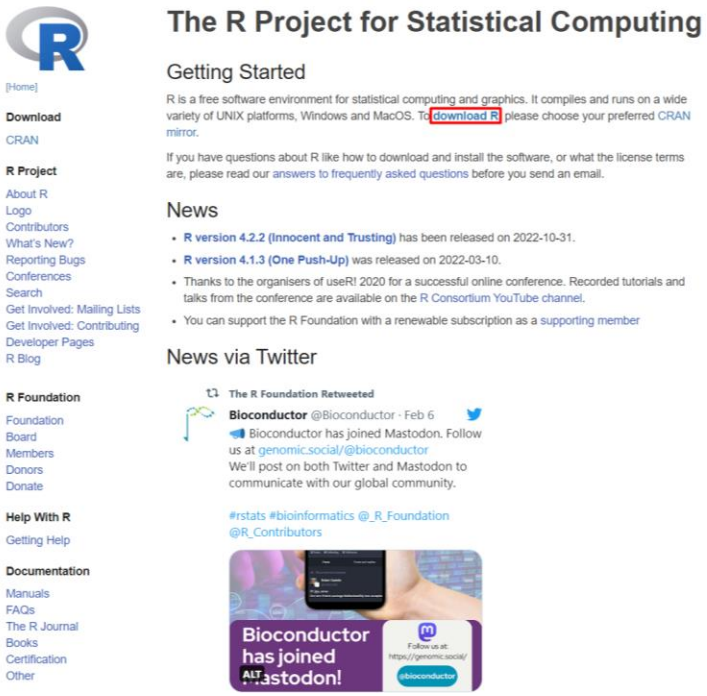
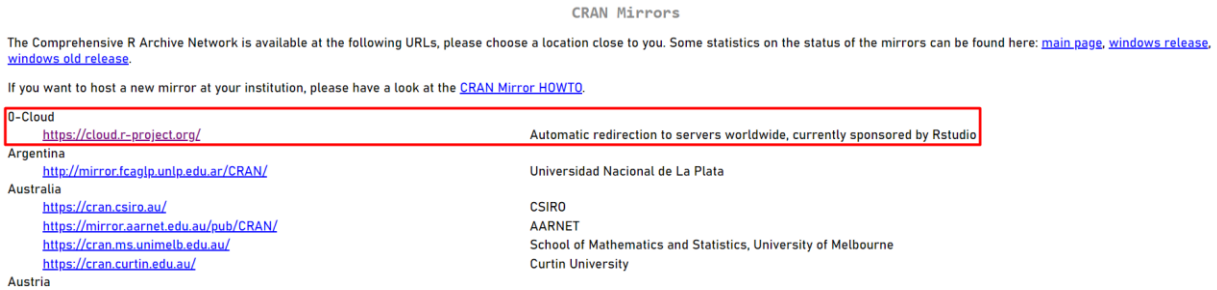
M1-FA1: Fundamentals and Concepts

I. Installing R and RStudio

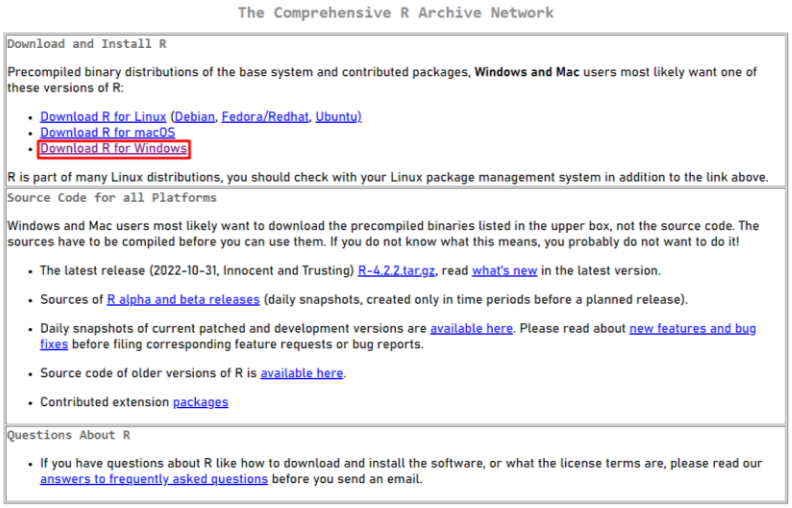
Using the provided R Programming Tutorial installation guide (R: The R Project for Statistical Computing, n.d.), programs R and RStudio were installed by following the steps below. The first programming tool utilized in the guide was R, whose executable file can be downloaded from here. Clicking the hyperlink should redirect the user to the website seen in the figure below.



As highlighted in the figure, the *download R* hyperlink was clicked to reach the Comprehensive R Archive Network (CRAN) Mirrors page, where the link below the *0-Cloud* option was selected. (The *0-Cloud* mirror provides automatic redirection to servers around the world and is currently sponsored by RStudio.)



After clicking the link, a new page should appear, where different versions of R could be downloaded according to three operating systems: Linux, MacOS, and Windows. Since the operating systems utilized by the members of this group is Windows, the option to download the appropriate operating system was selected.



The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- Download R for Linux (Debian, Fedora/Redhat, Ubuntu)
- Download R for macOS
- Download R for Windows

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

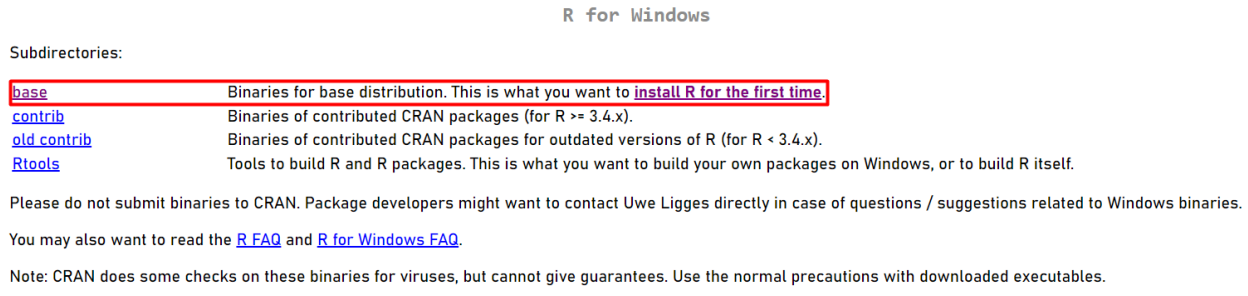
Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2022-10-31, Innocent and Trusting) R-4.2.2.tar.gz, read what's new in the latest version.
- Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are available here. Please read about new features and bug fixes before filing corresponding feature requests or bug reports.
- Source code of older versions of R is available here.
- Contributed extension packages

Questions About R

- 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.

