



# Introduction to R, R-studio and R Packages

R is a programming language and free software environment for statistical computing and graphics supported by the R Foundation for Statistical Computing. The R language is widely used among statisticians and data miners for developing statistical software and for data analysis. We use R throughout this course to demonstrate the implementation of various concepts that we cover in the course. This module provides an overview of the R language, and illustrate step by step guides for installation of R and R-studio. We will then introduce key data structures in R and overview input and output operations in R. Finally, control structures are introduced and discussed.

## What is R Programming?

- R is a free software environment for statistical computing and graphics. It's open source and therefore available free of charge.
- R is an object oriented programming language where we create objects and manipulate them as intended. Objects can be Data frames, vectors, matrices, lists, raw data, spatial objects , maps etc.

So what is R? Well, R is a software that provides a programming environment for doing statistical data analysis. This software was written by Robert Gentleman and Ross Ihaka and the name of the software bear the name of the creators. It is a free implementation of S which is another popular statistical software. R can be effectively used for data storage, data analysis and a variety of graphing functions. R is distributed free and it is an open source software.

## Why to learn R?

- R is open-source and freely available.
- R is cross-platform compatible.
- R has widespread acclaim.
- R has a huge, vibrant community and resource bank

So what are the advantages of over other statical software packages?

R is open-source and freely available. Unlike SAS or Matlab, you can freely install, use, update and clone R. This saves companies money, but it also allows for easy upgrades, which is useful for a statistical programming language.

Also, R is cross-platform compatible. This means that R can be run on Windows, Mac OS and Linux. It can also import data from Microsoft Excel, Microsoft Access, MySQL, SQLite, Oracle and other programs.

R is a powerful scripting language. As such, R can handle large, complex data sets. R is also amongst the top languages to use for heavy, resource intensive simulations and it can be used on high performance computer clusters.

R has widespread applications. With an estimate of more than million users, R is considered as a top programming language.

.

Finally, R has a huge, vibrant community and resource bank, with a global community of passionate users who regularly interact on discussion forums and attend

conferences.



The screenshot shows the R Project website. In the top left corner is the Kent State University logo. The main heading is "Installing R". Below it is a bullet point: "Go to <https://www.r-project.org/>". On the left is a sidebar with the R logo and links: "[Home]", "Download", "CRAN", "R Project", "About R", "Logo", and "Contributors". The main content area has the heading "The R Project for Statistical Computing" and a subheading "Getting Started". The text says: "R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred [CRAN mirror](#)." The words "download R" are circled in red. Below this, it says: "If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email." At the bottom of the main content area is the heading "News". The footer of the website shows "WWW.KENT.EDU" and a small graphic of three green leaves.

KENT STATE UNIVERSITY

## Installing R

- Go to <https://www.r-project.org/>

# The R Project for Statistical Computing

### Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To **download R**, please choose your preferred [CRAN mirror](#).

If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

### News

WWW.KENT.EDU

Let's start by installing R. The following few slides, shows an step by step guide for installing R on a Windows machine. To start, you should visit [www.r-project.org](http://www.r-project.org) and click on the download R link.

## Installing R

### CRAN Mirrors

The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. So [windows old release](#).

If you want to host a new mirror at your institution, please have a look at the [CRAN Mirror HOWTO](#).

#### 0-Cloud

<https://cloud.r-project.org/>

Automatic redirection to se

<http://cloud.r-project.org/>

Automatic redirection to se

#### Algeria

<https://cran.usthb.dz/>

University of Science and T

<http://cran.usthb.dz/>

University of Science and T

#### Argentina

<http://mirror.fcaglp.unlp.edu.ar/CRAN/>

Universidad Nacional de L

#### Australia

<https://cran.csiro.au/>

CSIRO

<http://cran.csiro.au/>

CSIRO

<https://mirror.aarnet.edu.au/pub/CRAN/>

AARNET

<https://cran.ms.unimelb.edu.au/>

School of Mathematics and

Subsequently, depending on the country of residence, the user should select a repository for downloading R.

# Installing R

## The Comprehensive R Archive Network

### Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users may want to download R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in advance.

### Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. If you do not know what this means, you probably do not want to do it.

The next step is to select an operation system.

