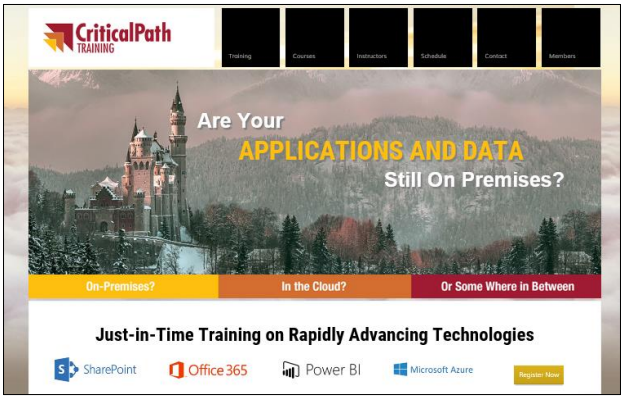
Introduction to R with Power BI

Critical Path Training

- Upcoming Power BI Bootcamp Classes (4 days)
  - Nov 30th – attend in Tampa, FL or remote
  - Jan 23rd – attend in Tampa, FL or remote
- Visit us at http://CriticalPathTraining.com for more info



Slides and Sample Code from this Webinar

- Webinar materials are available in GitHub

Agenda

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

What is R?

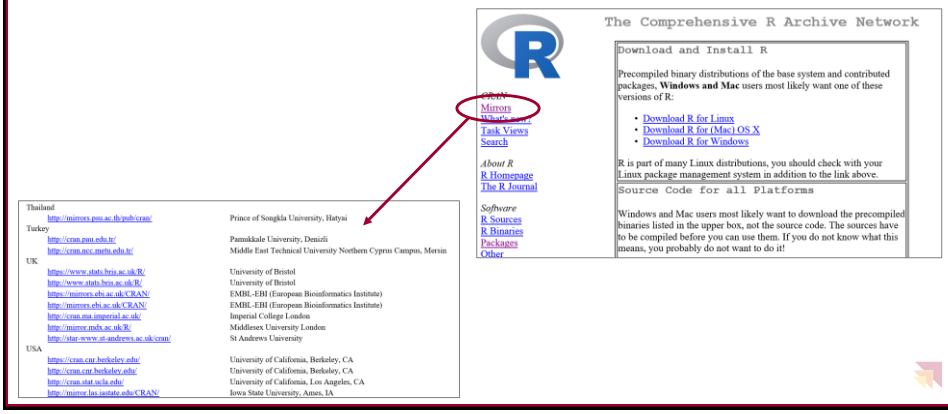
- What is R?
  - Platform for statistics, data analysis and visualization
  - Free, cross-platform, open source software
  - Programming language + Runtime layer + Libraries
  - R code distributed and versioned using packages
  - Flourishing ecosystem of R package authors
- Why do you need it?
  - Analyzing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

R Packages

- Package is versioned redistributable unit of code
  - Package contains functions, data and compiled code
  - R is installed with a default set of packages
  - Other packages can be downloaded and installed
- Examples of available domain-specific packages
  - Packages to download and unpack data in zip archive
  - Packages to create fancy charts and graphs
  - Packages to optimize financial portfolios
  - Packages predict component failure times
  - Packages to analyze genomic sequences

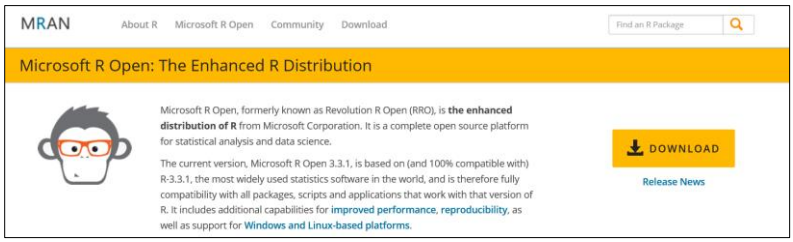
CRAN

- The Comprehensive R Archive Network
  - Public archive with over 8,000 downloadable packages
  - http://cran.us.r-project.org/



