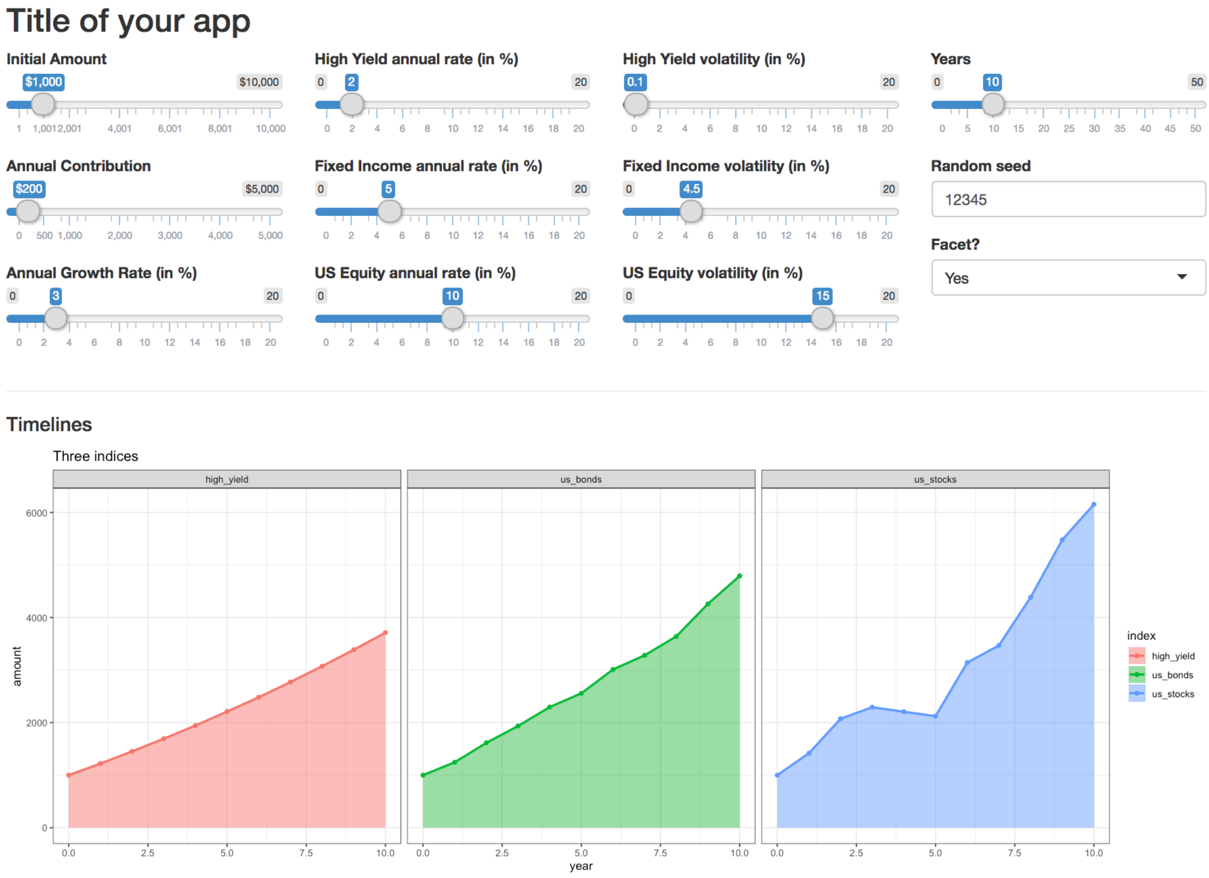
Workout 02: Shiny App

Stat 133, Fall 2019

The purpose of this assignment is to create a shiny app that allows you to visualize—in an interactive way—a couple of simple investing scenarios. Your app should have a similar appearance to the following screenshot:



Title of your app

Initial Amount: $1,000 (range $1 to $10,000)

Annual Contribution: $200 (range $0 to $5,000)

Annual Growth Rate (in %): 3 (range 0 to 20)

High Yield annual rate (in %): 2 (range 0 to 20)

