

Introduction to Python with Power BI



Agenda

- Understanding Python as a Data Platform
- Installing Python Developer Tools
- Getting Started Writing Code in Python
- Integrating Python with Power BI Desktop



What is Python?

- What is Python?
 - Platform for data extraction and data preparation
 - Platform for statistics, data analysis and visualization
 - Free, cross-platform, open source software
 - Programming language + Runtime layer + Libraries
 - Flourishing ecosystem of Python package authors
- Why do you need it?
 - Analyzing data and generating statistics
 - Creating rich graphs and charts
 - Fitting statistical models for predictive analysis



Agenda

- ✓ Understanding R as an Analytics Platform
- Installing Microsoft R Open and RStudio
 - Writing R Code in RStudio
 - Integrating R with Power BI Desktop



Install Microsoft R Open

- <https://mran.microsoft.com/download/>

Download Microsoft R Open 3.3.1, the enhanced R distribution

Microsoft R Open, **the enhanced distribution of R** from Microsoft, is a complete and free open source platform for statistical analysis and data science. Microsoft R Open 3.3.1 is based on (and 100% compatible with) the statistical language, R-3.3.1. It includes additional capabilities for performance, reproducibility and platform support. [Learn more...](#)

[Prerequisites & Install Docs](#) | [Forum](#) | [News](#) | [Past Releases](#)

Microsoft R Open & MKL Downloads

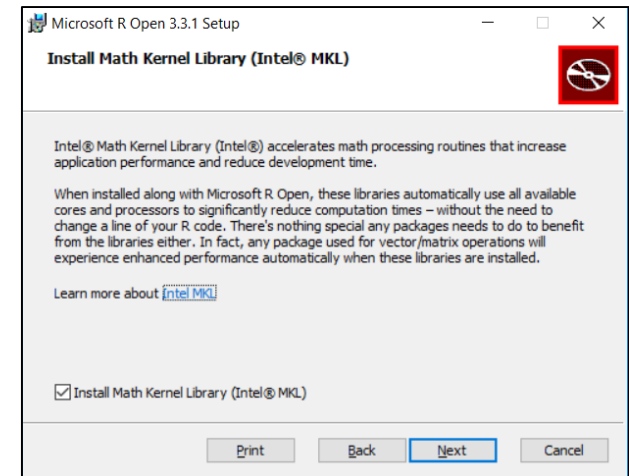
While the install of MKL, used for multithreaded performance, is **optional**, we recommend both Microsoft R Open & MKL for optimal performance on Windows and Linux. The OS X operating system has Math Libraries by default.

Platforms (64-Bit only)	Downloads
Windows - Windows 7.0 (SP1), 8.1, 10 and Windows Server® 2008 R2 (SP1), 2012 SHA 256: 0a99d2c9aa1465d25d9cb8cd0ff07e73a13c6746a45ffd03b79c85258599747	R Open / MKL
Ubuntu - 14.04, 15.04 SHA 256: b2568eb06f29964765136a4eb096659378d629a4cca9963b016bf731004eb71d	R Open / MKL
Red Hat Enterprise Linux - 6.5, 7.1 SHA 256: b2568eb06f29964765136a4eb096659378d629a4cca9963b016bf731004eb71d	R Open / MKL




Microsoft R Server Users:

- Get [R Open for R Server 2016](#)
- Get [R.R.O 8.0.3](#) for R.R.E 7.4.1



Installing R Studio

- <https://www.rstudio.com/products/rstudio/download/>

ProductsResourcesPricingAbout UsBlog

Download RStudioHome / Overview / RStudio / Download RStudio

RStudio is a set of integrated tools designed to help you be more productive with R. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management.

If you run R on a Linux server and want to enable users to remotely access RStudio using a web browser [please download RStudio Server](#).

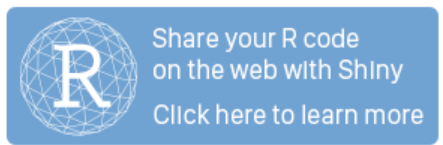

Do you need support or a commercial license? [Check out our commercial offerings](#)

RStudio Desktop 0.99.902 — [Release Notes](#)

RStudio requires R 2.11.1 (or higher). If you don't already have R, you can download it [here](#).