Next, under *Subdirectories*, the *base* hyperlink was selected because R was being installed into local machines that have not had R installed into them yet.



R for Windows

Subdirectories:

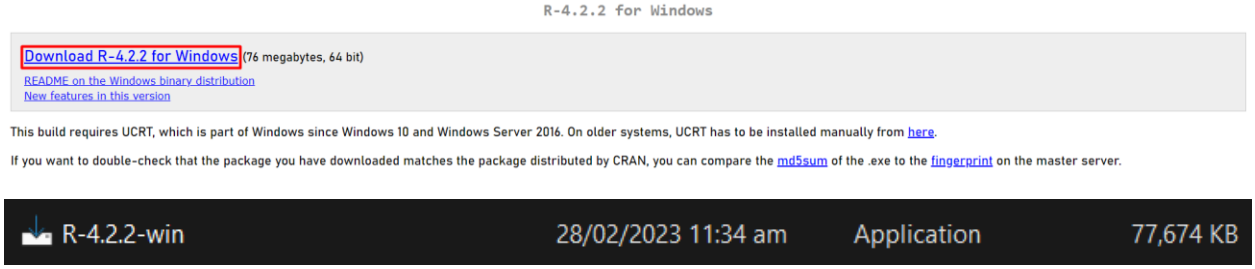
base	Binaries for base distribution. This is what you want to install R for the first time.
contrib	Binaries of contributed CRAN packages (for R >= 3.4.x).
old contrib	Binaries of contributed CRAN packages for outdated versions of R (for R < 3.4.x).
Rtools	Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

Please do not submit binaries to CRAN. Package developers might want to contact Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the R FAQ and R for Windows FAQ.

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.

Then, to download the executable file that will install R, the highlighted hyperlink was clicked, and the installer was run.



R-4.2.2 for Windows

Download R-4.2.2 for Windows (76 megabytes, 64 bit)

README on the Windows binary distribution

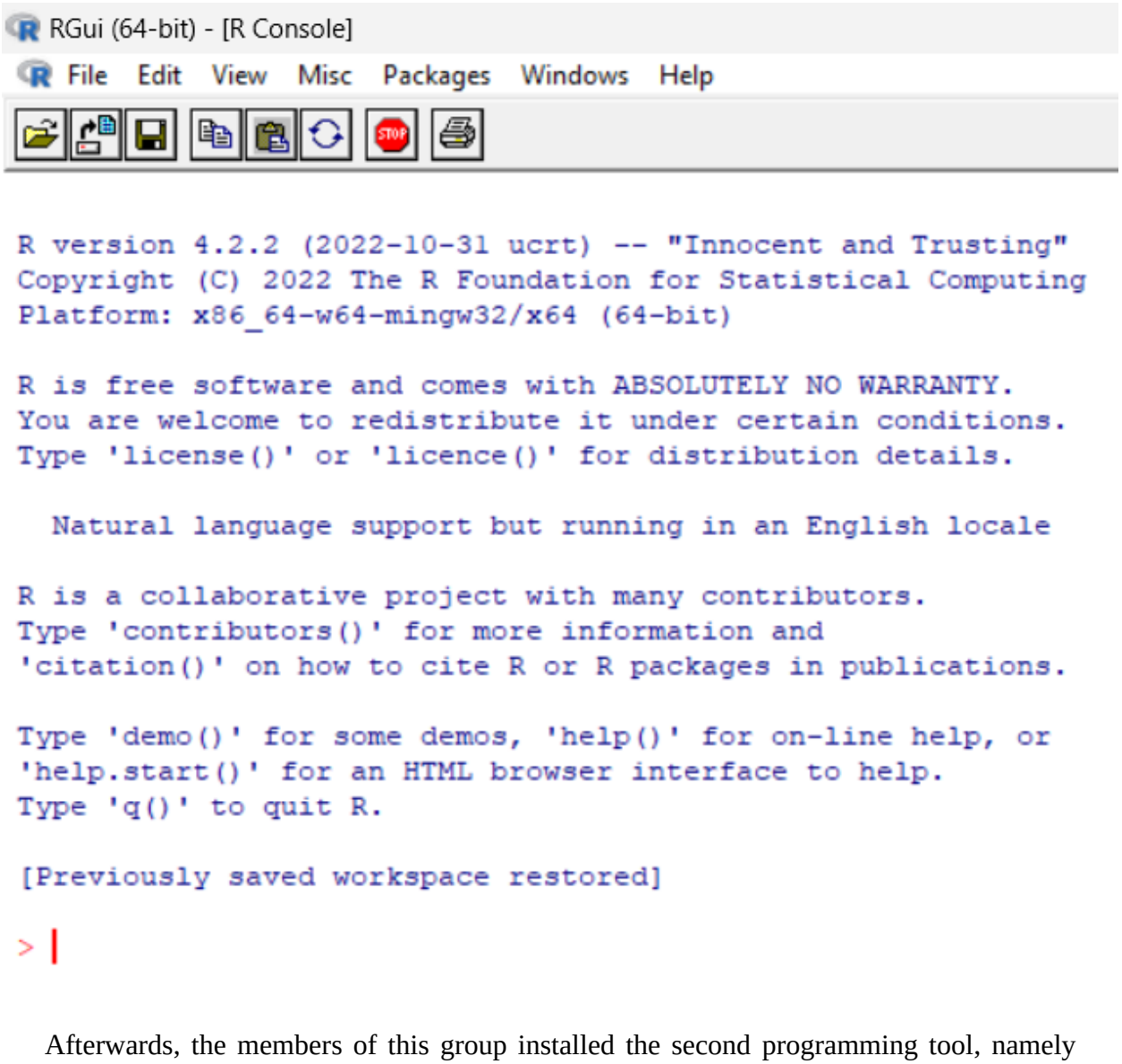
New features in this version

This build requires UCRT, which is part of Windows since Windows 10 and Windows Server 2016. On older systems, UCRT has to be installed manually from here.

