Choice Paper Simulation

Bijesh Mishra, Ph.D.

Table of contents

Techno-economic analysis of agrivoltaic systems in Alabama. A paper for Choice Magazine, AAEA.

1 Setting Up

1.1 Housekeeping

```
rm(list = ls()) # Clean the environment.
options(
  warn=0, # Warnings. options(warn=-1) / options(warn=0)
  scipen=999 # No scientific notations.
  )
```

1.2 Working directory

Codes and output are suppressed. Errors and warnings are visible. No warning and no error means code is working as it should.

1.3 Load libraries

```
library(tidyverse, warn.conflicts = FALSE, quietly = TRUE)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becon
```

```r
library(psych, warn.conflicts = FALSE, quietly = TRUE)
library(likert,  warn.conflicts = FALSE, quietly = TRUE) # Likert Items
library(mice,  warn.conflicts = FALSE, quietly = TRUE)
library(openxlsx2, warn.conflicts = FALSE, quietly = TRUE)
library(ggpubr, warn.conflicts = FALSE, quietly = TRUE) # Scatter plot
library(gmodels,  warn.conflicts = FALSE, quietly = TRUE) # Crosstab
library(reshape2, warn.conflicts = FALSE, quietly = TRUE) # Reshape data
library(pacman,  warn.conflicts = FALSE, quietly = TRUE) # Package Management
library(progress, warn.conflicts = FALSE, quietly = TRUE) #progress bar
library(arrow, warn.conflicts = FALSE, quietly = TRUE) #progress bar
```

```
Some features are not enabled in this build of Arrow. Run `arrow_info()` for more informatio
The repository you retrieved Arrow from did not include all of Arrow's features.
You can install a fully-featured version by running:
`install.packages('arrow', repos = 'https://apache.r-universe.dev')`.
```

```r
pacman::p_loaded()
```

```
 [1] "arrow"     "progress"  "pacman"    "reshape2"  "gmodels"   "ggpubr"
 [7] "openxlsx2" "mice"      "likert"    "xtable"    "psych"     "lubridate"
[13] "forcats"   "stringr"   "dplyr"     "purrr"     "readr"     "tidyr"
[19] "tibble"    "ggplot2"   "tidyverse"
```

1.4 Progress Bar

Tracking data processing progress. Code and results supressed.

1.5 Theme for plots

Setting theme for plots:

```
Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2
3.5.0.
i Please use the `legend.position.inside` argument of `theme()` instead.
```

2 Import data

Import necessary data.

