EDS 230 Assignment 3: Almond Profit Model Sensitivity Analysis

Elmera Azadpour, Mia Forsline, Alex Vand

2022-04-19

Set up

Introduction

1. Develop a profit model for almond yield

```
#read in R script to compute almond profit
#more details about the profit model can be found in the .R file
source(here("functions", "compute_profit_from_almonds.R"))

#read in other necessary R scripts

# 1. read in R script to compute almond yield anomaly
source(here("functions", "almond_yield_anomaly_annual.R"))

# 2. read in R script to compute net present value of almonds
source(here("functions", "compute_npv.R"))
```

Test the `almond_yield_anomaly_annual()` function - save the output as a CSV

```
# read in the data from .txt file
clim_raw <- read.csv(file = here("data", "clim.txt"),
                     header = T,
                     sep = "")

# clean data
clim_data <- clim_raw %>%
             janitor::clean_names() %>%
             mutate(d = lubridate::as_date(d), #convert d column to Date format rather than a character
                    year = lubridate::year(d))

# set coefficient parameter values
temp_coeff1 <- -0.015
temp_coeff2 <- -0.0046
precip_coeff1 <- -0.07
precip_coeff2 <- 0.0043
constant <- 0.28
```

```
# calculate annual almond yield for each water year (wy)
annual_almond_yield <- cbind(unique(clim_data$wy), almond_yield_anomaly_annual(clim_data)) %>%  #create
  as.data.frame() %>% #transform into a data frame
  rename(year = V1,
         yield = V2)

#save output as a CSV in the data folder
write_csv(x = annual_almond_yield, path = here("data", "annual_almond_yield.csv"))
```

Test the `compute_npv()` function

```
npv_test <- round(compute_npv(value = 10, time = 2, discount = 0.12), digits = 2)
```

With a discount rate of 12% over 2 years, a current value of $10 will have a net present value (NPV) of approximately $7.97.

2. Do a simple informal sensitivity analysis of almond yield profit using at least 2 parameters

- Similar to the in-class example, we plan to conduct a sensitivity analysis assuming +/- 15% uncertainty in the current price of almonds (`price`) and discount rate (`discount`)
- We assume a default almond price of $X/ton
- We assume a default discount rate of 0.12
- We begin by sampling X times from a uniform distribution

```
#parameter defaults
almond_price_default <- 3000 #$1.47/lb * 2000lb = approximately $3000/ton
discount_rate_default <- 0.12

#deviation %
deviation = 0.15

#number of samples
nsamples = 300

#sample a uniform distribution from the price default
price <- runif(min = almond_price_default - (deviation * almond_price_default),
                      max = almond_price_default + (deviation * almond_price_default),
                      n=nsamples)

discount = rnorm(mean=0.6, sd = 0.1, n=nsamples)

#bind price_thresh and discount_rate into a dataframe
parameters <- cbind.data.frame(discount, price)
```

```
#read in CSV of output from annual almond yield anomaly function
annual_almond_yield <- read_csv(here("data", "annual_almond_yield.csv"))

#note that parameters column names must match the input parameter names in the compute_profit_from_almo
## testing creation of results by using static values for almond_yield_anomaly and year
```

```r
results <- parameters %>%
  pmap(compute_profit_from_almonds,
       almond_yield_anomaly = 10,
       year = 1980)

#check the results
results[[1]]
```

```
##   scen almond_yield_anomaly year      net   netpre
## 1    1                   10 1980 32540.72 32540.72
```

```r
length(results)
```

```
## [1] 300
```

Can't get this code chunk to run - Mia

```r
# # now we can extract results from the list as above
# mean = map_df(results,`[`, c("mean"))
#
# # and we can add the parameter values for each run so we know what parameters gave us which mean valu
# mean_elect = cbind.data.frame(mean_elect, parms)
```

3. Create a single graph of the results

4. Output the graph as a stand along image