Empirical Research (with R)

Based On True Stories

Marco Kühne

2021-08-09

Contents

Dust and Dark

A dusty lecture hall. The light cut through the darkness from the left side of the room. A dozen of seats in each bench, only few occupied by small groups of students who were trying to make sure that they sit far from each other and as far as possible from the lecturer. The bearish but competent assistant professor explained how to analyze and evaluate the results of various memory and cognition experiments through boxplots, t-test and the like in that software. My creaky, slow but loyal laptop in front of me. That's where R was introduced in my psychology undergraduate studies.

Chapter 1

Introduction

Hello World

Second hi

https://medium.com/@delucmat/how-to-publish-bookdown-projects-with-github-actions-on-github-pages-6e6aecc7331e

I need a _book directory:

The .html files (which were compiled from the .Rmd files) are all stored within the _book directory which basically serves as a static website.

Chapter 2

Literature

Here is a review of existing methods.

https://mhweber.github.io/2020/02/12/trying-out-bookdown-and-publishing-to-github-pages/

Chapter 3

Methods

We describe our methods in this chapter.

Chapter 4

Slope Fields

4.1 Definition

> Slope fields (also called direction fields) are a graphical representation of the solutions to a first-order differential equation of a scalar function. Solutions to a slope field are functions drawn as solid curves. A slope field shows the slope of a differential equation at certain vertical and horizontal intervals on the x-y plane, and can be used to determine the approximate tangent slope at a point on a curve, where the curve is some solution to the differential equation.
>
> Wikipedia contributors. (2021, May 17). Slope field. In Wikipedia, The Free Encyclopedia. Retrieved 05:51, June 16, 2021, from https://en.wikipedia.org/w/index.php?title=Slope_field&oldid=1023635282

4.2 Slope Fields in R

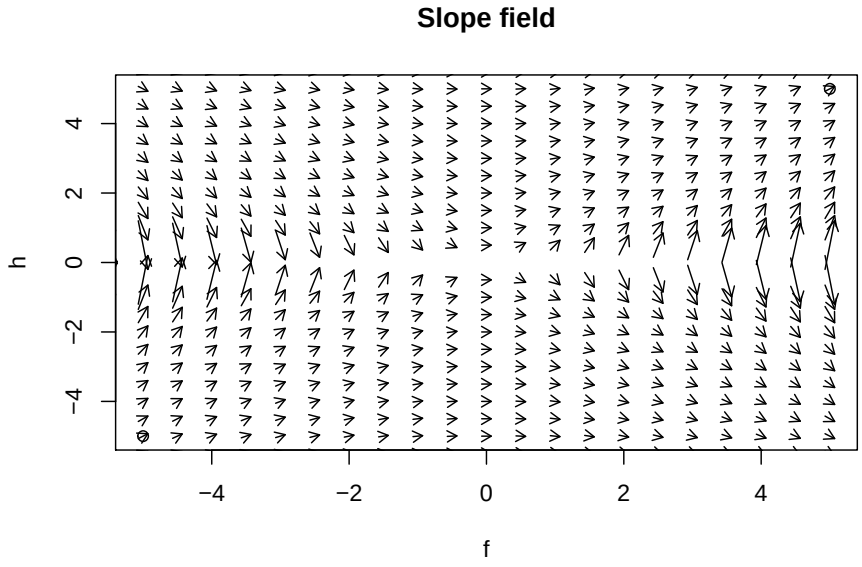
There's no need to build everything from scratch. The first useful resource I can find is:

https://www.r-bloggers.com/2014/09/generate-slope-fields-in-r-and-python/

Show me the code.

```
# Our differential equation
diff <- function(x,y)
{
  return(x/y) #Try also x+y
}
```

```
# Line function
TheLine <- function(x1,y1,slp,d)
{
  z = slope*(d-x1)+y1
  return(z)
}

# Domains
x = seq(-20,20,0.5)
y = seq(-20,20,0.5)

# Points to draw our graph
f = c(-5,5)
h = c(-5,5)
plot(f,h,main="Slope field")


# Let's generate the slope field
for(j in x)
{
  for(k in y)
  {
    slope = diff(j,k)
    domain = seq(j-0.07,j+0.07,0.14)
    z = TheLine(j,k,slope,domain)
    arrows(domain[1],z[1],domain[2],z[2],length=0.08)
  }
}
```



The post https://stackoverflow.com/questions/47984874/how-to-create-a-slope-field-in-r puts this into a `SlopeField` function.