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the md5sum of the .exe to the fingerprint on the master server.

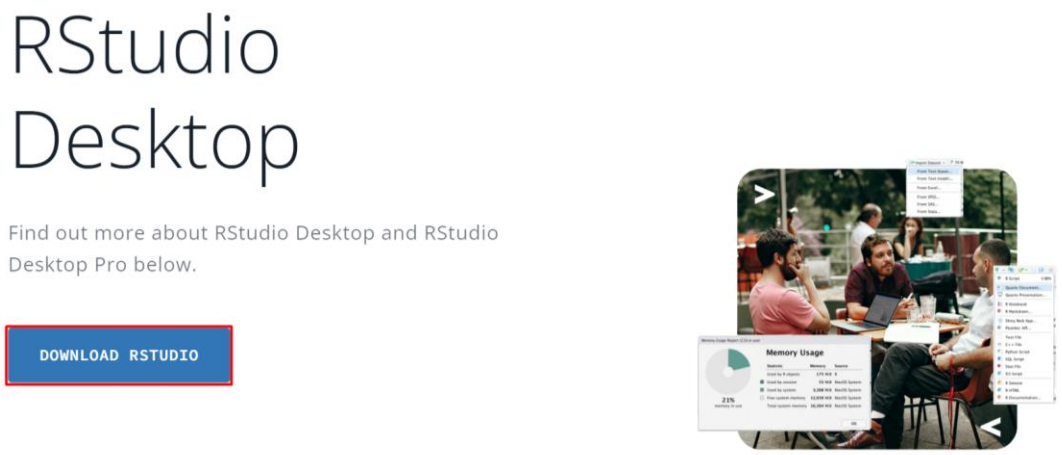
R-4.2.2-win 28/02/2023 11:34 am Application 77,674 KB

R was opened by double-clicking the shortcut that was made upon the application's installation. Selecting the icon should open the R Console, as seen in the figure below.

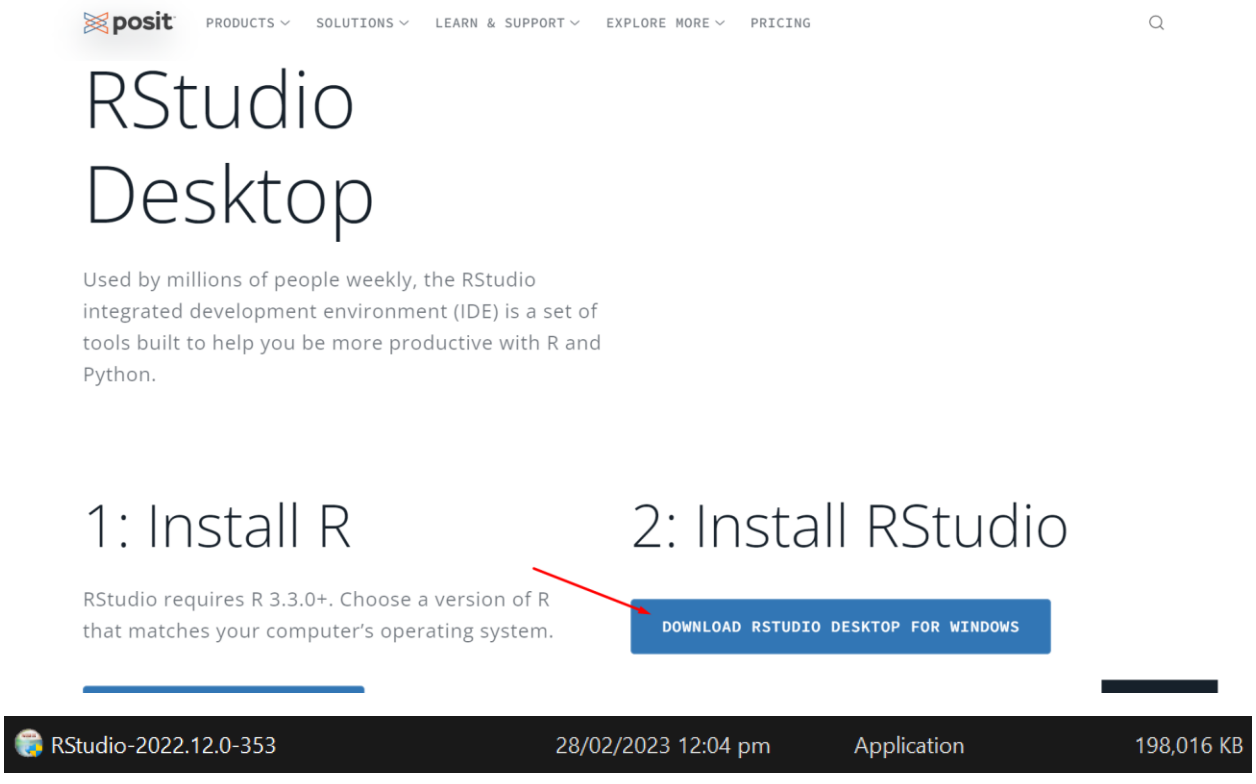


Afterwards, the members of this group installed the second programming tool, namely RStudio, from this link. Clicking the link should redirect the user to the website, as seen in the figure below. Click **DOWNLOAD RSTUDIO**.