Fixed Income annual rate (in %): 5 (range 0 to 20)

US Equity annual rate (in %): 10 (range 0 to 20)

High Yield volatility (in %): 0.1 (range 0 to 20)

Fixed Income volatility (in %): 4.5 (range 0 to 20)

US Equity volatility (in %): 15 (range 0 to 20)

Years: 10 (range 0 to 50)

Random seed: 12345

Facet?: Yes

Timelines

Three Investing Scenarios

We are going to consider three investing instruments:

- High Yield Savings Account
- U.S. Fixed Income Index Fund (U.S. Bonds)
- U.S. Equity Index Fund (U.S. Stocks)

High Yield Savings Account: A high-yield savings account is a type of savings account that typically pays 20-25 times the national average of a standard savings account. Interest rates as of this writing (Oct 2019) tend to range between 1.8% and 2%. Keep in mind that these rates can change over time (i.e. there's a little bit of volatility). For simulation purposes, let's assume the following rate of return and volatility:

- Average annual return: $\mu = 2\%$
- Volatility (standard deviation): $\sigma = 0.1\%$

U.S. Fixed Income Index Fund (U.S. Bonds): This type of funds have a portfolio formed by government-related issues, corporate bonds, agency mortgage-backed pass-throughs, consumer asset-backed securities, and commercial mortgage-backed securities. In general, fixed income instruments offer conservative investors opportunities to obtain current income at a reasonable amount of risk. For simulation purposes, let's assume the following rate of return and volatility:

- Average annual return: $\mu = 5\%$
- Volatility (standard deviation): $\sigma = 4.5\%$

U.S. Equity Index Fund (U.S. Stocks): This type of funds tend to have a portfolio formed by stocks of American companies across different industries. The most popular funds seek to track the performance of market indices such as S&P 500, Dow Jones, or NASDAQ. For simulation purposes, let's assume the following rate of return and volatility:

- Average annual return: $\mu = 10\%$
- Volatility (standard deviation): $\sigma = 15\%$

Simulation of Returns

Say yo begin with an initial investment of $1000—at the beginning of the year—in one of the three instruments described above. For instance, suppose you put this money in an **index fund of U.S. Bonds** that has an average annual rate of return r. Also, suppose you decide to contribute a growing amount of $200 at the end of every year, growing at an annual fixed rate g. Assume an investment period of 10 years.

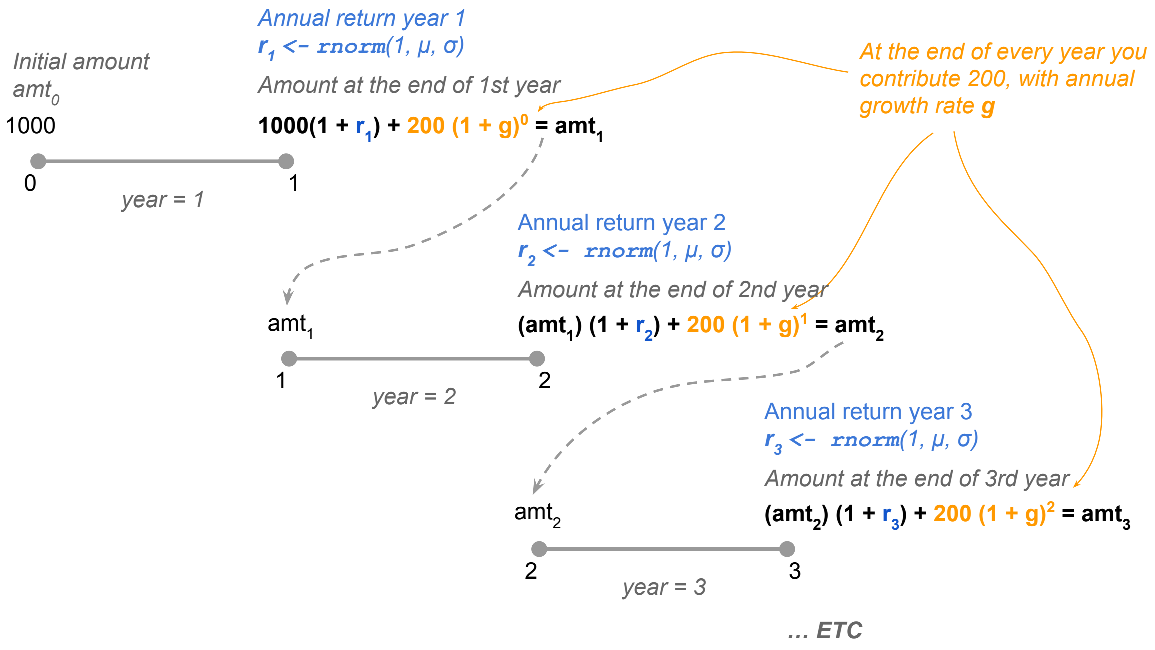
To make things a bit more realistic, we will assume that the the annual rates of return follow a normal distribution:

$$\text{return}_i \sim N(\mu = \text{rate}_i, \ \sigma = \text{volatility}_i)$$

where:

- i is the type of investment (high-yield savings, U.S. Bonds, U.S. stocks)
- rate_i is the provided annual average rate for type i
- volatility_i is the provided annual volatility for type i

Here's a conceptual diagram illustrating the flow of the investment during the first three years:


Initial amount
amt_0
1000
0
year = 1
1
Annual return year 1
r_1 <- rnorm(1, μ, σ)
Amount at the end of 1st year
$1000(1 + r_1) + 200(1 + g)^0 = amt_1$
At the end of every year you contribute 200, with annual growth rate g
amt_1
1
year = 2
2
Annual return year 2
r_2 <- rnorm(1, μ, σ)
Amount at the end of 2nd year
$(amt_1)(1 + r_2) + 200(1 + g)^1 = amt_2$
amt_2
2
year = 3
3
Annual return year 3
r_3 <- rnorm(1, μ, σ)
Amount at the end of 3rd year
$(amt_2)(1 + r_3) + 200(1 + g)^2 = amt_3$
... ETC

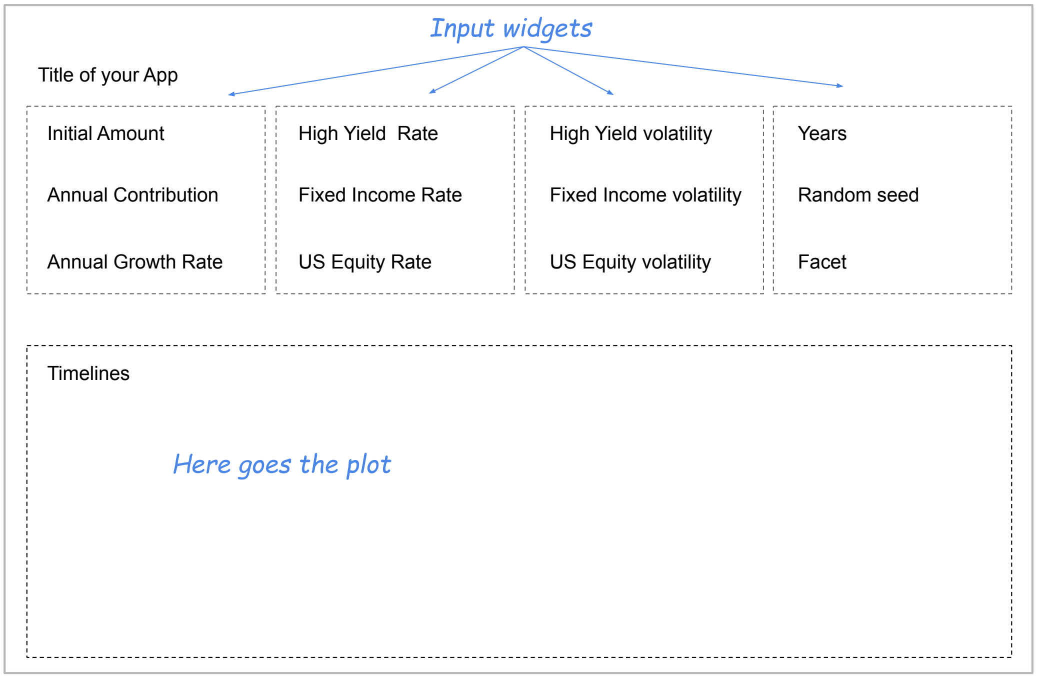
As you can tell from the figure above, an annual rate of return for a given instrument needs to be randomly generated—via `rnorm()`. This is the function in R that allows you to generate random numbers from a Normal distribution. Here's an example that illustrates how to generate a random annual return for an index fund of U.S. Bonds. The provided reference annual rate is `rate_bonds`; the provided volatility is `vol_bonds`. The function `set.seed()` generates a random seed for replication purposes.

```r
# generate a random annual return rate
set.seed(12345)
rate_bonds <- 0.05     # U.S. bonds annual avg return
vol_bonds <- 0.045     # U.S. annual volatility
rnorm(1, mean = rate_bonds, sd = vol_bonds)
```

```
## [1] 0.0763488
```