Show me the code.

```
SlopeField = function(FUN,xi = -5,xs = 5,yi = -5,ys = 5, radius = 0.1, grid.by = 0.25){
  # FUN   - given function ODE i.e:
  # xi,xs - lower and upper bound - x - plot
  # yi,ys - lower and upper bound - y - plot

  # grid points
  seqx = seq(xi,xs,grid.by)
  seqy = seq(yi,ys,grid.by)

  # plot
  f = c(xi,xs)
  h = c(yi,ys)
  plot(f,h,main="Slope field", ylab = "Dependet variable", xlab = "Independet variable", pch = ".

  # arrows

  for(x in seqx){
    for(y in seqy){
      ym = y
      xm = x
```

```
      slope = unlist(FUN(x,y))

      if(is.na(slope)){
        cor = "black"
      } else if(slope > 0){
        cor = "blue"
      }else if (slope < 0) {
        cor = "red"
      }else if(slope == 0) {
        cor = "green"
      }
      arrows(radius*cos(atan(slope)+pi)+xm,
             radius*sin(atan(slope)+pi)+ym,
             radius*cos(atan(slope))+xm,
             radius*sin(atan(slope))+ym,
             length = 0.2*radius, col= cor)
    }
  }
}
```

4.3 Technical Example

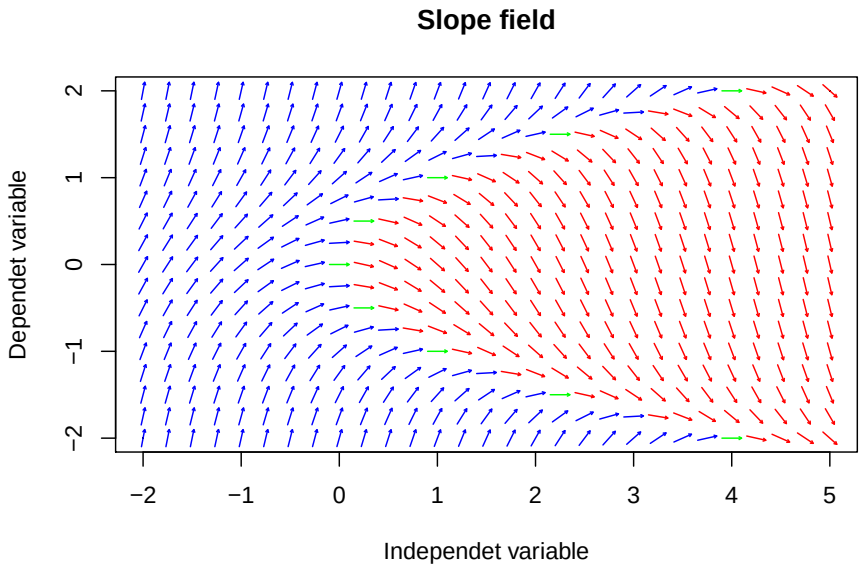
We can specify an ODE in another function and plot its slope field. The suggested example is

$$y'(t) = y^2 - t$$

```
ode = function(t, y){
  dydt <- y^2-t
  list(dydt)
}
```

Let's draw the slope field.

```
SlopeField(ode, xi = -2, xs = 5, yi = -2, ys = 2,radius = 0.1, grid.by = 0.25)
```



That looks good. But can how can we get the ODE solution from this graph?