2.1 Tomato

- Yield = Total tomato production (total bucket of 25 lb) from 1 acres of land which varies from 10% to 200% of total production (100%). The range was simulated by multiplying 100% yield by yldvar.
- yldvar = Yield variation parameter ranges from 10% to 200%.
- Rev17 to Rev23 = Revenue for price ranges of \$17 to \$23 per bucket of tomato.
- Total cost = Total cost of production for the given yield.
- rolac17 to rolac23= Return to operator, labor and capital for price range of \$17 to \$23.
- operator Cost = Operator labor cost at \$15/hour for given yield. For 100% yield, total hours = 90.
- rlc17 to 23 = Return to land and capital after subtracting operator cost from total revenue.

```
tomato <- read_xlsx("Data/Parameters.xlsx",
                    sheet = "Tomato",
                    start_row = 2,
                    start_col = 9,
                    skip_empty_rows = TRUE,
                    skip_empty_cols = TRUE,
                    col_names = TRUE) %>%
  rename(yield = Yield,
         yldvar = `Yield Variation (%)`)
dim(tomato)
```

```
[1] 21 25
```

```
str(tomato)
```

```
'data.frame':	21 obs. of  25 variables:
 $ yldvar       : num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield        : num  2720 2584 2448 2312 2176 ...
 $ Rev17        : num  46240 43928 41616 39304 36992 ...
 $ Rev18        : num  48960 46512 44064 41616 39168 ...
 $ Rev19        : num  51680 49096 46512 43928 41344 ...
 $ Rev20        : num  54400 51680 48960 46240 43520 ...
 $ Rev21        : num  57120 54264 51408 48552 45696 ...
 $ Rev22        : num  59840 56848 53856 50864 47872 ...
```

```
 $ Rev23        : num  62560 59432 56304 53176 50048 ...
 $ Total Cost   : num  24561 23863 23165 22467 21769 ...
 $ rolac17      : num  21679 20065 18451 16837 15223 ...
 $ rolac18      : num  24399 22649 20899 19149 17399 ...
 $ rolac19      : num  27119 25233 23347 21461 19575 ...
 $ rolac20      : num  29839 27817 25795 23773 21751 ...
 $ rolac21      : num  32559 30401 28243 26085 23927 ...
 $ rolac22      : num  35279 32985 30691 28397 26103 ...
 $ rolac23      : num  37999 35569 33139 30709 28279 ...
 $ Operator Cost: num  2700 2565 2430 2295 2160 ...
 $ rlc17        : num  18979 17500 16021 14542 13063 ...
 $ rlc18        : num  21699 20084 18469 16854 15239 ...
 $ rlc19        : num  24419 22668 20917 19166 17415 ...
 $ rlc20        : num  27139 25252 23365 21478 19591 ...
 $ rlc21        : num  29859 27836 25813 23790 21767 ...
 $ rlc22        : num  32579 30420 28261 26102 23943 ...
 $ rlc223       : num  35299 33004 30709 28414 26119 ...
```

```r
head(tomato)
```

```
  yldvar yield Rev17 Rev18 Rev19 Rev20 Rev21 Rev22 Rev23 Total Cost  rolac17
3    2.0  2720 46240 48960 51680 54400 57120 59840 62560   24560.62 21679.38
4    1.9  2584 43928 46512 49096 51680 54264 56848 59432   23862.62 20065.38
5    1.8  2448 41616 44064 46512 48960 51408 53856 56304   23164.62 18451.38
