B. A. (2014). Investigating variation in replicability. *Social Psychology*, *45*(3), 142–152. doi:10.1027/1864-9335/a000178

Nosek, B. A., Spies, J. R., & Motyl, M. (2012). Scientific utopia. *Perspectives on Psychological Science*, *7*(6), 615–631. doi:10.1177/1745691612459058

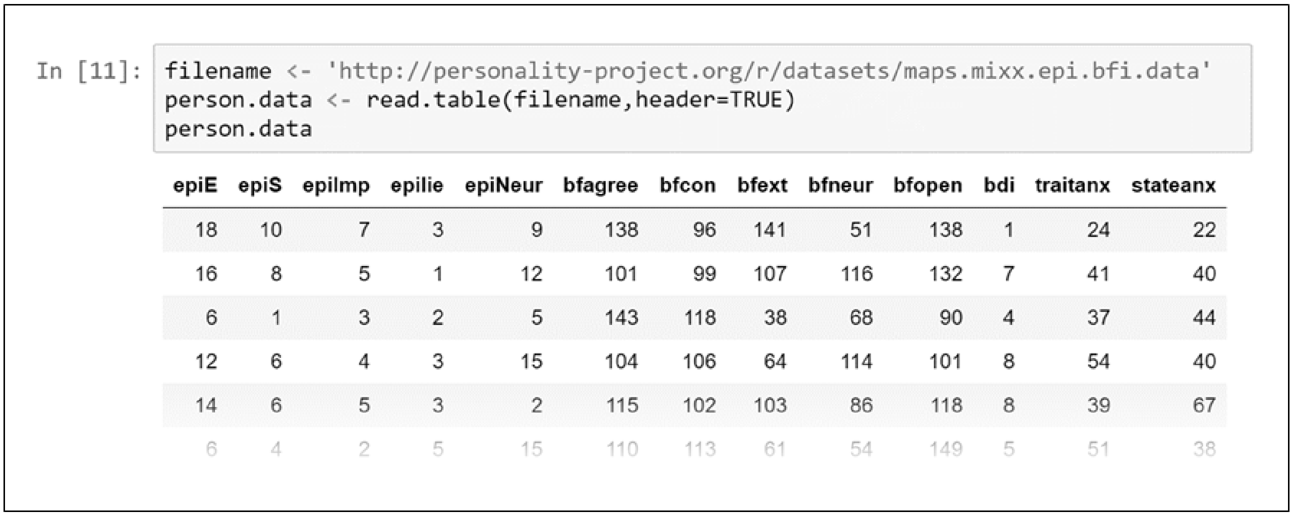
Pashler, H. & Wagenmakers, E. J. (2012). Editors' introduction to the special section on replicability in psychological science. *Perspectives on Psychological Science*, *7*(6), 528–530. doi:10.1177/1745691612465253

Perez, F. & Granger, B. E. (2007). Ipython: a system for interactive scientific computing. *Computing in Science & Engineering*, *9*(3), 21–29. doi:10.1109/mcse.2007.53

Shen, H. (2014). Interactive notebooks: sharing the code. *Nature*, *515*(7525), 151–152. doi:10.1038/515151a

Yong, E. (2012). Replication studies: bad copy. *Nature*, *485*(7398), 298–300. doi:10.1038/485298a

Figure 8 ■ Printing data loaded from the remote Personality Project



```
In [11]: filename <- 'http://personality-project.org/r/datasets/maps.mixx.epi.bfi.data'
         person.data <- read.table(filename,header=TRUE)
         person.data
```

epiE	epiS	epiImp	epilie	epiNeur	bfagree	bfcon	bfext	bfneur	bfopen	bdi	traitanx	stateanx
18	10	7	3	9	138	96	141	51	138	1	24	22
16	8	5	1	12	101	99	107	116	132	7	41	40
6	1	3	2	5	143	118	38	68	90	4	37	44
12	6	4	3	15	104	106	64	114	101	8	54	40
14	6	5	3	2	115	102	103	86	118	8	39	67
6	4	2	5	15	110	113	61	54	149	5	51	38

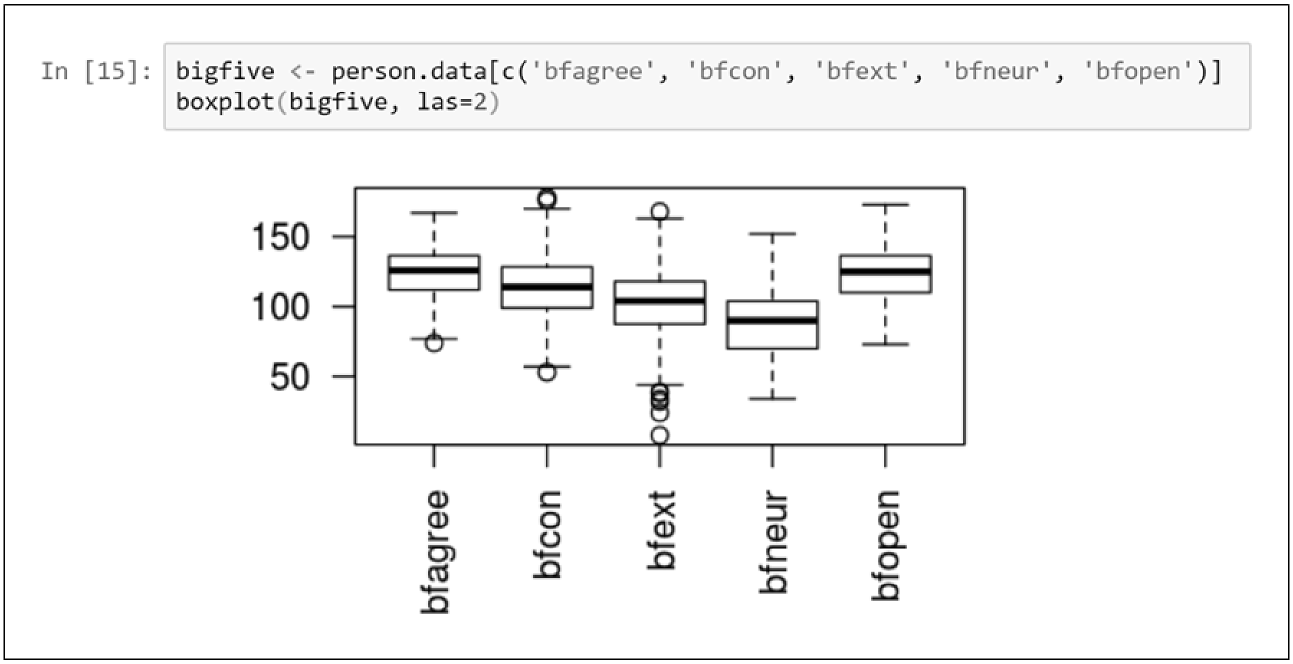
Citation

Sprengholz, P. (2018). Welcome to jupyter: improving collaboration and reproduction in psychological research by using a notebook system. *The Quantitative Methods for Psychology*, *14*(2), 137–146. doi:10.20982/tqmp.14.2.p137

Received: 25/11/2017 ∼ Accepted: 22/02/2018

Figure 9 follows.

Figure 9 ■ Descriptive analysis using a boxplot



```
In [15]: bigfive <- person.data[c('bfagree', 'bfcon', 'bfext', 'bfneur', 'bfopen')]
         boxplot(bigfive, las=2)
```