
Pension Simulation Calculator Manual

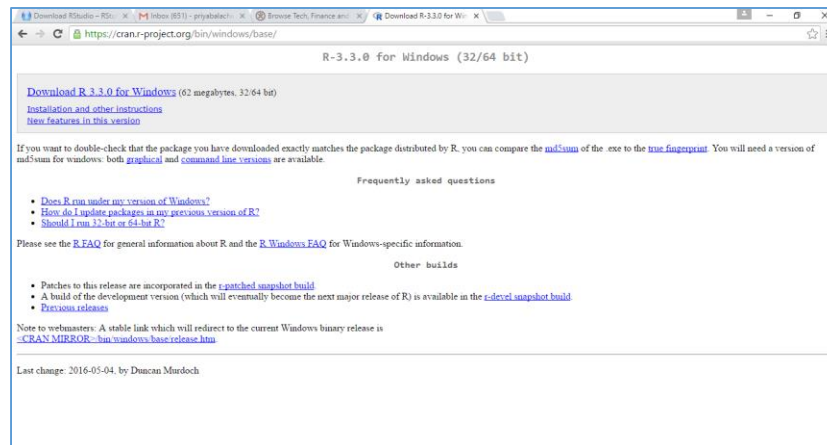
TABLE OF CONTENTS

1	Environment Setup	3
1.1	Install R 3.2.4	3
1.2	Install Rstudio 0.99.902	6
1.3	Install R packages.....	8
2	Project Setup.....	9
3	List Of Files	11
4	ShinyApps.io setup and deployment	13

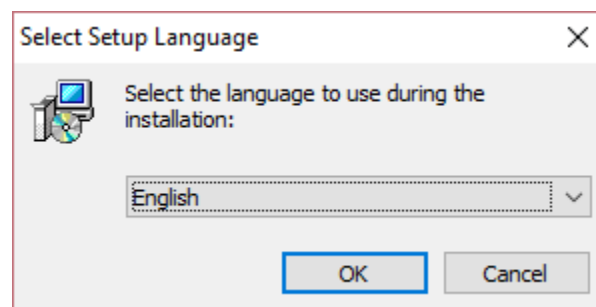
1. Environment Setup

1.1 Install R 3.2.4

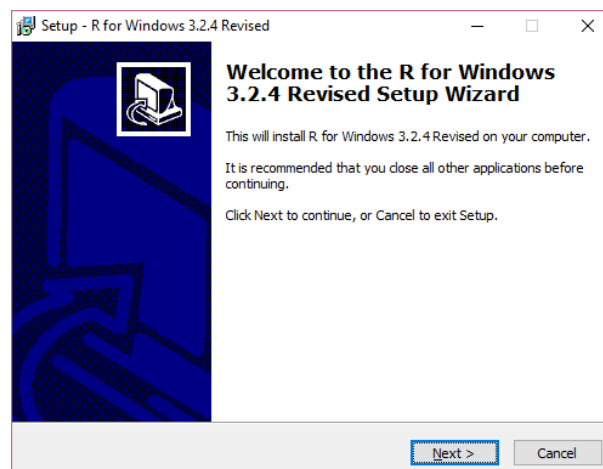
- 1.1.1 Download R 3.2.4 for windows from the CRAN website <https://cran.r-project.org/bin/windows/base/> from the previous releases link.