4.4 Understandable Examples

4.4.1 y' = y

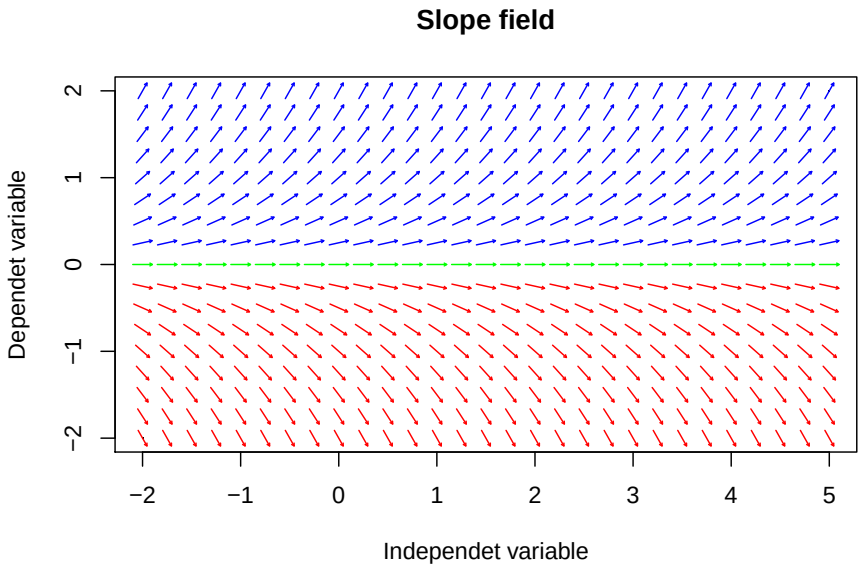
Even without a background in differential equations, think about the following:

$$y'(x) = y(x)$$

We are looking for a function $y(x)$ that is identical to its first derivative $y'(x)$. Perhaps you remember such a function from your last math class (analysis). It is the exponential function, that basically does not change by differentiation.

$$y(x) = e^x \Rightarrow y'(x) = e^x$$

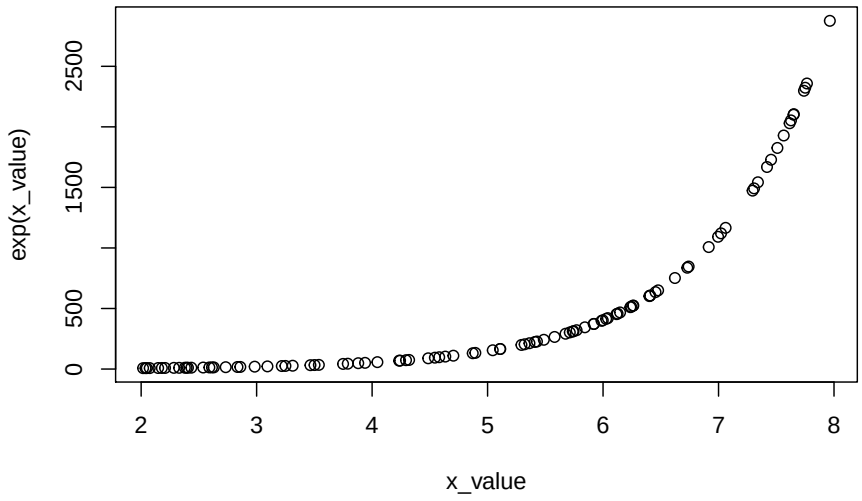
```
ode_1 = function(t, y){
  dydt <- y
  list(dydt)
}

SlopeField(ode_1, xi = -2, xs = 5, yi = -2, ys = 2,radius = 0.1, grid.by = 0.25)
```



Can you see something like the exponential function in this slope field?

Show me how an exponential function looks like.

```
x_value<-runif(100,2,8)
plot(x_value,exp(x_value))
```



Perhaps you can spot something like the $y(x) = e^x$ in the blue area. But how about the red arrows? Any mirror at the x-axis is also a valid solution to the problem. How do we mirror? Use a factor in front of the e function. This may be negative.