6    1.7  2312 39304 41616 43928 46240 48552 50864 53176   22466.62 16837.38
7    1.6  2176 36992 39168 41344 43520 45696 47872 50048   21768.62 15223.38
8    1.5  2040 34680 36720 38760 40800 42840 44880 46920   21070.62 13609.38
   rolac18  rolac19  rolac20  rolac21  rolac22  rolac23 Operator Cost    rlc17
3 24399.38 27119.38 29839.38 32559.38 35279.38 37999.38          2700 18979.38
4 22649.38 25233.38 27817.38 30401.38 32985.38 35569.38          2565 17500.38
5 20899.38 23347.38 25795.38 28243.38 30691.38 33139.38          2430 16021.38
6 19149.38 21461.38 23773.38 26085.38 28397.38 30709.38          2295 14542.38
7 17399.38 19575.38 21751.38 23927.38 26103.38 28279.38          2160 13063.38
8 15649.38 17689.38 19729.38 21769.38 23809.38 25849.38          2025 11584.38
     rlc18    rlc19    rlc20    rlc21    rlc22   rlc223
3 21699.38 24419.38 27139.38 29859.38 32579.38 35299.38
4 20084.38 22668.38 25252.38 27836.38 30420.38 33004.38
5 18469.38 20917.38 23365.38 25813.38 28261.38 30709.38
6 16854.38 19166.38 21478.38 23790.38 26102.38 28414.38
7 15239.38 17415.38 19591.38 21767.38 23943.38 26119.38
8 13624.38 15664.38 17704.38 19744.38 21784.38 23824.38
```

```r
tail(tomato)
```

```
   yldvar yield Rev17 Rev18 Rev19 Rev20 Rev21 Rev22 Rev23 Total Cost    rolac17
18    0.5   680 11560 12240 12920 13600 14280 14960 15640   14090.62  -2530.617
19    0.4   544  9248  9792 10336 10880 11424 11968 12512   13392.62  -4144.617
20    0.3   408  6936  7344  7752  8160  8568  8976  9384   12694.62  -5758.617
21    0.2   272  4624  4896  5168  5440  5712  5984  6256   11996.62  -7372.617
22    0.1   136  2312  2448  2584  2720  2856  2992  3128   11298.62  -8986.617
23    0.0     0     0     0     0     0     0     0     0   10600.62 -10600.617
      rolac18    rolac19     rolac20     rolac21     rolac22     rolac23
18  -1850.617  -1170.617   -490.6174    189.3826    869.3826   1549.3826
19  -3600.617  -3056.617  -2512.6174  -1968.6174  -1424.6174   -880.6174
20  -5350.617  -4942.617  -4534.6174  -4126.6174  -3718.6174  -3310.6174
21  -7100.617  -6828.617  -6556.6174  -6284.6174  -6012.6174  -5740.6174
22  -8850.617  -8714.617  -8578.6174  -8442.6174  -8306.6174  -8170.6174
23 -10600.617 -10600.617 -10600.6174 -10600.6174 -10600.6174 -10600.6174
   Operator Cost      rlc17      rlc18      rlc19      rlc20       rlc21
18           675  -3205.617  -2525.617  -1845.617  -1165.617   -485.6174
19           540  -4684.617  -4140.617  -3596.617  -3052.617  -2508.6174
20           405  -6163.617  -5755.617  -5347.617  -4939.617  -4531.6174
21           270  -7642.617  -7370.617  -7098.617  -6826.617  -6554.6174
22           135  -9121.617  -8985.617  -8849.617  -8713.617  -8577.6174
23             0 -10600.617 -10600.617 -10600.617 -10600.617 -10600.6174
        rlc22      rlc223
18   194.3826    874.3826
19  -1964.6174  -1420.6174
20  -4123.6174  -3715.6174
21  -6282.6174  -6010.6174
22  -8441.6174  -8305.6174
23 -10600.6174 -10600.6174
```

