Worksheet 14 - model simulation

Monday, March 24, 2025

DS 002R - Jo Hardin

Name: _____

Names of people you worked with: _____

Thinking about this week's quiz… what is the hardest part of functions and `map()`?

Task: Here are some standard random normal values. You will need to choose 5 of them. Start somewhere and go in any direction (forward, backward, up, down, diagonal, etc.).

```
rnorm(30, mean = 0, sd = 1)
```

```
 [1]  1.99469634  0.71114251  0.18540528 -0.28176501  0.10877555 -1.08573747
 [7] -0.98548216  0.01513086 -0.25204590 -1.46575030 -0.92245624  0.03960243
[13]  0.49382018 -1.82822917  0.09147291  0.67077922 -0.08107805  1.26424109
[19] -0.70338819 -0.04057817 -1.56616208  0.24914817 -0.34041599  0.41719084
[25] -0.32646679 -0.89029402 -1.60815993 -2.32237229 -1.96721918  0.02752681
```

Consider the following function which generates a random investment value for each step. Assume alpha = 0.5.

```
calculate_return <- function(step, alpha) {
  risk_free_rate <- 1.03
  risky_rate <- rnorm(1, mean = 0, sd = 1) * 0.05 + 1
  return(data.frame(step = step,
                    return = (1 - alpha) * risk_free_rate + alpha * risky_rate))
}
```

1. Provide the investment return rate independently for 5 time steps (e.g., 5 years). You should have 5 return rates (return for year 1, return for year 2, return for year 3, etc.).
2. Using the same random normal deviates as in #1, provide the investment return rate cumulatively for 5 time steps (e.g., 5 years). You should have 5 return rates (return for year 1, return for year 1 + 2, return for years 1 + 2 + 3, etc.).

Solution:

```
set.seed(74)
map(1:5, calculate_return, alpha = 0.5) |>
  list_rbind() |>
  mutate(cum_return = cumprod(return))
```

```
  step    return cum_return
1    1 1.0284431   1.028443
2    2 0.9936017   1.021863
3    3 1.0431397   1.065946
4    4 0.9656035   1.029281
5    5 1.0283057   1.058415
```