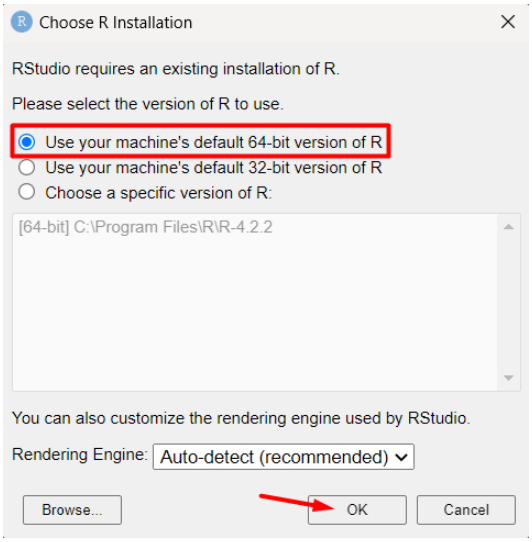
Clicking the button should redirect the user to the downloads page, where they will be given the option to choose between downloading *RStudio Desktop* or *RStudio Server*. For this process, the *RStudio Desktop* was selected by clicking the highlighted button in the figure below.



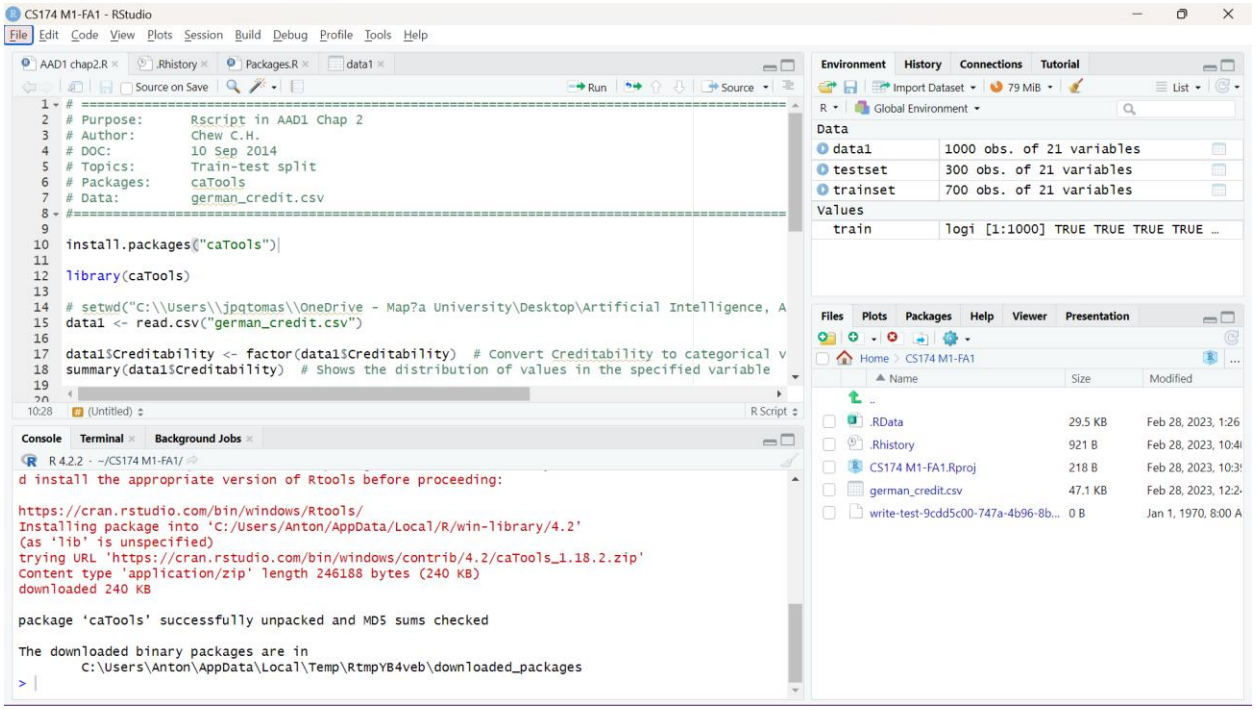
Click **DOWNLOAD RSTUDIO DESKTOP FOR WINDOWS** to download the executable file to install RStudio Desktop for Windows.



Next, open RStudio by selecting the rstudio application inside the RStudio folder in the C: drive. The following pop-up window should appear. Select the default option, then click OK.



Since RStudio had already been used by the data scientists, opening RStudio resulted in the figure below. Nevertheless, the same contents and windowpanes should still appear upon initial startup.



II. Running the R Script

Before running the provided R script, the following line of code was added prior to the first line to install the required library package that contains basic utility functions like window statistic functions, LogitBoost classifier, and base64 encoder/decoder, etc. (Jarek Tuszynski jaroslaw.w.tuszynski@saic.com, n.d.)

```
install.packages("caTools")
```

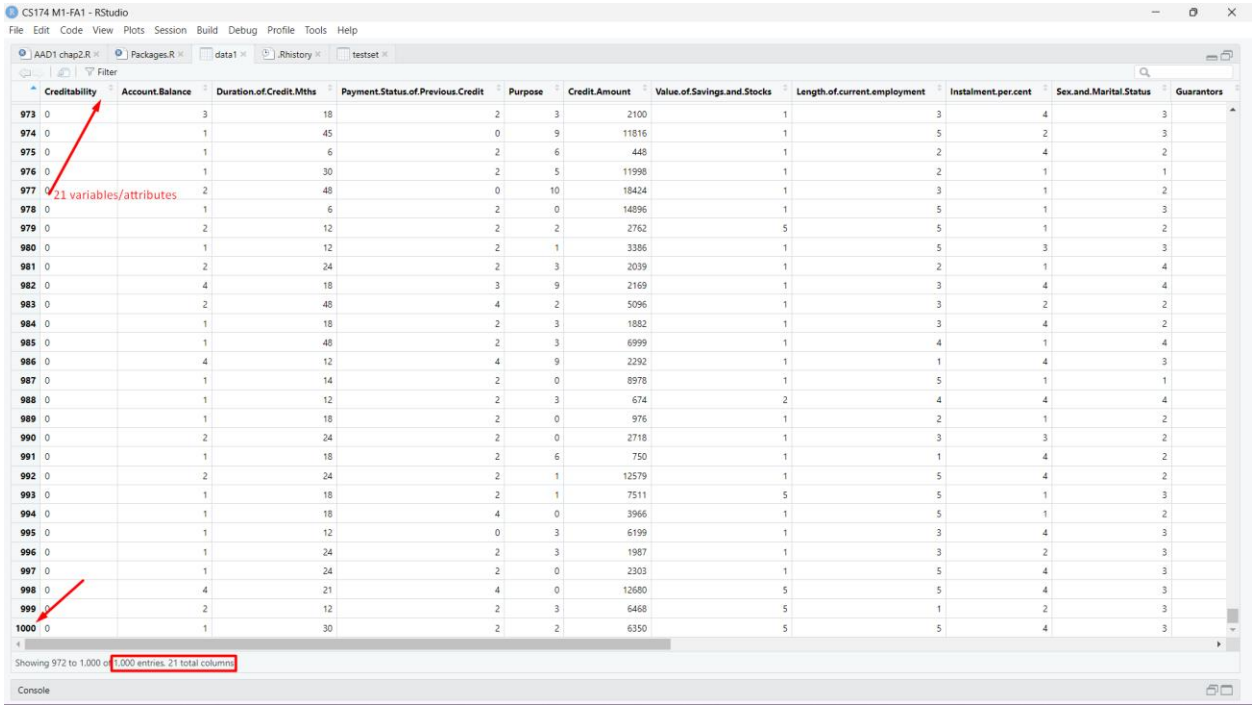
```
> install.packages("caTools")
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proc
eeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/Anton/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
also installing the dependency 'bitops'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/bitops_1.0-7.zip'
Content type 'application/zip' length 31679 bytes (30 KB)
downloaded 30 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/caTools_1.18.2.zip'
Content type 'application/zip' length 246188 bytes (240 KB)
downloaded 240 KB

package 'bitops' successfully unpacked and MD5 sums checked
package 'caTools' successfully unpacked and MD5 sums checked
```