Microsoft R Open

- What is Microsoft R Open?
  - An enhanced distribution of R from Microsoft
  - Improved performance and multithreading
  - Reproducibility through package versioning stability
  - Free, cross-platform, open source software
  - Available at https://mran.microsoft.com/open/



Stages of R Awareness

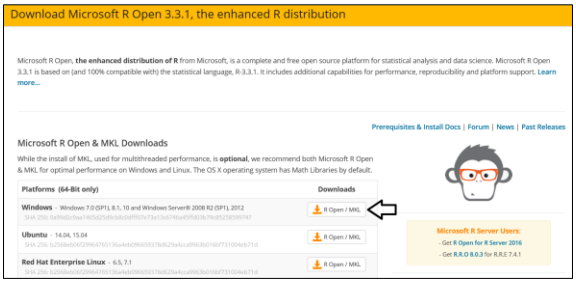
- Stage 1: Standing Up
  - Installing the environment and playing with data
- Stage 2: Walking
  - Writing & testing R code and creating graphs and charts
- Stage 3: Jogging
  - Crunching numbers to generate advanced statistics
- Stage 4: Running
  - Creating a domain-specific predictive model
- Stage 5: Sprinting
  - Distributing your predictive model as a CRAN package

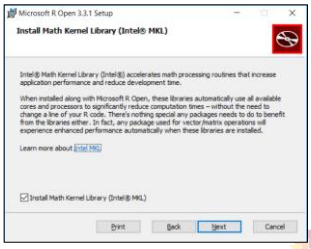
Agenda

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

