Reviewing Python and R basics

Intro to Advanced Data Science - DSC 205 - Lyon College Spring'24

Marcus Birkenkrahe (pledged)

Time-stamp: *<2024-02-09 Fri 07:23>*

Contents

1 Bonus: Math plots with R

- This section is prompted by a Calculus II exercise that I came across. Especially when dealing with trigonometric functions, it can be useful to plot them and get a visual on the problem.

- Plot a complicated function, e.g. $f(x) = sin^2(x)sin(2x)$:

  1. Base R (`package:base`) has trigonometric functions pre-loaded. Check `?sin` on the command line to see documentation and function names.
  2. To plot, all you need is the `plot` function.
  3. In Emacs, you can change the code block header arguments. The following metadata will store the result in a file `graph.png` and it will also link to the file in the `#+RESULTS` output. If you don't do this, the plot will appear in a separate window.

     ```
     #+begin_src R :results output graphics file :file graph.png
     ```

- Here's the code that opens the graph in a separate window:

```
## function definition
f <- function(x) {
  return (sin(x)**2 * sin(2*x))
}
```

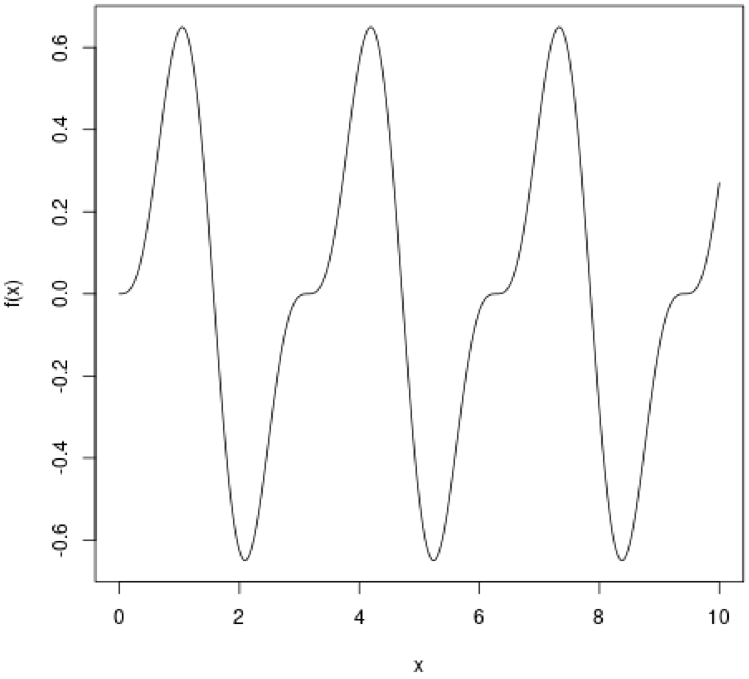
```
## arguments (independent variable)
x = seq(from = 0,to = 10,by = 0.01)

## plot as line plot
plot(x,f(x), type="l")
```

- Here's the code with the altered header line and with some customization:

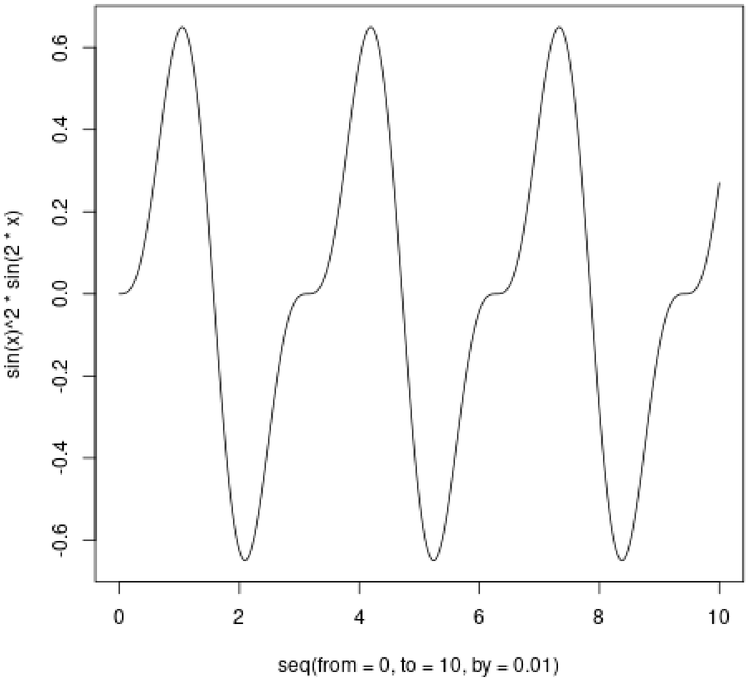
  ```
  ## function definition
  f <- function(x) {
    return (sin(x)**2 * sin(2*x))
  }
  ## arguments (independent variable)
  x = seq(from = 0,to = 10,by = 0.01)

  ## plot as line plot
  plot(x,f(x), type="l")
  ```



- Here is the minimal code without function, in one line:

```
plot(x=seq(from = 0,to = 10,by = 0.01),
     sin(x)**2 * sin(2*x),
     type="l")
```


sin(x)^2 * sin(2 * x)
-0.6
-0.4
-0.2
0.0
0.2
0.4
0.6
0
2
4
6
8
10
seq(from = 0, to = 10, by = 0.01)