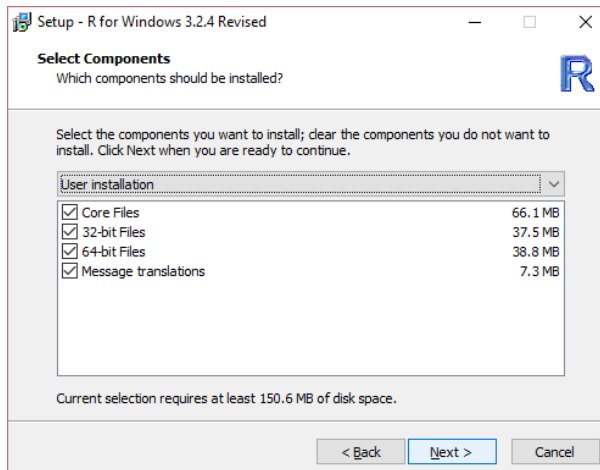
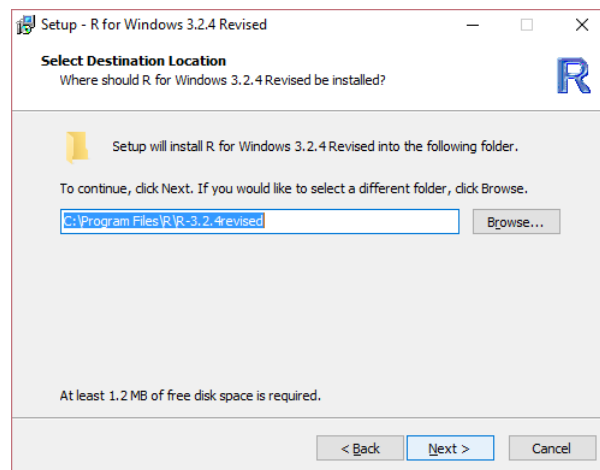
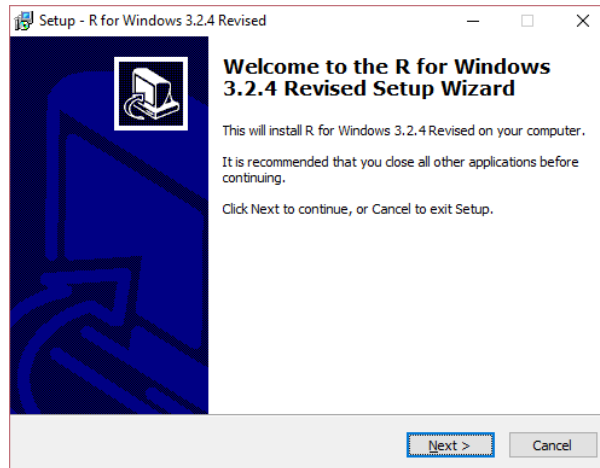


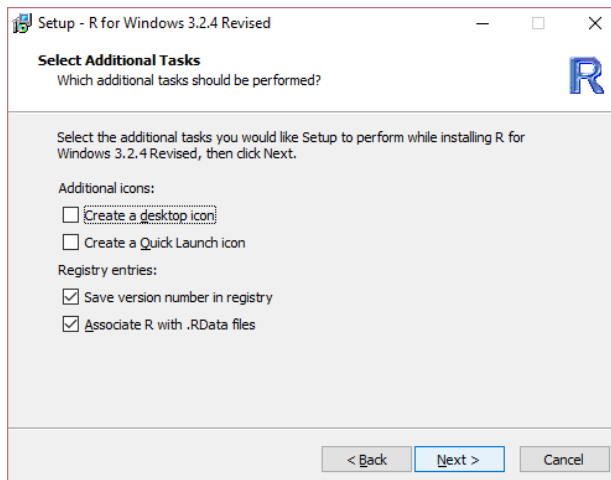
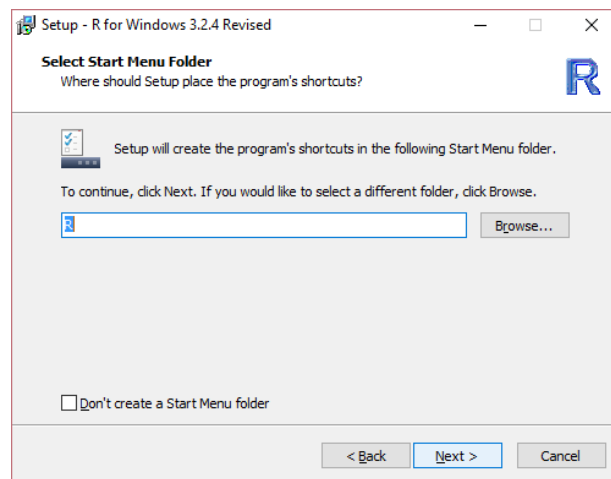
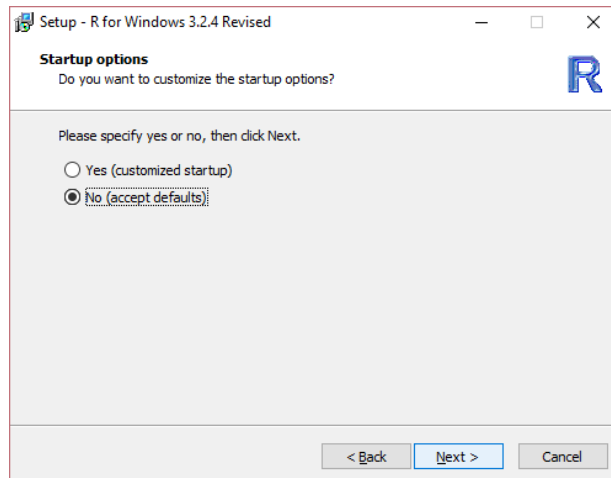
- 1.1.2 Select the language for the R setup