# Installing R

## R for Windows

Subdirectories:

[base](#)

[contrib](#)

[old contrib](#)

[Rtools](#)

Binaries for base distribution. This is what you want to [install R for the first time](#)

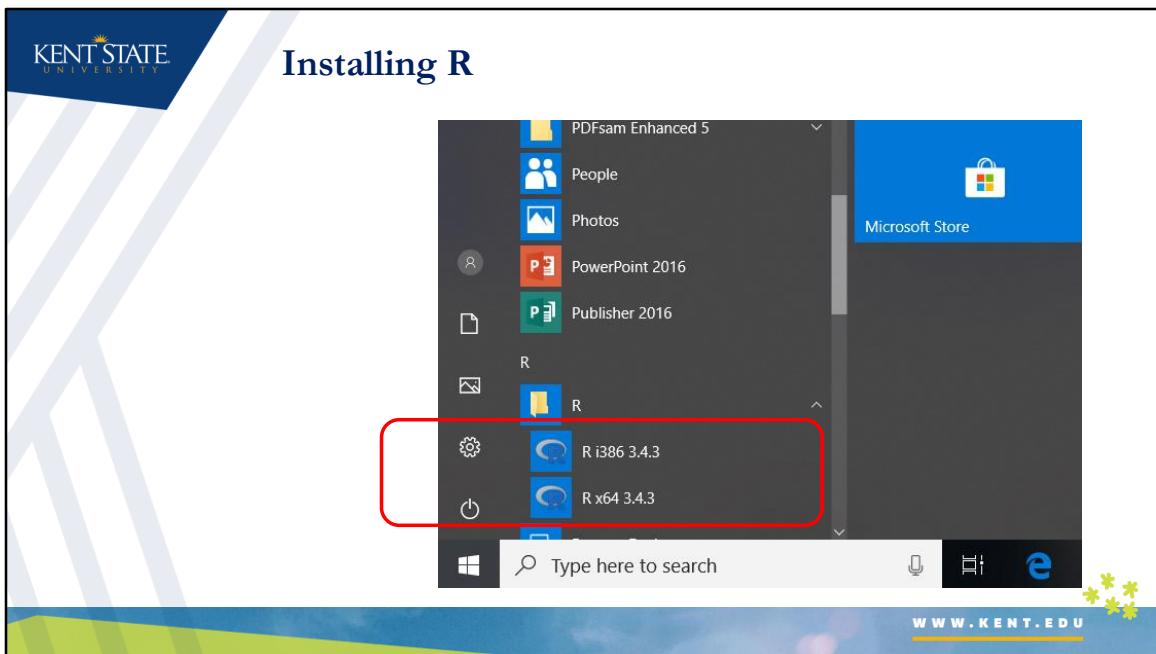
Binaries of contributed CRAN packages (for R  $\geq$  2.13.x; managed by University of Wisconsin available for CRAN Windows services and corresponding environment and

Binaries of contributed CRAN packages for outdated versions of R (for R  $<$

Tools to build R and R packages. This is what you want to build your own ]

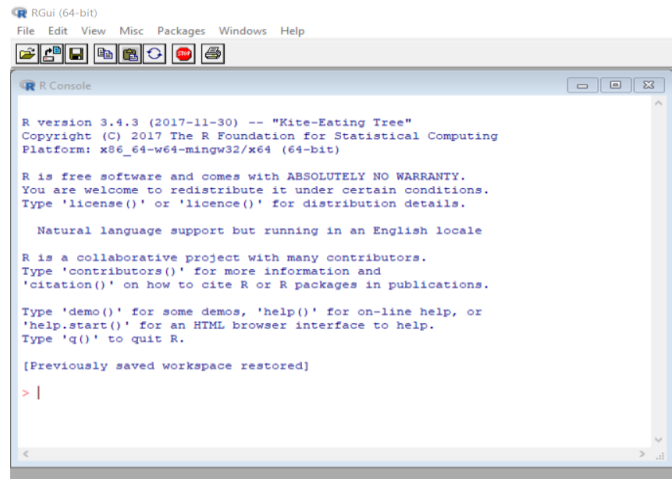
For installation of R on a windows machine, the easiest way is to click on the base and download and install from the binary file. This will be similar to installation of any other windows application.





Once the installation is finished, you should be able to find R in the list of your applications.