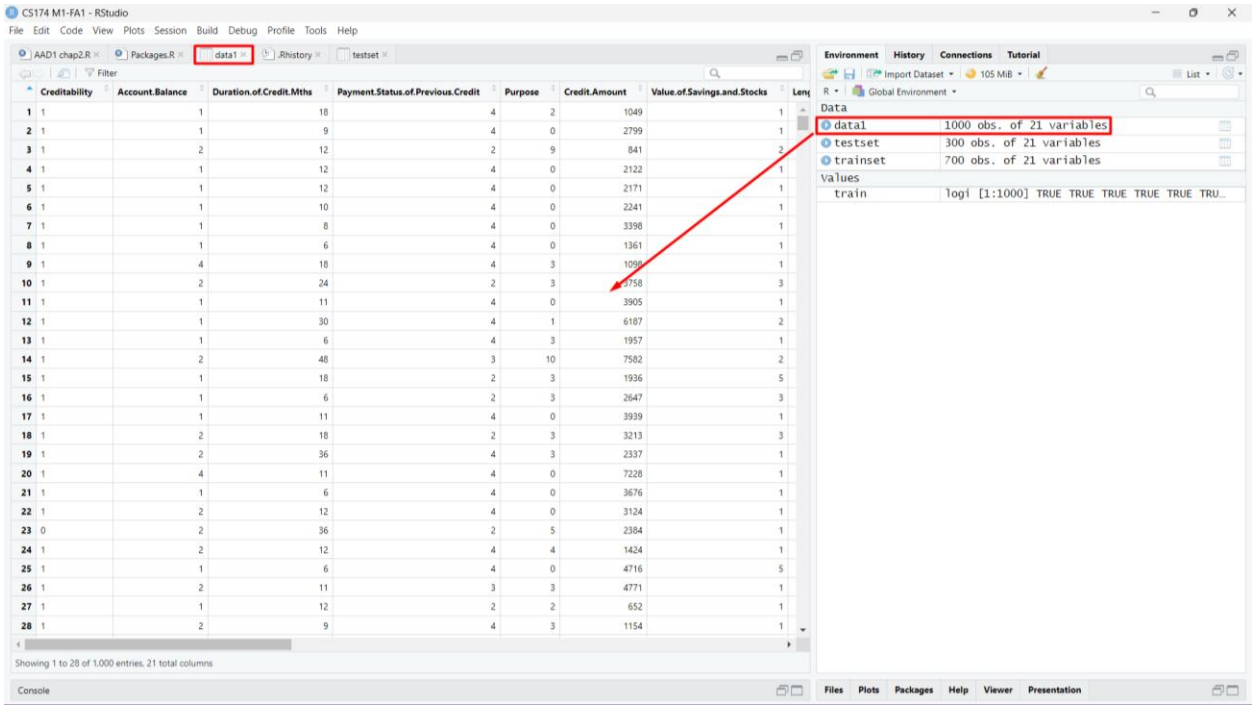
After the caTools package has been installed, the rest of the code was run without errors, as seen in the code snippet below.

```
> library(caTools)
> data1 <- read.csv("german_credit.csv")
> data1$Creditability <- factor(data1$Creditability)  # Convert Creditabili
ty to categorical variable
> summary(data1$Creditability)  # Shows the distribution of values in the s
pecified variable
  0   1
300 700
> prop.table(table(data1$Creditability))  # Shows the proportion of values
in the specified variable.

  0   1
0.3 0.7
> # set the random number sequence for random split.
> # Reproducible in future with the same number that you choose.
> set.seed(2014)
> # Stratify on Y and randomly split data into train vs test set based on S
plit ratio
> train <- sample.split(Y = data1$Creditability, SplitRatio = 0.7)
> # Get training and test data
> trainset <- subset(data1, train == T)
> testset <- subset(data1, train == F)
> # Check that the proportion of Y is the same in Trainset and Testset
> prop.table(table(trainset$Creditability))

  0   1
0.3 0.7
> prop.table(table(testset$Creditability))

  0   1
0.3  0.7
```

III. Determining the Number of Observable Data

The total number of observable data from the dataset is **twenty-one thousand entries**, with one thousand rows per column. (The dataset contains twenty-one attributes, each of which are listed per column.) A more comprehensive understanding of this description can be found below.





IV. Determining the Percentage of the Training and Test Set

Before the percentages of the training and testing set were identified, the dataset was first randomly split into training and testing sets based on a split ratio of 7.0. This was accomplished by running the following code snippet:

```
> # Stratify on Y and randomly split data into train vs test set based on
Split ratio
> train <- sample.split(Y = data1$Creditability, SplitRatio = 0.7)
```

Note: The line of code above did not yield any output because it merely performed the random splitting of the dataset.