Installers for Supported Platforms

Installers	Size	Date	MD5
RStudio 0.99.902 - Windows Vista/7/8/10	77.1 MB	2016-05-14	8feae61d13b1d81ded7587a1da760d95
RStudio 0.99.902 - Mac OS X 10.6+ (64-bit)	60 MB	2016-05-14	f741e4a1345985c16e692967adb210
RStudio 0.99.902 - Ubuntu 12.04+/Debian 8+ (32-bit)	81.6 MB	2016-05-14	363952616a10553aa51f3a9129b9adeb
RStudio 0.99.902 - Ubuntu 12.04+/Debian 8+ (64-bit)	88.3 MB	2016-05-14	d035622f39928246048972ed2064c89a
RStudio 0.99.902 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	81 MB	2016-05-14	6f14d4717b01e7763d18f1cdad8e6474
RStudio 0.99.902 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	81.9 MB	2016-05-14	af9e8cd892a497a92aafce8629f90e90





The RStudio IDE

~/RProject1 - RStudio

File Edit Code View Plots Session Build Debug Tools Help

Go to file/function Addins

Demo1.R* x CreatingDatasets.R x Ch03 Getting started with graphs.R x

Source on Save Run Source

```
1 message <- "Hello World"
2 message
3
4 # create a dataset
5 x <- pretty(c(-3, 3), 100)
6 y <- dnorm(x)
7
8 # plot the dataset
9 plot(x, y,
10      xlab="Normal Deviation",
11      ylab = "Density",
12      yaxs="i")
13
```

13:1 (Top Level) R Script

Environment History

Global Environment

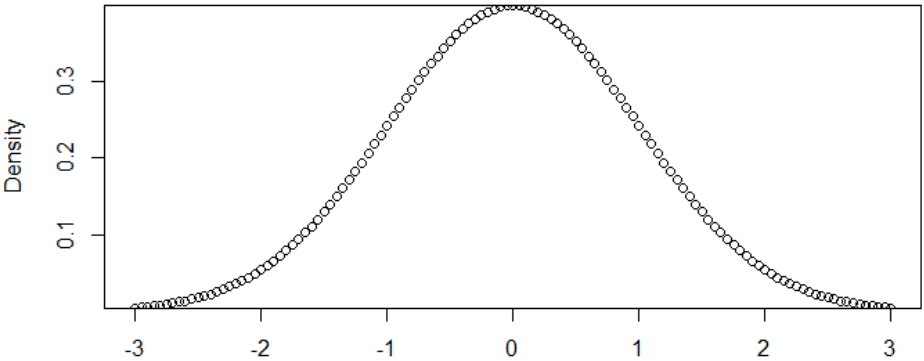
values

message	"Hello World"
x	num [1:121] -3 -2.95 -2.9 -2.85 -2.8 -2.75 ...
y	num [1:121] 0.00443 0.00514 0.00595 0.00687...

Files Plots Packages Help Viewer

Zoom Export Publish

Density



Normal Deviation

Console ~/RProject1/

```
> message <- "Hello world"
> message
[1] "Hello world"
> # create a dataset
> x <- pretty(c(-3, 3), 100)
> y <- dnorm(x)
> # plot the dataset
> plot(x, y,
+ xlab="Normal Deviation",
+ ylab = "Density",
+ yaxs="i")
Hit <Return> to see next plot:
>
>
```

Agenda

- ✓ Understanding R as an Analytics Platform
- ✓ Installing Microsoft R Open and RStudio
- Writing R Code in RStudio
- Integrating R with Power BI Desktop



R Projects and Workspaces

- R projects based on folder structure
 - Data and scripts added to current working directory
- Each R project defines a workspace
 - Workspace tracks set of user-defined objects
 - Workspace defines set of loaded packages
 - Workspace data saved/loaded using .RData files

```
Console ~/RProject1/ ↗  
> getwd()  
[1] "C:/Users/Student/Documents/RProject1"  
> .libPaths()  
[1] "C:/Users/Student/Documents/R/win-library/3.2"  
[2] "C:/Program Files/Microsoft/MRO/R-3.2.4/library"  
> |
```