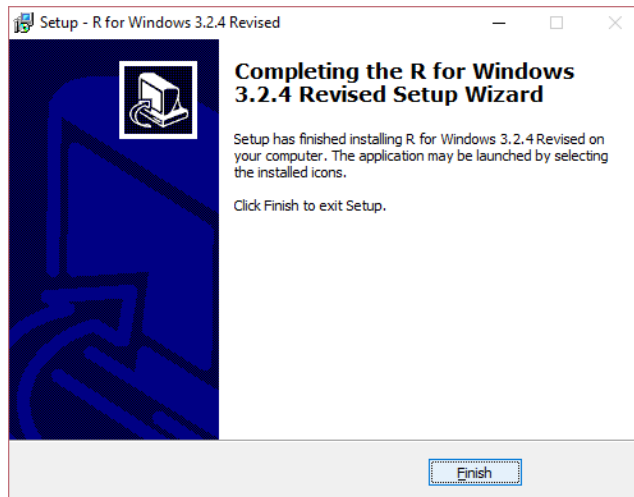


- 1.1.3 Follow the set of screenshots for the default R 3.3.0 Windows installation



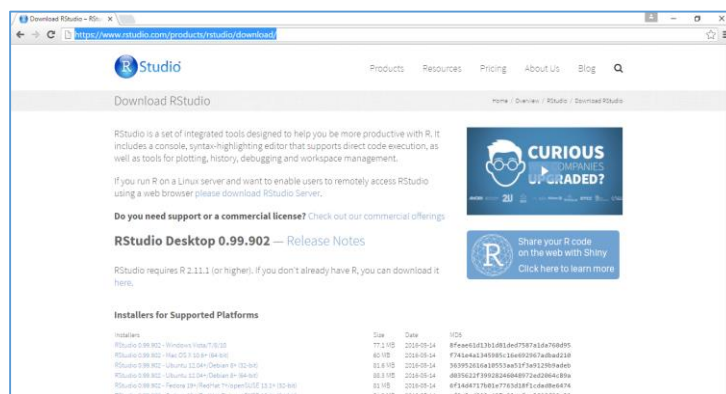




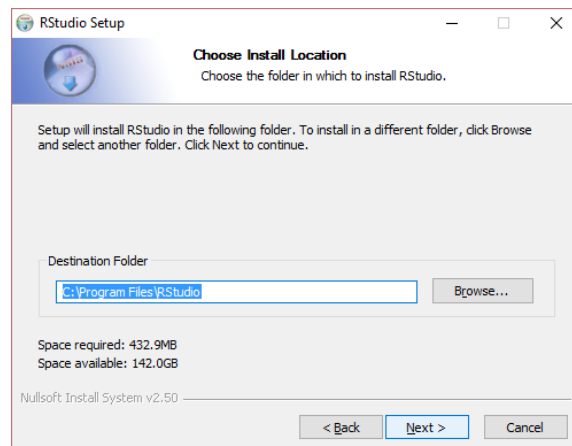


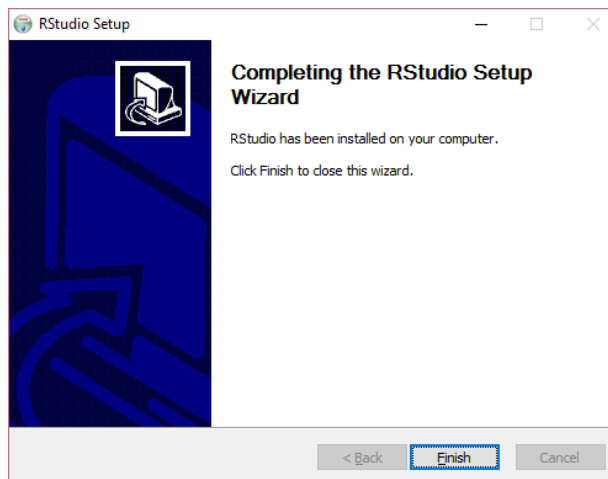