To get the proportions of the training and test sets, the following codes were run. First, Boolean values of True (1) and False (0) were assigned to the trainset and testset, respectively, using the subset() function.

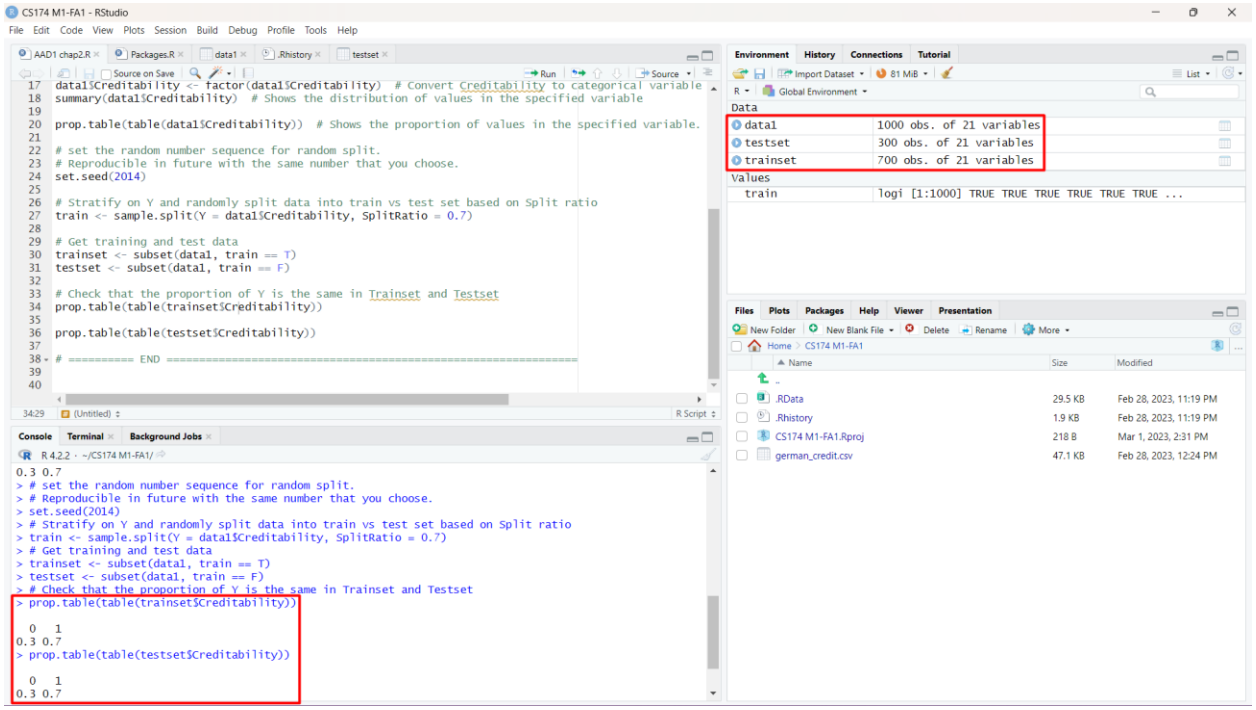
```
> # Get training and test data
> trainset <- subset(data1, train == T) # T is equivalent to 1
> testset <- subset(data1, train == F) # F is equivalent to 0
```

According to (Quick-R: Subsetting Data, n.d.), the subset() function is the most efficien t way to select variables and observations. Therefore, the lines of code above state that all ro ws from the dataset with a Boolean value of True or 1 were specified as the trainset and all r ows from the dataset with a Boolean value of False or 0 were specified as the testset.

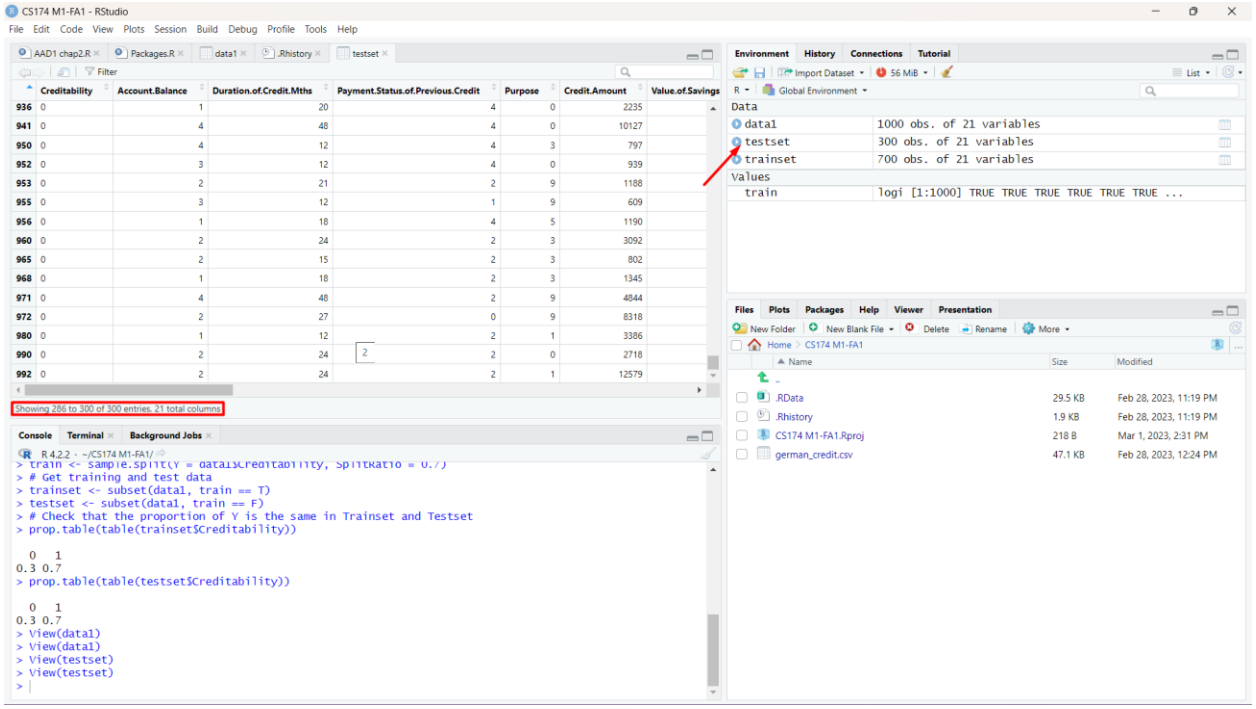
```
> # Check that the proportion of Y is the same in Trainset and Testset
> prop.table(table(trainset$Creditability))

  0   1
0.3 0.7
> prop.table(table(testset$Creditability))

  0   1
0.3 0.7
```