$$y(x) = C \cdot e^x \Rightarrow y'(x) = C \cdot e^x$$

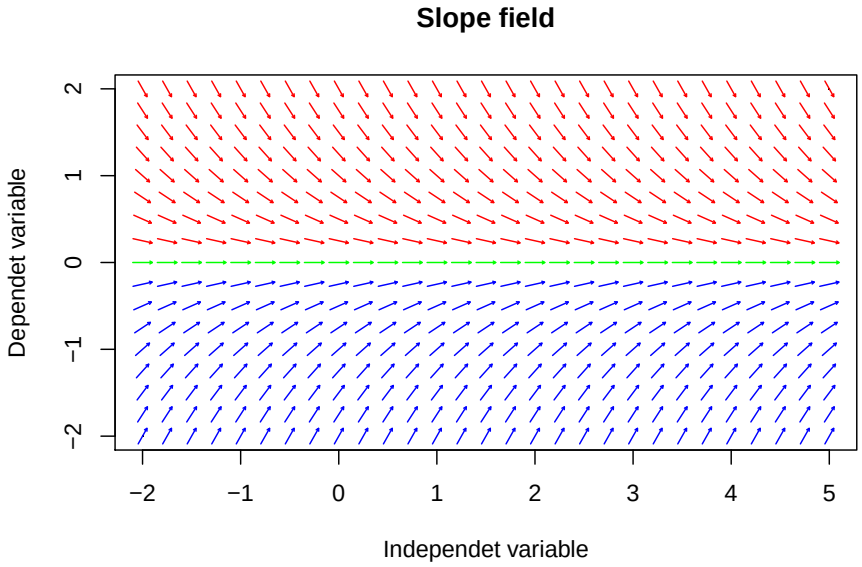
4.4.2 y' = -y

Let's give it another try. Which function equals its first derivative (after changing sign).

$$y'(x) = -y(x)$$

It is a sibling of the e^x. The solution is $y(x) = C \cdot e^{-x}$.

```
ode_2 = function(t, y){
  dydt <- -y
  list(dydt)
}

SlopeField(ode_2, xi = -2, xs = 5, yi = -2, ys = 2,radius = 0.1, grid.by = 0.25)
```



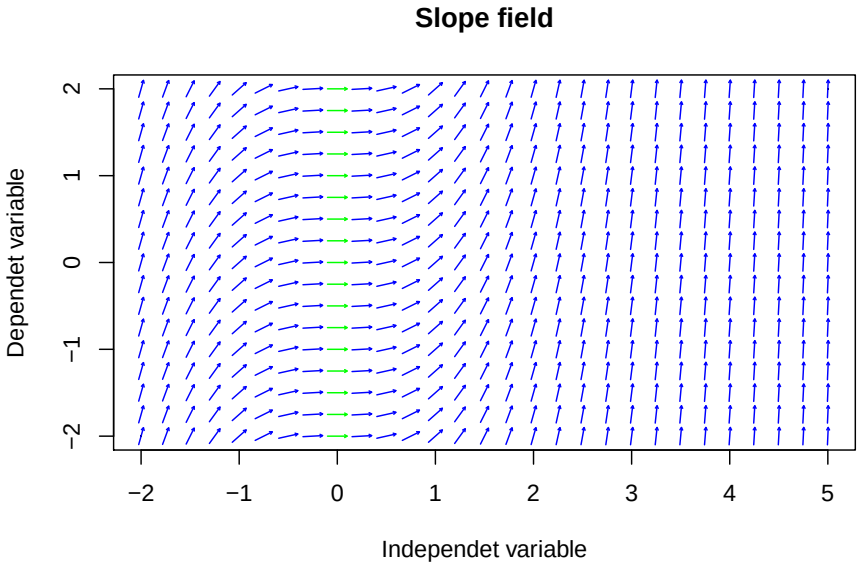
4.4.3 y' = x^2

Wait, there's no more y on the RHS. Don't worry. That'll make it even easier. What is the integral of x^2? You may expect

$$y'(x) = x^2 \Rightarrow \int x^2 dx = 1/3 \cdot x^3 + C$$

Okay, the general shape is a cubic function. Can you spot a cubic function?