Writing and Testing R Code in Scripts

```
01_GettingStarted.R ×
Source on Save
1 # use <- for variable assignment
2 message <- "Hello world"
3
4 print(message)
5
6 # create vector using the c function
7 vector1 <- c(2, 4, 6, 8)
8
9 # create vectors using sequence
10 vector2 <- 1:10
11 vector3 = letters[1:5]
12 vector4 = LETTERS[24:26]
13 vector6 = 2^(1:8)
14
15 # create vector with election years
16 election.years <- seq(from = 1996, to = 2016, by = 4)
17
18 # enumerate through election years using for loop
19 for (year in election.years){
20   print(paste(year, "is an election year"))
21 }
22
23 # remove all objects from workspace
24 rm(list=objects())
```



R Objects

- In R, variables represent named objects
- Object names can contain
 - Letters
 - Numbers
 - Underscores (`_`)
 - Dots (`.`)



Essential Data Structures in R

- Vector
 - One-dimensional, single-mode array
- Matrix
 - Two-dimensional, single-mode array
- Array
 - N-dimensional, single-mode array
- List
 - Ordered collection of multi-mode objects
- Data frame
 - Two-dimensional, multi-mode array
- Factor
 - Integer-backed list of categorical values





DEMO

Writing and Testing R Code in RStudio



DEMO

Creating Graphs using RStudio

Agenda

- ✓ Understanding R as an Analytics Platform
- ✓ Installing Microsoft R Open and RStudio
- ✓ Writing R Code in RStudio
- Integrating R with Power BI Desktop



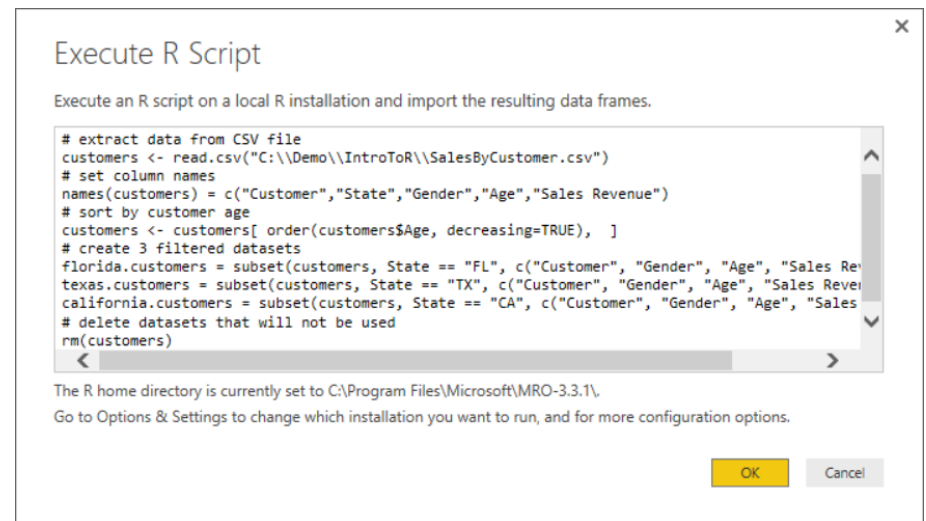
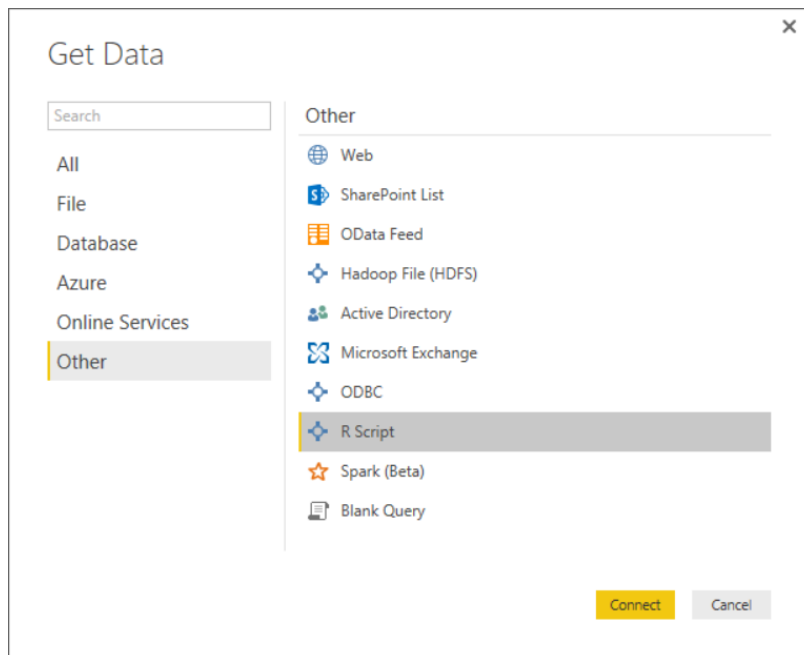
Where Can You Use R Code in PBIDT?