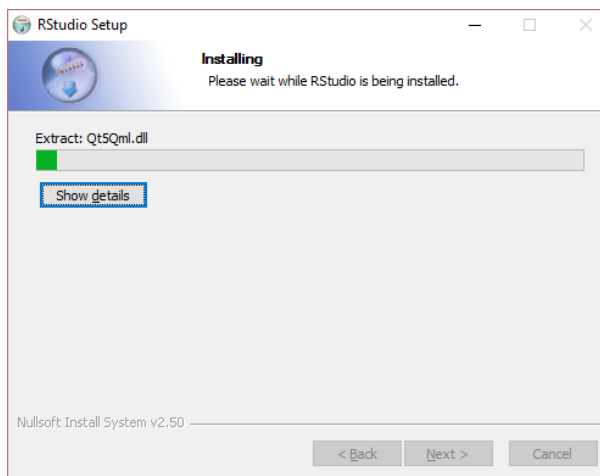
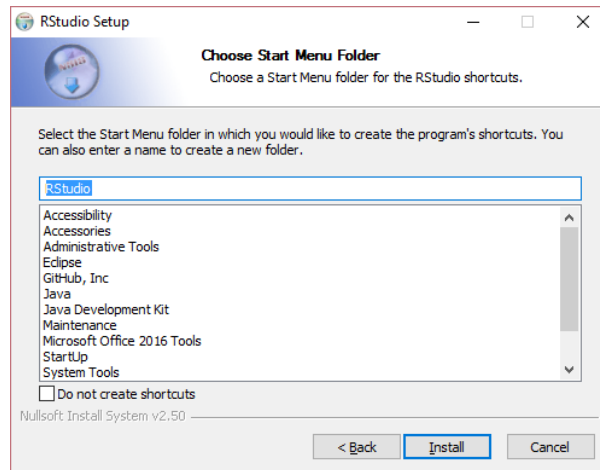
1.2 Install RStudio 0.99.902

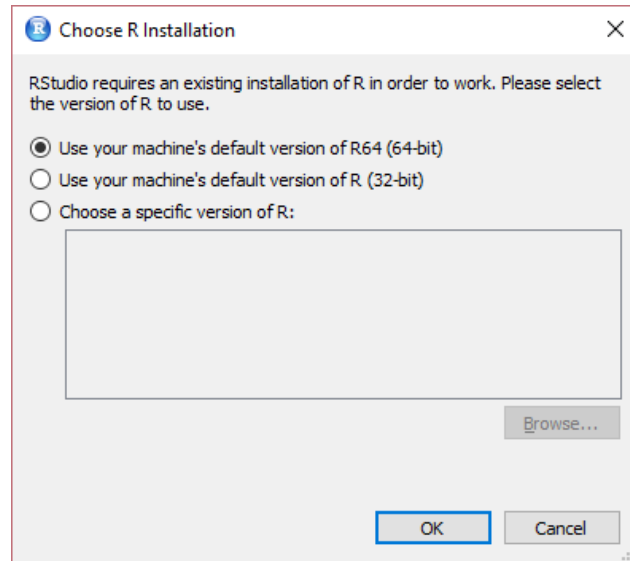
1.2.1 Download RStudio 0.99.902 from <https://www.rstudio.com/products/rstudio/download/>