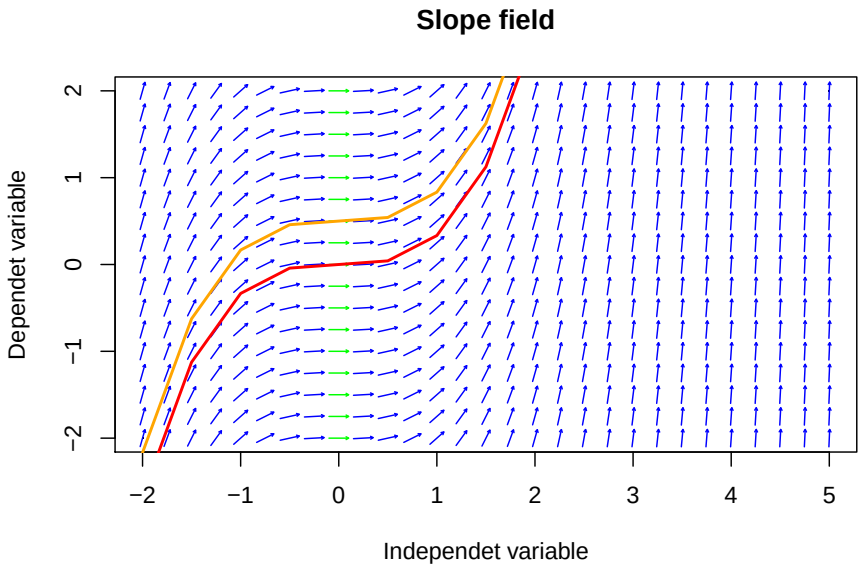
```
ode_3 = function(t, y){
  dydt <- t^2
  list(dydt)
}

SlopeField(ode_3, xi = -2, xs = 5, yi = -2, ys = 2,radius = 0.1, grid.by = 0.25)
```



4.5 Initial value problem

Recap, the prior solution was $y(x) = 1/3 \cdot x^3 + C$. Due to C this is a family of curves. If we specify some of them, that's called the initial value problem. This *problem* makes is actually even more easy to understand the slope field. Let's add two particular cubic functions for $C = 0$ and $C = 0.5$.

```
SlopeField(ode_3, xi = -2, xs = 5, yi = -2, ys = 2,radius = 0.1, grid.by = 0.25)
lines(y,1/3*x*x*x,col="red", lwd=2)
lines(y,1/3*x*x*x+0.5,col="orange", lwd=2)
```



Enjoy. Learn. Share.

Chapter 5

Git and Github

5.1 Install Github and Git

- Get RStudio https://www.rstudio.com/products/rstudio/download/
- Get Github account https://github.com/
- Set up Github pages https://pages.github.com/
- Download Git https://git-scm.com/

Create a "special repository" for GitHub Pages. Only this can turn the branch into docs (1 for free).

- Repository: https://github.com/MarcoKuehne/marcokuehne.github.io
- Homepage: https://marcokuehne.github.io/

Create (or re-open) your R project inside RStudio.

In R console use `usethis::use_git()` which always gives three different answers in random order. Confirm accordingly.

5.2 Working with Git

Use the terminal inside RStudio. Copy paste into terminal with: `CTRL + SHIFT + V`.

5.2.1 git version

Let's start with commands that cannot do any harm (it's 2.32 in 08/2021):

```
git version
```

5.2.2 git config

First, configure github user information on your system. Use the R builtin terminal:

```
git config --global user.email MAIL
git config --global user.name NAME
```

Now, you can commit (upload) changes from RStudio to Github Pages. Check out `git config --list`.

5.2.3 git init

Use `git init` to initialize the repository. It is used to create a new empty repository or directory consisting of files' with the hidden directory. '.git' is created at the top level of your project, which places all of the revision information in one place.

Show what is connected with:

```
git remote -v show
```

5.2.4 git status and diff

Shows you which files are in this staging area, and which files have changes that haven't yet been put there. In order to compare the file as it currently is to what you last saved, you can use `git diff filename`, e.g. `git diff README.md` in the terminal. `git diff` without any file names will show you all the changes in your repository, while git diff directory will show you the changes to the files in some directory.