- As a data source to a query
 - You can use R code to import and reshape data
- Within a Query Applied Step
 - You can use R code to add transforms to a query
- Inside an R Visual in a Power BI Report
 - You can use R code to creates charts from your data



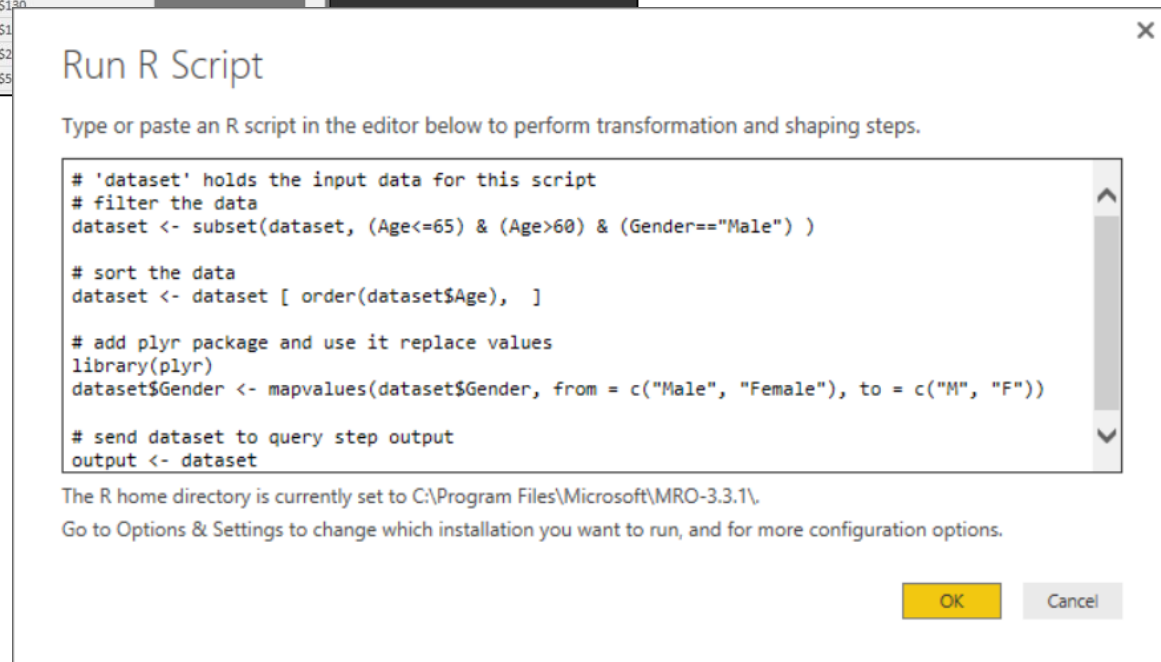
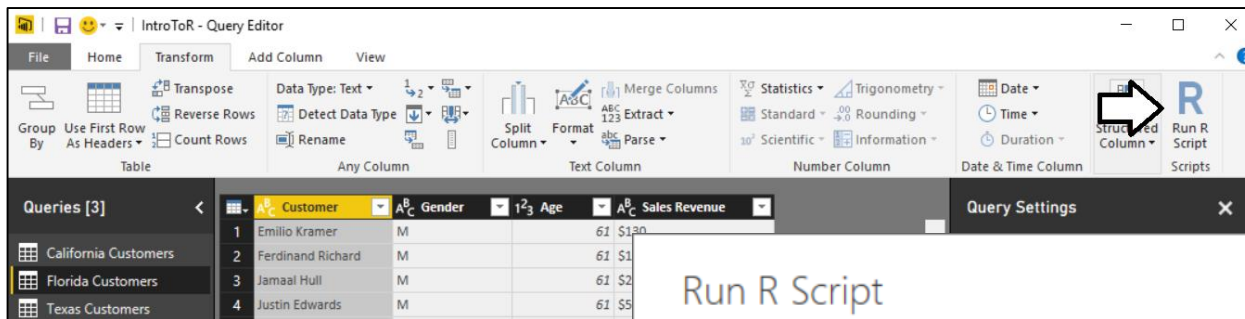
Using R Code as a Query Data Source