Install Microsoft R Open

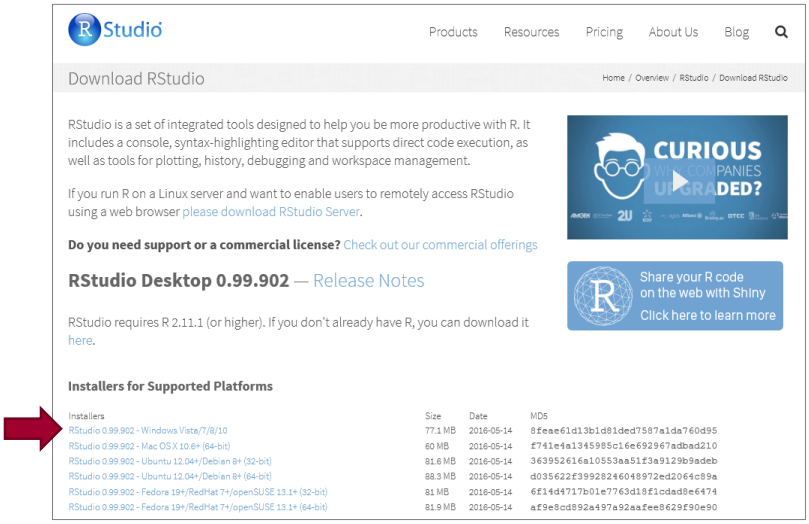
- https://mran.microsoft.com/download/



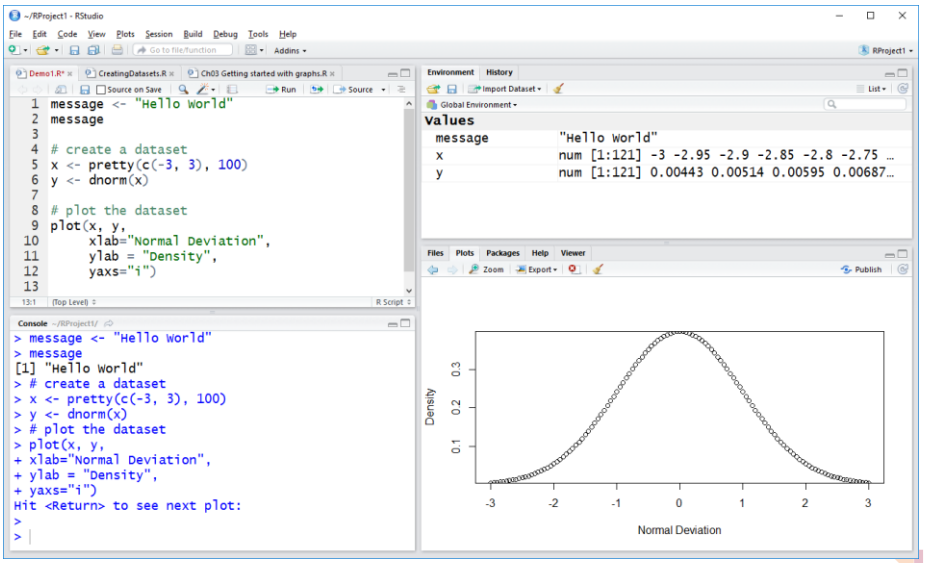


Installing R Studio

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



The RStudio IDE



Agenda

- ✓ Understanding R as an Analytics Platform
- ✓ Installing the 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

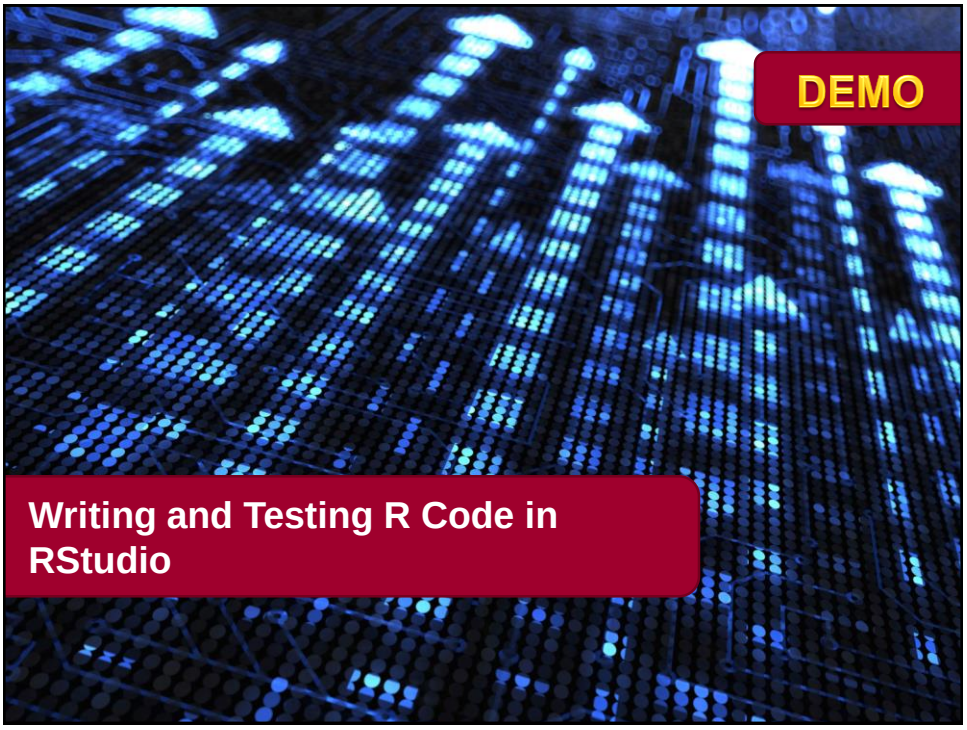
```
1  # use <- for variable assignment
2  message <- "Hello World"
3
4  print(message)
5
6  # create vector using the c function
7  vector1 <- c(2, 4, 6,  8)
8
9  # create vectors using sequence
10 vector2 <- 1:10
11 vector3 = letters[1:5]
12 vector4 = LETTERS[24:26]
13 vector6 = 2^(1:8)
14
15 # create vector with electin 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())
```