## Running R



The screenshot shows the R GUI (64-bit) window. The title bar reads "RGui (64-bit)". The menu bar includes "File", "Edit", "View", "Misc", "Packages", "Windows", and "Help". The toolbar contains icons for file operations and R-specific functions. The "R Console" window is open, displaying the following text:

```
R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

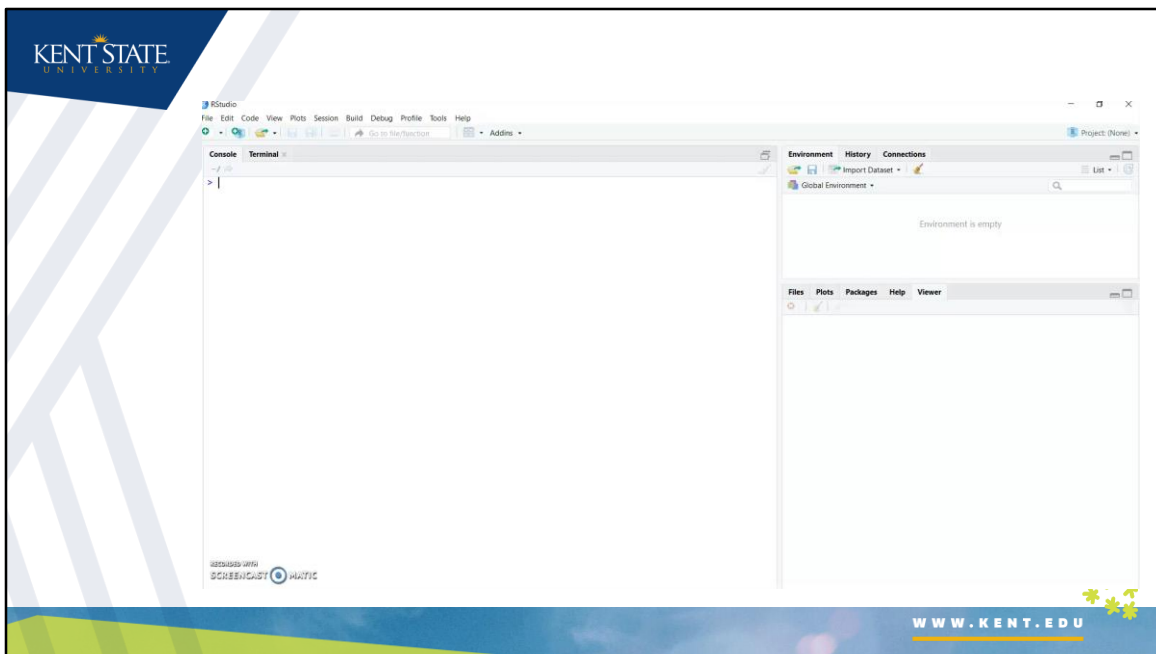
> |
```

If you click on the R on the list of your applications, a basic R GUI terminal should appear.

## R-studio

- R studio is a powerful and productive user interface for R. It's free and open source, and works great on Windows, Mac, and Linux.
- It is open-source (i.e. free) and available at <http://www.rstudio.com/>

RStudio is an integrated development environment or an IDE that allows users to interact with R more easily. RStudio is similar to the standard RGui, but is considerably more user friendly. It has more drop-down menus, windows with multiple tabs, and many customization options. R studio is supported on multiple operation systems and can be downloaded from [www.rstudio.com](http://www.rstudio.com). The next slide demonstrates different parts of R-studio.



So this is how the R studio screen looks like. In here, you can see the console and console is where you can type the commands and see the outputs. So, for example if we want to define a new variable A, we can say A is equal to one, or we can also define a data frame b, b is equal to data frame, and then we can define a variable x in that data frame and add values of one and two. So, as you can see, the console is very interactive. As soon as you write command and you press enter, the command is actually executed. In here we have got the environment tab. Environment tab shows all the active objects that are in your current r session. So, in this case, we have got a variable A, which we define with a value of one, and we also have a data frame in here with two observations and one variable. So, if you click in here, you can also see the content of the data frame. In this case, the data frame has one variable x, which has two observations or two records, or two rows, one and two in here. Also, in here you have got the history tab; the history tab shows you all the previous commands that have been executed. Then there is the connection tab, where you can basically create a connection to data bases and read data directly from the data bases. So, if you click on the new collection and see that you have the option to connect to the ODBC or spark data sources. In here, you can see that you have files where you can see all the current local files in your system. You can change your directories as well. The plots are shown in here. So, for example if you plot anything it will be shown in