Shiny App Layout

Your app should have a layout like the following diagram (see specifications below).



As you can tell from the above diagram, the layout of the app involves three distinctive elements:

- **title**: main title for your app (give it a meaningful name)
- **input widgets**: 12 widgets arranged in four columns
- **plot**: an output graph to display the yearly balances

Input widgets

- Slider input for *Initial Amount*, from \$0 to \$10,000, in steps of \$100. Default value of \$1,000.
- Slider input for *Annual Contribution*, from \$0 to \$5,000, in steps of \$100. Default value of \$200.
- Slider input for *Annual Growth Rate* in percentage, from 0% to 20%, in steps of 0.1%. Default value of 2%.
- Slider input for *High Yield rate* in percentage, from 0% to 20%, in steps of 0.1%. Default value of 2%.
- Slider input for *Fixed Income rate* (U.S. Bonds) in percentage, from 0% to 20%, in steps of 0.1%. Default value of 5%.
- Slider input for *US Equity rate* (U.S. Stocks) in percentage, from 0% to 20%, in steps of 0.1%. Default value of 10%.
- Slider input for *High Yield volatility* in percentage, from 0% to 20%, in steps of 0.1%. Default value of 0.1%.
- Slider input for *Fixed Income volatility* (U.S. Bonds) in percentage, from 0% to 20%, in steps of 0.1%. Default value of 4.5%.
- Slider input for *US Equity volatility* (U.S. Stocks) in percentage, from 0% to 20%, in steps of 0.1%. Default value of 15%.
- Slider input for *Years*, from 0 to 50, in steps of 1. Default value of 20.
- Numeric input for *Random Seed*, this is the value to be passed to `set.seed()`. Default value of 12345. To be used for generating new runs of random numebrs.
- Select input for *Facet?*. Choices: “Yes”, and “No”. Default value “Yes”.

Timeline Graphs

Below the input widgets, your app should display a plot for the timeline charts of each savings modality (facetted plot by default).

Resources

You may want to take a look at the Shiny gallery: https://shiny.rstudio.com/gallery/

To learn about how to organize widgets in columns, take a look at the User Interface of the following app:

https://shiny.rstudio.com/gallery/basic-datatable.html

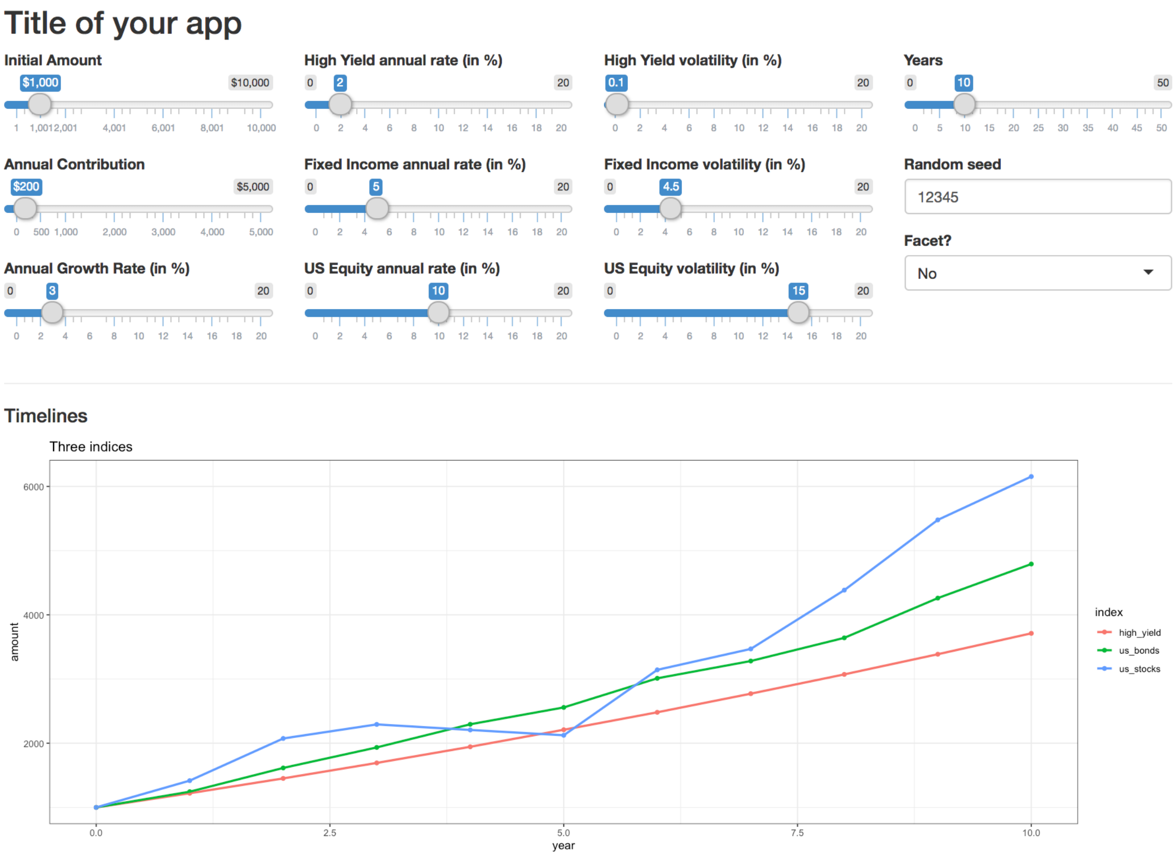
Of course, you can take a look at other apps displayed in the Shiny gallery to get some inspiration.

Facet?

One of the input widgets involves a select option to determine whether the timeline graphs should be facetted or not.

Facet option: No

When the user selects the facet option **No**, then the plot should display a facetted graph with an area below the timeline, like in the following screenshot.



Title of your app

Initial Amount: $1,000

High Yield annual rate (in %): 2

High Yield volatility (in %): 0.1

Years: 10

Annual Contribution: $200

Fixed Income annual rate (in %): 5

Fixed Income volatility (in %): 4.5

Random seed: 12345

Facet?: No

Annual Growth Rate (in %): 3

US Equity annual rate (in %): 10

US Equity volatility (in %): 15

Timelines

Shiny App Code

To develop your shiny app, you can include all the code in the `app.R` file, or alternatively, you can also write functions in a separate `R` script and `source()` into the `app.R` file.

Please use good coding practices:

- Consistent names
- Commented code
- Indented code
- Use of white space
- If you write functions, include roxygen documentation for them

Submission

Upload the file(s) of your shiny app to your private GitHub classroom repository. In theory, the filestructure of your github repo should look like this:

```
workouts/
   README.md
   demo/
   workout1/
      ...   # files of workout1
   workout2/
      README.md
      app.R
      ...   # optional extra files
```

You will have to submit the link of your github repo to bCourses (in the corresponding assignment).