5.2.5 git commit

To save the changes in the staging area, you use the command `git commit`. It always saves everything that is in the staging area as one unit: as you will see later, when you want to undo changes to a project, you undo all of a commit or none of it.

Commit requires a message (comment). How to write a good git commit mess age:l

https://chris.beams.io/posts/git-commit/

5.2.6 git add

...

5.2.7 git remote add

I would like to have something like `git remote add ...` and `git push ...`, not working.

I can add and remove origins, don't know what it means: `git remote rm origin`

Use git add . in your bash to add all the files to the given folder.

Use git status in your bash to view all the files which are going to be staged to the first commit.

Create **git remote add**

git remote add origin https://github.com/MarcoKuehne/marcokuehne.github.git git remote add origin https://github.com/MarcoKuehne/marcokuehne.github.io git remote add origin https://github.com/MarcoKuehne/marcokuehne.github.io.git

git remote add git@github.com:/.git git remote add git@github.com:MarcoKuehne/marcokuehne.github.git git remote add git@github.com:MarcoKuehne/marcokuehne.github.io does not appear to be a repository git remote add git@github.com:MarcoKuehne/marcokuehne.github.io.git combine both

5.2.8 git push

git push -u origin main git push -u origin master git push origin master git push origin main git push –set-upstream origin main git push -f origin main # worked somehow!!!

If positive, it asks for github credentials.

https://www.datacamp.com/community/tutorials/git-push-pull

5.3 Basic Routines

5.3.1 Add + Commit + Push

I make changes to any one of my markdown files, e.g. the `README.md` (this can also be done via GitHub web interface) on my local machine. After notoriously saving this on my local machine (STRG+S) it appears in the *stage area* in the right upper panel in RStudio. Commit and push can be either done in RStudio or via terminal line (also in RStudio).

First, let's do it via "clicking". Select the staged file. Click "commit". A new windows opens and displays the changes in your file. Write a new commit message of amend a previous commit. Click "close". From this additional window (or the right upper panel in RStudio) click the green push arrow. Hopefully it

will say "HEAD -> main". Close. Done. The result should be available on your github repository immediately.

The same procedure via the terminal:

```
git add 05-git.Rmd
git commit -m "Updated the git short tutorial"
git push
```

How to push? Easy …

I need to `pull` these changes into RStudio. Check the right upper panel "Diff". Select `pull`. You can also find big blue down and green up arrows.

I add this sentence. I save on RStudio, thus it appears as a change to `README.md` in the Git panel. I select this file and green up arrow (push). I enter my credentials and close (see https://docs.github.com/en/get-started/getting-started-with-git/why-is-git-always-asking-for-my-password). Here you might use a personal access token. I click `commit`, enter a commit message, click `commit` again and close the extra window.

5.4 Resources

You can start reading about bookdowns from the inventor:

https://bookdown.org/yihui/bookdown/

This is pretty advanced and I can't understand a tiny piece of it.

Another option is to start a repo and book by copying "awesome book" from another repo.

https://jules32.github.io/bookdown-tutorial/setup.html

This tutorial worked to push a book to a standard repo. But there were problem with the doc branch.

After I forgot how to start my project or where to find it, I checked:

https://happygitwithr.com/rstudio-git-github.html#make-local-changes-save-commit

started a new project with my repo link and a new session. This forks or fetches or pulls or downloads the repo. Now I will try to re-upload the minimal change from today.

Another personal description can be found here:

https://rachaellappan.github.io/bookdown/

`This enourmously helped with the understanding of commits and pushs on git:`

`<http://www.differencebetween.net/technology/difference-between-commit-and-push/>`

- `git commit -m "Started book"`

I followed the video “How to create a bookdown book in 5 minutes”:

https://www.youtube.com/watch?v=m5D-yoH416Y

It did not work.

Explaining terminology:

https://www.notion.so/zarkom/Introduction-to-Git-ac396a0697704709a12b6a0e545db049

Learn Git in 15 Min:

https://www.youtube.com/watch?v=USjZcfj8yxE

Learn GitHub in 20 Min:

https://www.youtube.com/watch?v=nhNq2kIvi9s