here, so in that case the previous plot has been shown in here. We can execute new commands to create new plots which are shown in here. You can export these plots as well. So, if you click on here you have the option to save them as image or PDF or you can copy them into the clipboard or place them into the document. You can have the option of zooming or zooming out to the plot as well. And the package tab shows all the R packages that are currently installed in your system. And you can basically use it to also install additional packages as well. The help tab shows you, as the name implies, the help documents for any particular command that you are looking for. In this case, if I print for example question mark plot, then this would be shown to me as you can see in here, also you can search for commands from here as well. Also. another interesting point is if you want to basically write your commands in a script, so that you can save them and modify them later on, you can create a new file and create a new r script and in this environment you can basically type all of your commands. A is equal to one and B is equal to two and ten so on and so forth. The difference between the script and the console is that the script is basically that the script is not interactive which means that in order to execute these commands, you need to select them and press the run button, or you can save them into the file and call them by the name of the file. You can save this file as the R script file and you can open them and modify them later.

## R Packages

- What is an R package?
  - In R, the fundamental unit of shareable code is the package.
  - A package bundles together code, data, documentation, and tests, and is easy to share with others.
- Once installed a copy of the package lives on the computer and does not need to be reinstalled
- Updating R
  - Must reinstall packages
  - May lose packages that are not kept updated

Packages are the fundamental units of reproducible R code. They include reusable R functions, the documentation that describes how to use them, and sample data. Thousands of R packages have been developed for different purposes which makes R a **very powerful tool**. Once installed a copy of a package stay on the computer and does not need to be reinstalled. However, you may need to call the package by using the `library()` function when you want to use the package. Also, note that after updating R you may need to reinstall some of the packages.

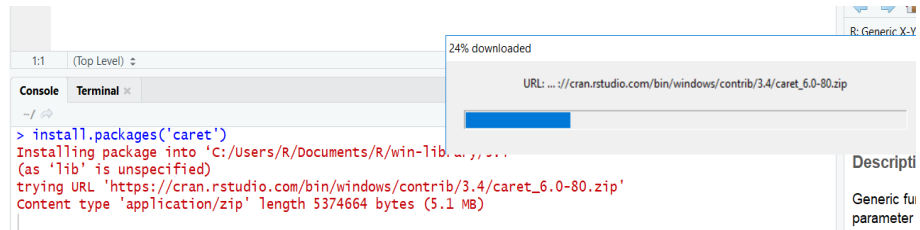
## Installing R packages

- `install.packages()` function can be used to install new R packages.
- The function automatically connects to the Comprehensive R Archive Network (CRAN) repository to find the package and download and install it. Therefore, make sure you are connected to the Internet.
- After you install a package, you need to use the function `library()` to load and attach the function before you can use it.

The easiest way to install a package is to use the `install.packages()` function. If you are connected to the internet, the function automatically search the R archive network, known as CRAN to find, download and install the package. Once a package is installed, you can use the `library()` function to load the package.

## Example

- Installing the 'caret' package



```
> install.packages('caret')
Installing package into 'C:/Users/R/Documents/R/win-lib...'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/caret_6.0-80.zip'
Content type 'application/zip' length 5374664 bytes (5.1 MB)
```

24% downloaded

URL: .../cran.rstudio.com/bin/windows/contrib/3.4/caret\_6.0-80.zip

R: Generic X-Y

Descripti

Generic fu

parameter

Here is an example of installing a package called 'caret'. `install.packages('caret')` will download and install the 'caret' package .



## Example

```
> install.packages('caret')
Installing package into 'C:/Users/R/Documents/R/win-library/3.4'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/caret_6.0-80.zip'
Content type 'application/zip' length 5374664 bytes (5.1 MB)
downloaded 5.1 MB

package 'caret' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\R\AppData\Local\Temp\RtmpqGBHn\downloaded_packages
> library('caret')
Loading required package: lattice
Loading required package: ggplot2
RStudio Community is a great place to get help:
https://community.rstudio.com/c/tidyverse.
Warning messages:
1: package 'caret' was built under R version 3.4.4
2: package 'ggplot2' was built under R version 3.4.4
> |
```

Once a package is installed you need to use the `library()` function to load the package. Note that when a package is loaded it may have dependency and requires other packages to load as well. Such messages are printed in red but they do not imply an error. Once a package is loaded using the `library()` function, we can use any function or dataset that is included in that package.

*"I don't think anyone actually believes that R is designed to make everyone happy. For me, R does about 99% of the things I need to do, but sadly, when I need to order a pizza, I still have to pick up the telephone!"*

*-- Roger D. Peng*