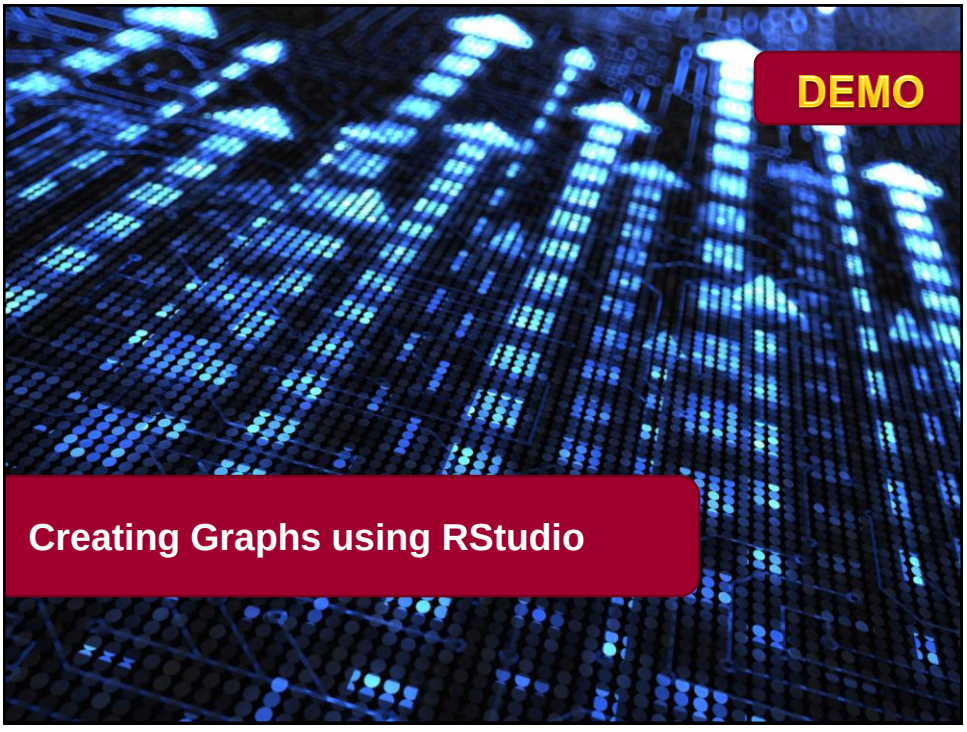
R Objects

- 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





Agenda

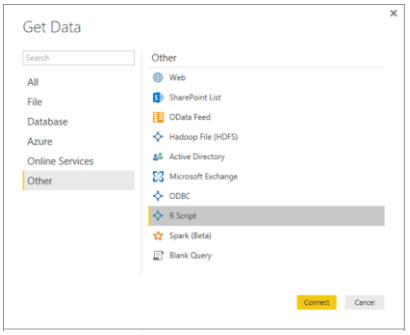
- ✓ Understanding R as an Analytics Platform
- ✓ Installing the Microsoft R Open and RStudio
- ✓ Writing R Code in RStudio
- ➢ Integrating R with Power BI Desktop

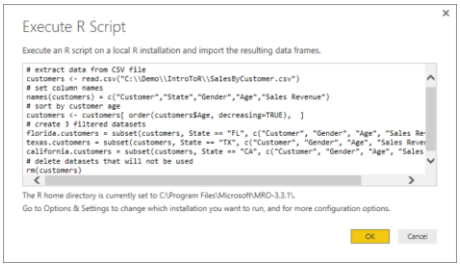
Where Can You Use R Code in PBIDT?

- 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

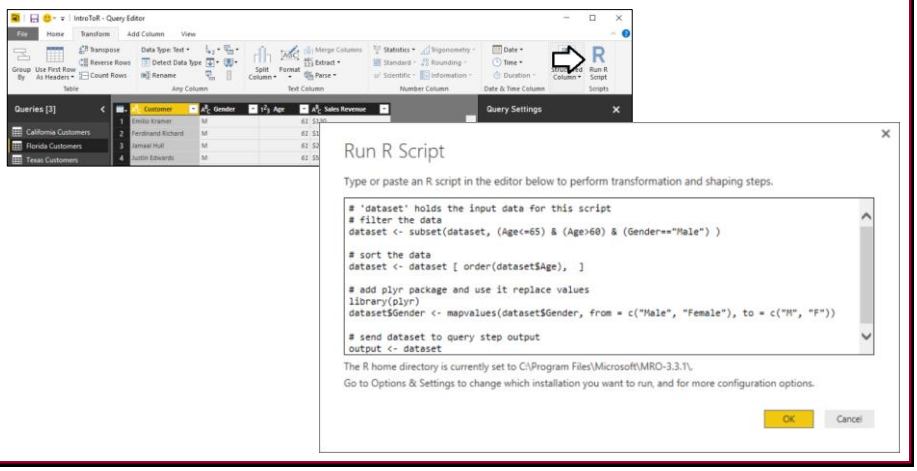
- Create new query based on R script
  - Copy and paste code from RStudio into PBIDT