1.2.2 Run the binary for RStudio and follow the set of screenshots for the installation

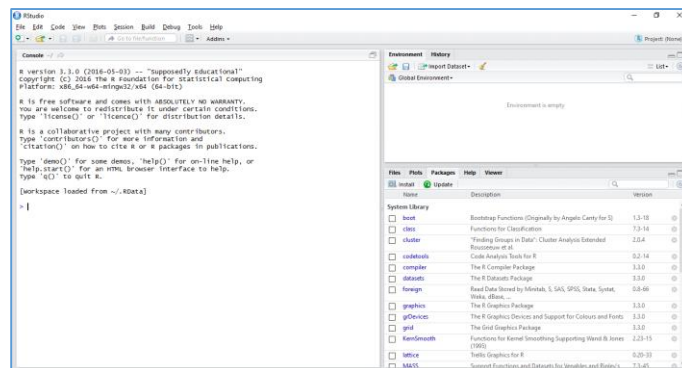




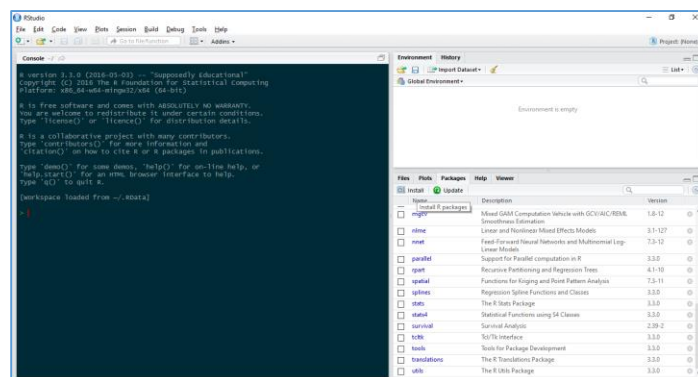


1.3 Install R packages

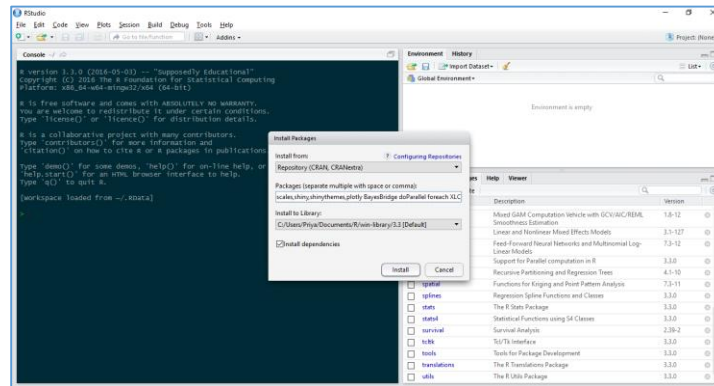
1.3.1 Open RStudio and click on packages tab as shown below



1.3.2 Click on install R packages

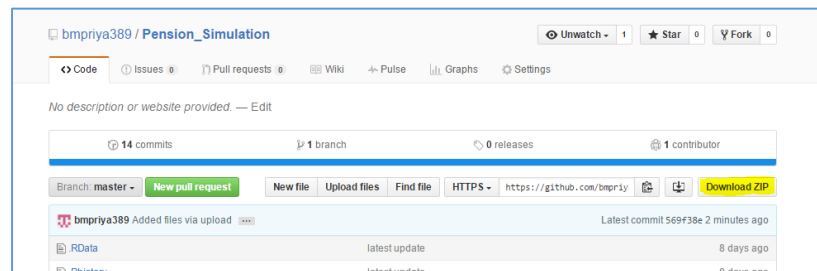


1.3.3 Install the packages scales, shiny, shinythemes, plotly, BayesBridge, doParallel, foreach, XLConnect, xlsx, xlsxjars, devtools