Then, to check that the split was correctly implemented and that the proportion of Y is the same as the training and testing set, the prop.table() function was run to calculate the value of each table cell as a proportion of all values. (2021) The results reveal that the dataset was divided into proportions of 0.3 and 0.7 for the testing set (Boolean value = 0) and the training set (Boolean value = 1), respectively. Furthermore, the percentages of the training and testing sets can be inferred from the results: **70% of the dataset was proportioned for the training set** and **30% of the dataset for the testing set**.

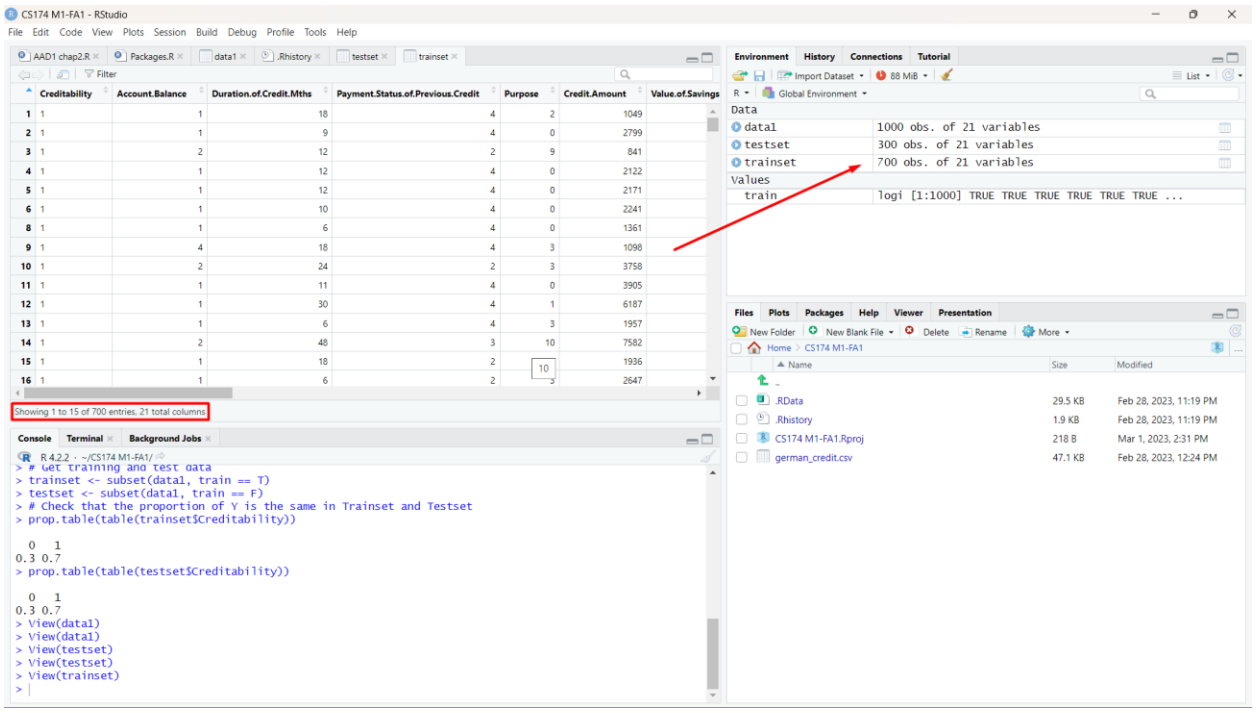
The screenshot below shows a visualization of how the dataset was split into the testset and trainset.



Testing and Training Set Proportions



Testing Set Proportion (30%)

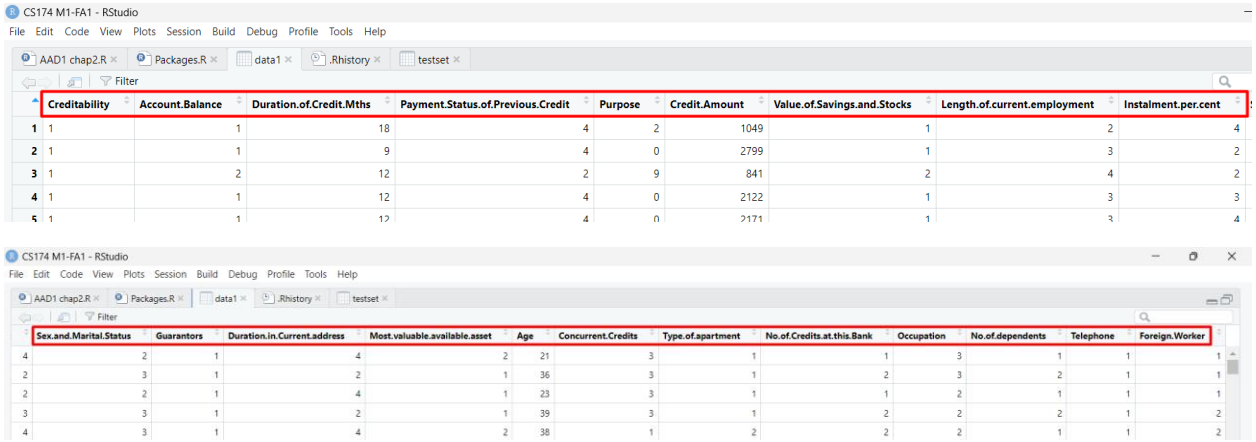


Training Set Proportion (70%)