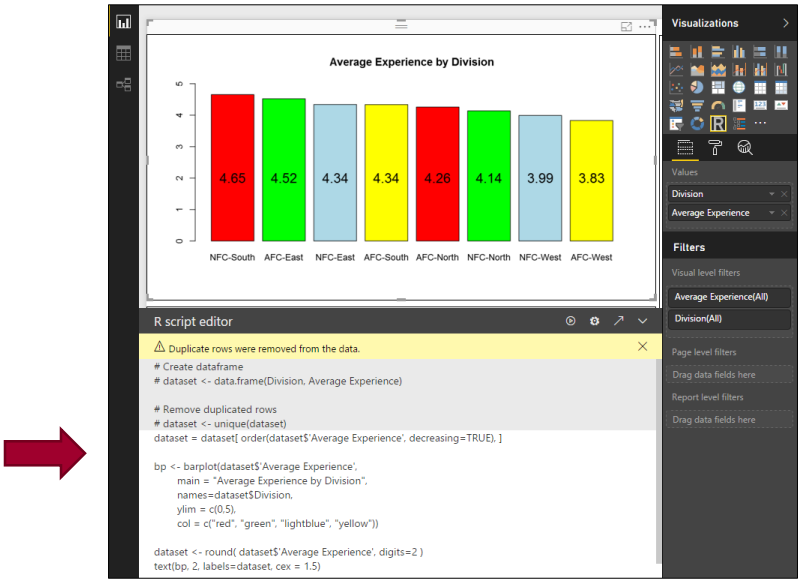


Using R Code as an Applied Query Step

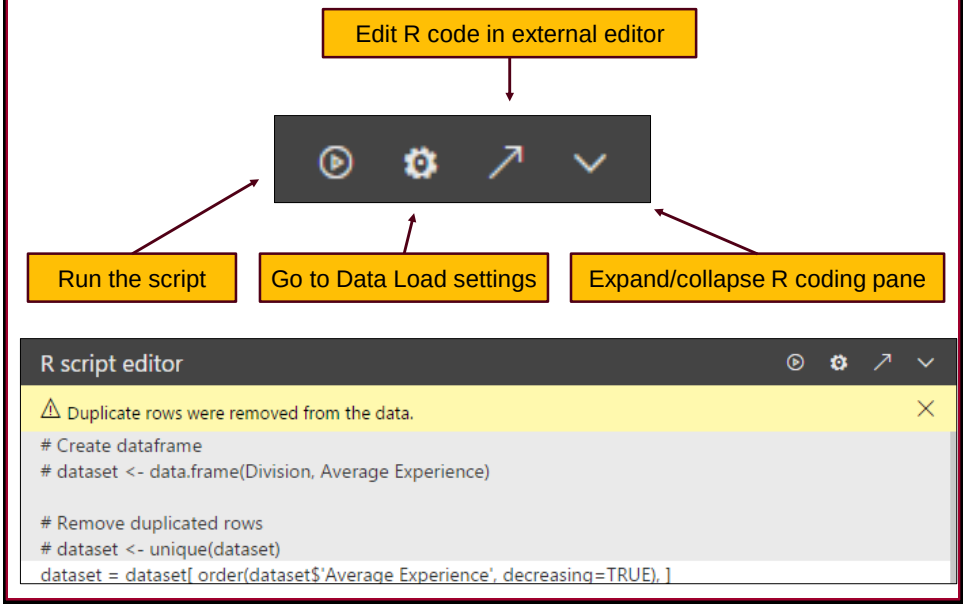
- Add new Run R Script step to query
  - Use R code and R packages to transform data



R Visuals in Power BI



New R Features in September Release



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 the Microsoft R Open and RStudio
- ✓ Writing R Code in RStudio
- ✓ Integrating R with Power BI Desktop