2 Project Setup

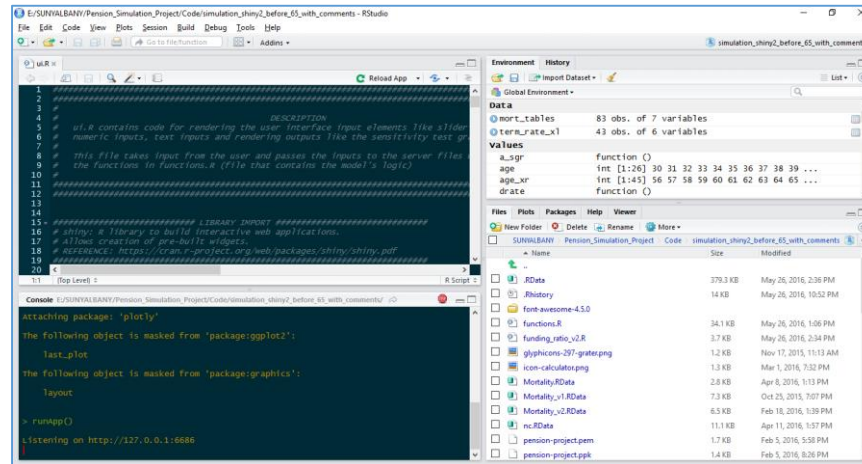
2.1 Download zip file from https://github.com/bmpriya389/Pension_Simulation and extract file at desired location.



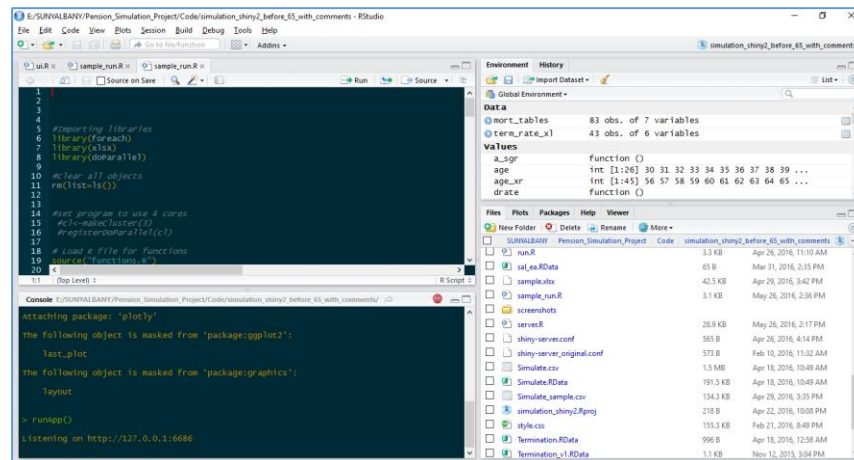
2.2 Go to File> New Project, select the “Existing Directory” option and provide the path to the extracted zip folder in step 1.4.1. Click on the ‘Create Project’ button.

By now, all the files necessary to run the model, dataset and shiny application is available.

2.3 To run the Shiny application open 'ui.R' or 'server.R' from the files window and click on RunApp button.



2.4 To generate a sample data set with 100 observations open the sample_run.R from the files section. Select all the text in the file as shown in the figure below and click on the run button. The sample dataset with the 100 observations will be generated in the zip file location from step 1.4.1.



3 List of files

1. functions.R

This file contains the logic for the entire model. The server makes function calls to functions.R whenever there is a change in user input.

2. ui.R

This file contains code for rendering the user interface input elements like sliderinputs, numeric inputs, text inputs and rendering outputs like the sensitivity test graphs. This file takes input from the user and passes the inputs to the server files which calls the functions in functions.R (file that contains the model's logic).