- Create new query based on R script
 - Copy and paste code from RStudio into PBIDT

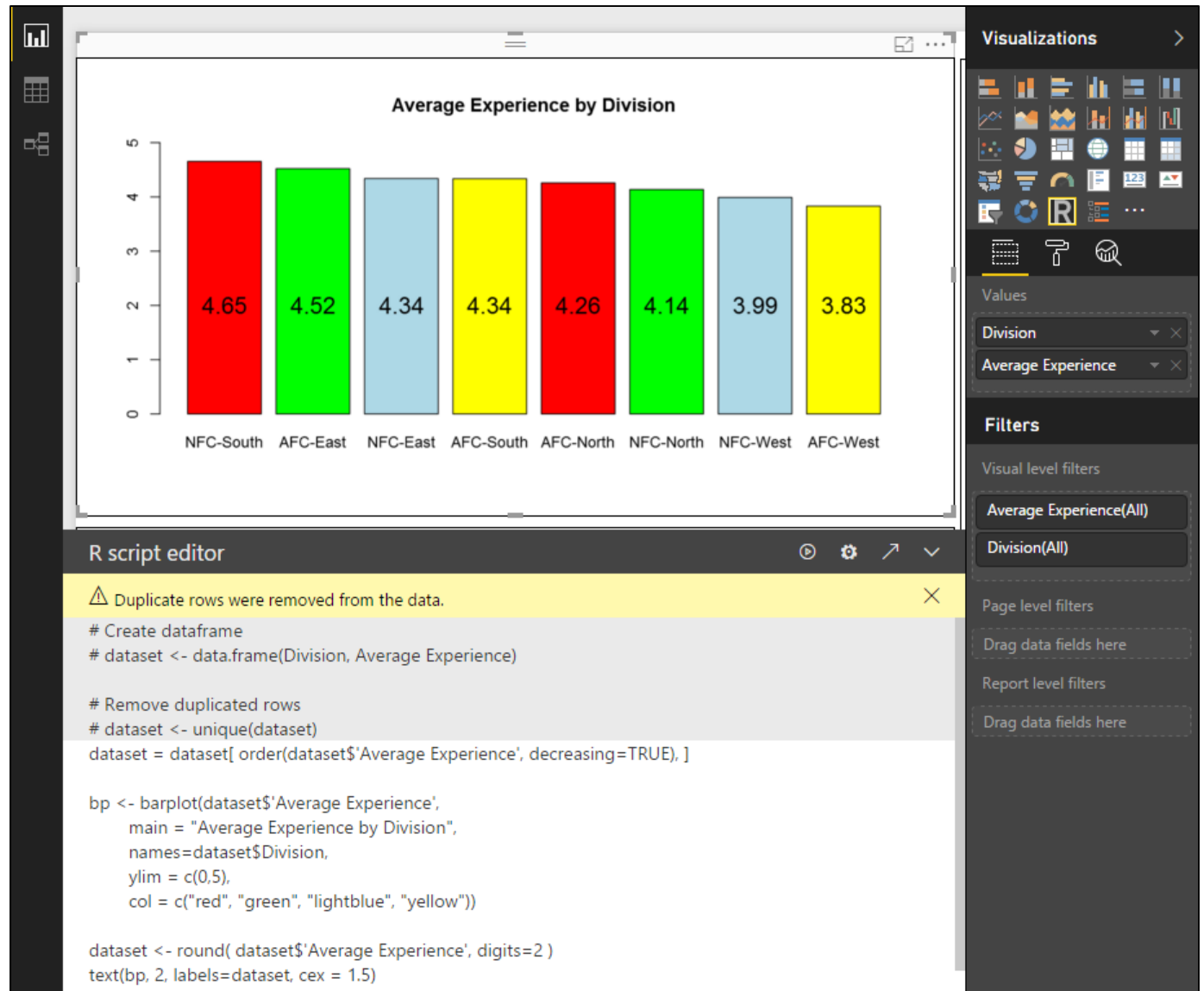


Using R Code as an Applied Query Step

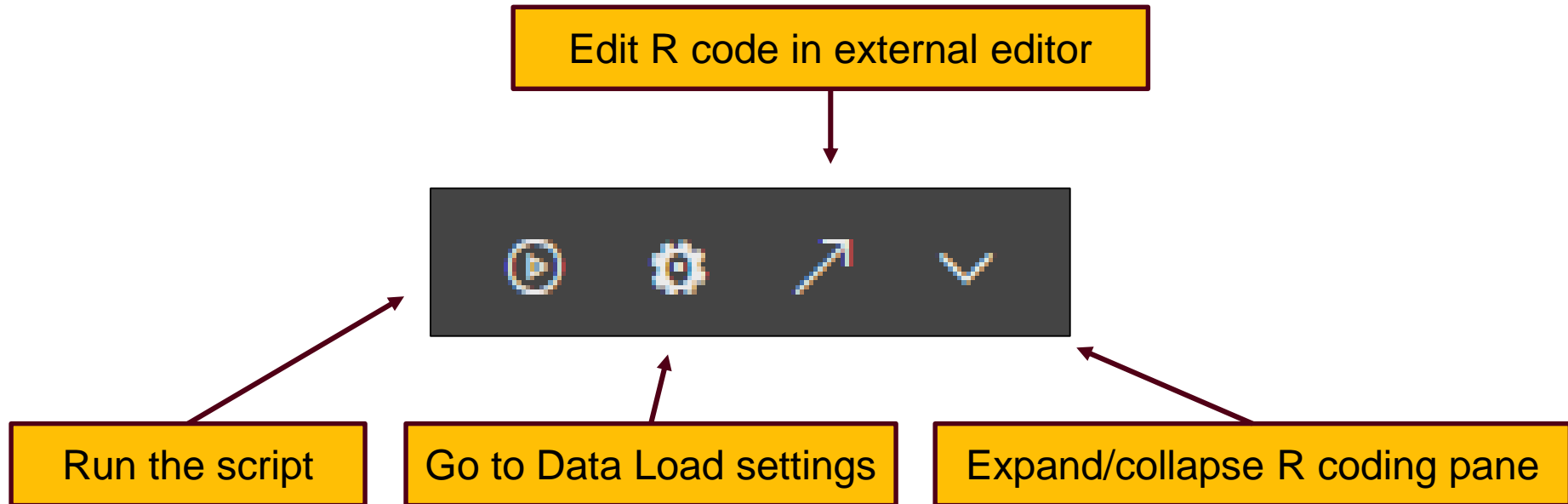
- Add new Run R Script step to query
 - Use R code and R packages to transform data



R Visuals in Power BI



New R Features in September Release



The screenshot shows the "R script editor" window. At the top, there is a title bar with the text "R script editor" and four icons: a play button, a gear, an upward-pointing arrow, and a downward-pointing arrow. Below the title bar, a yellow message box with a warning icon and a close button (X) displays the text: "Duplicate rows were removed from the data." Below the message box, the R code is displayed in a light gray background:

```
# Create dataframe
# dataset <- data.frame(Division, Average Experience)

# Remove duplicated rows
# dataset <- unique(dataset)
dataset = dataset[ order(dataset$'Average Experience', decreasing=TRUE), ]
```


R Integration Limitations with Power BI

- Power BI Desktop R Limitations
 - Only data frames are imported
 - Complex columns and Vector columns are not imported
 - Values that are N/A are translated to NULL values
 - Prompting for user input halts script
 - R visual data for plotting is limited to 150,000 rows
 - R visual calculation times out with error after 5 minutes
 - R visual is not interactive – no highlighting support
 - Plots can only be displayed to R default display device



Summary

- ✓ Understanding R as an Analytics Platform
- ✓ Installing Microsoft R Open and RStudio
- ✓ Writing R Code in RStudio
- ✓ Integrating R with Power BI Desktop