V. Listing All Variables

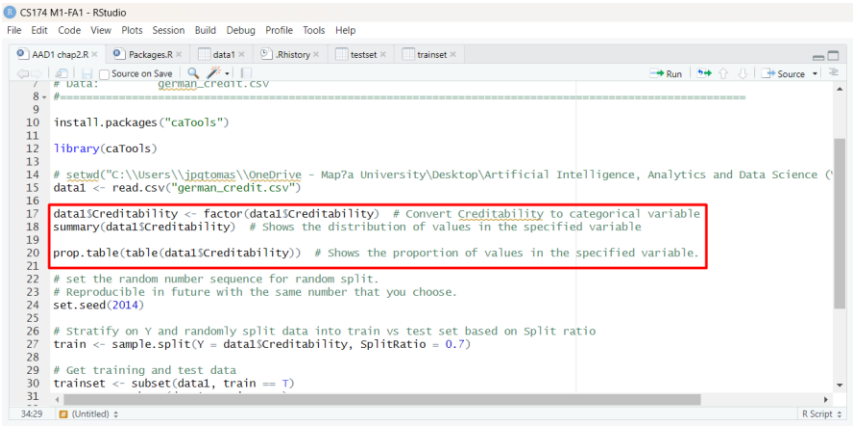
The provided dataset (*german_credit.csv)* includes the following variables: *Creditability*, *Account Balance*, *Duration of Credit Mths*, *Payment Status of Previous Credit*, *Purpose*, *Credit Amount*, *Value of Savings and Stocks*, *Length of current employment*, *Instalment per cent*, *Sex and Marital Status*, *Guarantors*, *Duration in Current address*, *Most valuable available asset*, *Age*, *Concurrent Credits*, *Type of apartment*, *No of Credits at this Bank*, *Occupation*, *No of dependents*, *Telephone*, and *Foreign Worker*.

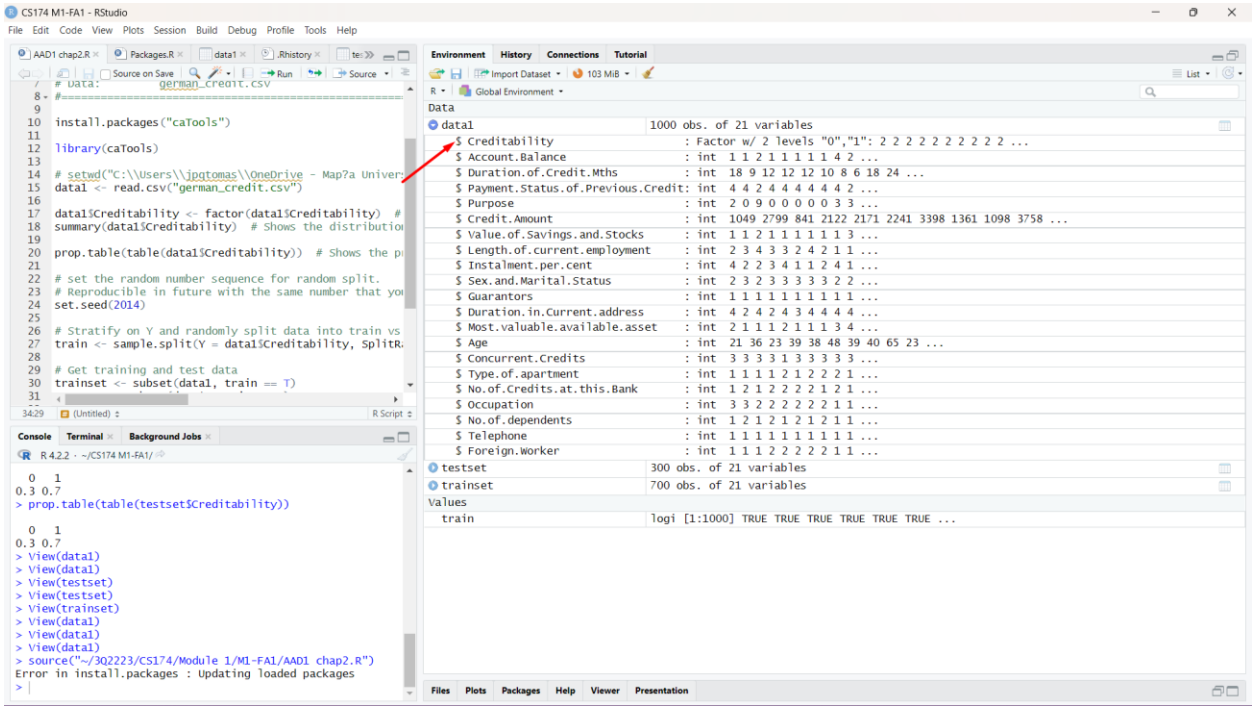
The screenshots below show the dataset variables that were read through RStudio.



VI. Identifying Three Qualitative and Three Quantitative Attributes

Quantitative variables are referred to as “numeric” variables, which are variables that represent a measurable quantity. On the other hand, qualitative variables are sometimes referred to as “categorical” variables, which are those that take on names or labels that fit into categories. (2020) Three quantitative variables from the dataset include *Credit Amount*, *Instalment per cent*, and *Account Balance*. Three qualitative variables from the dataset include *Creditability*, which was converted to a categorical variable in the code, *Purpose*, and *Occupation*.





***Credibility* as a categorical (quantitative) variable**

References

freeCodeCamp.org. (2019, June 6). *R Programming Tutorial - Learn the Basics of Statistical Computing*. YouTube. https://www.youtube.com/watch?v=_V8eKsto3Ug&feature=youtu.be

Jarek Tuszynski jaroslaw.w.tuszynski@saic.com. (n.d.). *caTools-package function - RDocumentation*. https://www.rdocumentation.org/packages/caTools/versions/1.17.1/topics/caTools-package

Quick-R: Subsetting Data. (n.d.). https://www.statmethods.net/management/subset.html

R: The R Project for Statistical Computing. (n.d.). https://www.r-project.org/

Z. (2020, September 18). *Qualitative vs. Quantitative Variables: What's the Difference?* Statology. https://www.statology.org/qualitative-vs-quantitative-variables/

Z. (2021, December 3). *How to Use prop.table() Function in R (With Examples)*. Statology. https://www.statology.org/r-prop-table/