3. server.R

server.R contains code that monitors the user input and makes calls to the model's logic dynamically whenever the value set for an UI element (text inputs, numeric inputs, slider bars) is changed and refreshes the output graphs generated. This file creates the actual graph output based on the dataframe or vector returned by the function call made.

4. test_function.R

This file allows you to enter Values for desired inputs and check vectors and values for all functions in functions.R.

5. run.R

This file creates a dataframe and xlsx sheet of observations of ARC, Normal Cost, AAL, expected value of annuity, retirement annuity, replacement rate for all possible combinations of entry age, retirement age, current age, funding level, discount rate, salary growth rate, payroll growth rate(%), COLA, AFC, benefit factor, cost method, mortality table, amortization period in the current working directory.

6. sample_run.R

This file creates a dataframe and xlsx sheet of 100 observations of ARC, Normal Cost, AAL, expected value of annuity, retirement annuity, replacement rate for an random combination entry age, retirement age, current age, funding level, discount rate, salary growth rate, payroll growth rate(%), COLA, AFC, benefit factor, cost method, mortality table, amortization period in the current working directory.

7. mortality.R

This file contains code for extracting data from an excel sheet titled 'mortality-reference-table' and making an RData file containing this data.

This file is later accessed to by the functions.R file to utilize the mortality rates.

8. PPD_import.R

This file imports data from the PPD using the PPD API

9. Termination.R

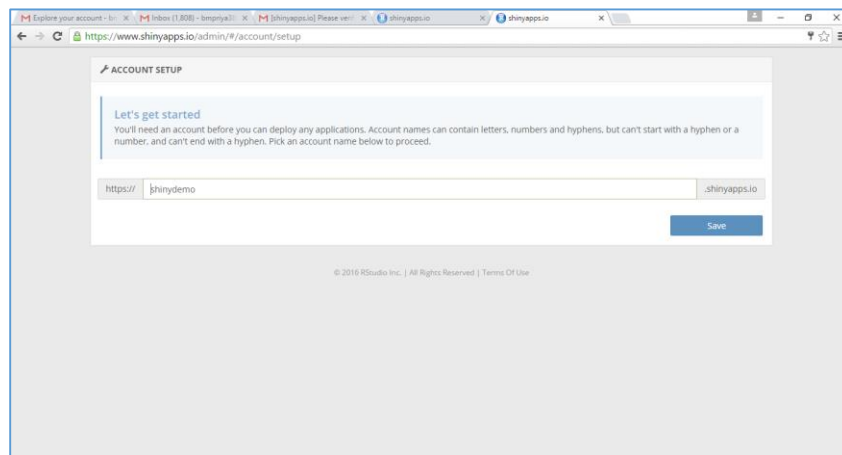
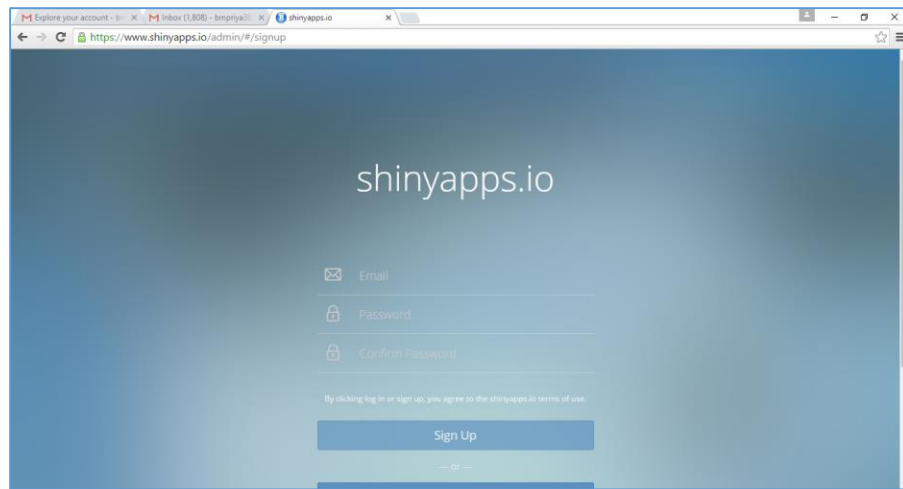
This file contains code for extracting data from an excel sheet titled 'soa_Summary_Tables.xls' and making an RData file containing this data. This file is later accessed to by the functions.R file to utilize the termination rates.

10. Winklevoss_import.R

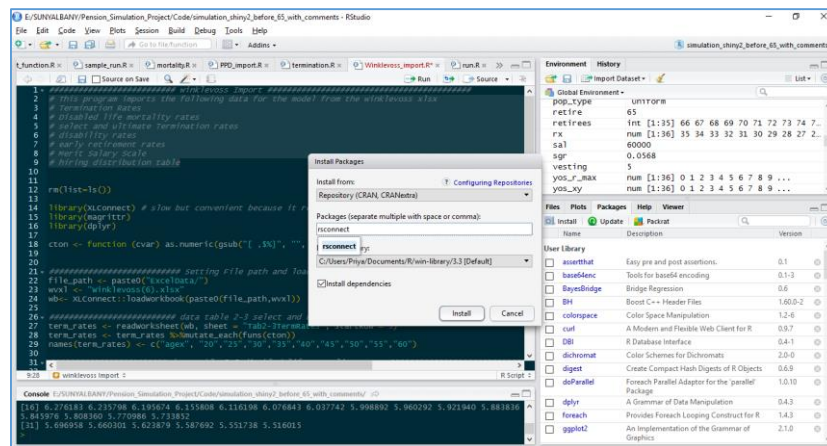
This program imports termination rates, disabled life mortality rates, select and ultimate termination rates, disability rates, early retirement rates, merit salary scale, hiring distribution table data for the model from the winklevoss.xlsx file.

4 ShinyApps.io setup and deployment

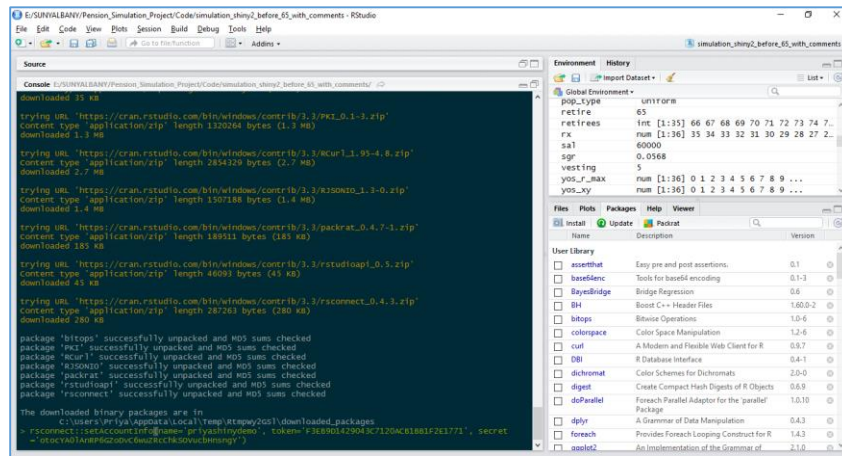
1. Create an account at <https://www.shinyapps.io/> as shown below



2. Install rconnect package in RStudio



- Run the authorize account command in the rstudio console



- Load rsconnect library in the console and deploy application as shown below

```

> rsconnect::deployApp('E:/SUNYALBANY/Pension_Simulation_Project/code/simulation_shiny2_before_65_with_comments')
Preparing to deploy application...
update application currently deployed at
https://priyashinydemo.shinyapps.io/simulation_shiny2_before_65_with_comments/? [Y/n]
DONE
uploading bundle for application: 105989...
Detecting system locale ... en_US
DONE
Deploying bundle: 469120 for application: 105989 ...
waiting for task: 186931623
  building: Parsing manifest
##### Begin Task Log #####
##### End Task Log #####

```

- The application will be deployed at <https://<accountname>.shinyapps.io/<projectname>/>