2.2 Strawberry

- Everything same as tomato.
- Numbers 3 to 9 in names are price ranges for strawberry.

```r
strawberry <- read_xlsx("Data/Parameters.xlsx",
                        sheet = "Strawberry",
                        start_row = 2,
                        start_col = 7,
```

```r
                         skip_empty_rows = TRUE,
                         skip_empty_cols = TRUE,
                         col_names = TRUE) %>%
  rename(yield = Yield,
         yldvar = `Yield Variation (%)`)
dim(strawberry)
```

```
[1] 21 25
```

```r
str(strawberry)
```

```
'data.frame':   21 obs. of  25 variables:
 $ yldvar       : num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield        : num  6150 5842 5535 5228 4920 ...
 $ Rev3         : num  18450 17528 16605 15682 14760 ...
 $ Rev4         : num  24600 23370 22140 20910 19680 ...
 $ Rev5         : num  30750 29212 27675 26138 24600 ...
 $ Rev6         : num  36900 35055 33210 31365 29520 ...
 $ Rev7         : num  43050 40898 38745 36592 34440 ...
 $ Rev8         : num  49200 46740 44280 41820 39360 ...
 $ Rev9         : num  55350 52582 49815 47048 44280 ...
 $ Total Cost   : num  20190 19845 19499 19154 18808 ...
 $ rolac3       : num  -1740 -2317 -2894 -3471 -4048 ...
 $ rolac4       : num  4410 3525 2641 1756 872 ...
 $ rolac5       : num  10560 9368 8176 6984 5792 ...
 $ rolac6       : num  16710 15210 13711 12211 10712 ...
 $ rolac7       : num  22860 21053 19246 17439 15632 ...
 $ rolac8       : num  29010 26895 24781 22666 20552 ...
 $ rolac9       : num  35160 32738 30316 27894 25472 ...
 $ Operator Cost: num  2700 2565 2430 2295 2160 ...
 $ rlc3         : num  -4440 -4882 -5324 -5766 -6208 ...
 $ rlc4         : num  1710 960 211 -539 -1288 ...
 $ rlc5         : num  7860 6803 5746 4689 3632 ...
 $ rlc6         : num  14010 12645 11281 9916 8552 ...
 $ rlc7         : num  20160 18488 16816 15144 13472 ...
 $ rlc8         : num  26310 24330 22351 20371 18392 ...
 $ rlc9         : num  32460 30173 27886 25599 23312 ...
```

```r
head(strawberry)
```

```
  yldvar  yield    Rev3  Rev4    Rev5  Rev6    Rev7  Rev8    Rev9 Total Cost
3    2.0 6150.0 18450.0 24600 30750.0 36900 43050.0 49200 55350.0   20190.49
4    1.9 5842.5 17527.5 23370 29212.5 35055 40897.5 46740 52582.5   19844.85
5    1.8 5535.0 16605.0 22140 27675.0 33210 38745.0 44280 49815.0   19499.20
6    1.7 5227.5 15682.5 20910 26137.5 31365 36592.5 41820 47047.5   19153.56
7    1.6 4920.0 14760.0 19680 24600.0 29520 34440.0 39360 44280.0   18807.91
8    1.5 4612.5 13837.5 18450 23062.5 27675 32287.5 36900 41512.5   18462.27
     rolac3     rolac4    rolac5   rolac6   rolac7   rolac8   rolac9
3 -1740.495 4409.50503 10559.505 16709.51 22859.51 29009.51 35159.51
4 -2317.350 3525.15003  9367.650 15210.15 21052.65 26895.15 32737.65
5 -2894.205 2640.79503  8175.795 13710.80 19245.80 24780.80 30315.80
6 -3471.060 1756.44003  6983.940 12211.44 17438.94 22666.44 27893.94
7 -4047.915  872.08503  5792.085 10712.09 15632.09 20552.09 25472.09
8 -4624.770  -12.26997  4600.230  9212.73 13825.23 18437.73 23050.23
  Operator Cost      rlc3      rlc4     rlc5      rlc6     rlc7     rlc8
3          2700 -4440.495  1709.505 7859.505 14009.505 20159.51 26309.51
4          2565 -4882.350   960.150 6802.650 12645.150 18487.65 24330.15
5          2430 -5324.205   210.795 5745.795 11280.795 16815.80 22350.80
6          2295 -5766.060  -538.560 4688.940  9916.440 15143.94 20371.44
7          2160 -6207.915 -1287.915 3632.085  8552.085 13472.09 18392.09
8          2025 -6649.770 -2037.270 2575.230  7187.730 11800.23 16412.73
      rlc9
3 32459.51
4 30172.65
5 27885.80
6 25598.94
7 23312.09
8 21025.23
```

```r
tail(strawberry)
```

```
   yldvar  yield   Rev3 Rev4   Rev5 Rev6    Rev7  Rev8    Rev9 Total Cost
18    0.5 1537.5 4612.5 6150 7687.5 9225 10762.5 12300 13837.5   15005.82
19    0.4 1230.0 3690.0 4920 6150.0 7380  8610.0  9840 11070.0   14660.17
20    0.3  922.5 2767.5 3690 4612.5 5535  6457.5  7380  8302.5   14314.53
21    0.2  615.0 1845.0 2460 3075.0 3690  4305.0  4920  5535.0   13968.88
22    0.1  307.5  922.5 1230 1537.5 1845  2152.5  2460  2767.5   13623.24
23    0.0    0.0    0.0    0    0.0    0     0.0     0     0.0   13277.59
      rolac3     rolac4     rolac5     rolac6     rolac7     rolac8     rolac9
18 -10393.32  -8855.820  -7318.320  -5780.820  -4243.320  -2705.820  -1168.320
19 -10970.17  -9740.175  -8510.175  -7280.175  -6050.175  -4820.175  -3590.175
20 -11547.03 -10624.530  -9702.030  -8779.530  -7857.030  -6934.530  -6012.030
```

```
21 -12123.88 -11508.885 -10893.885 -10278.885  -9663.885  -9048.885  -8433.885
22 -12700.74 -12393.240 -12085.740 -11778.240 -11470.740 -11163.240 -10855.740
23 -13277.59 -13277.595 -13277.595 -13277.595 -13277.595 -13277.595 -13277.595
   Operator Cost      rlc3      rlc4       rlc5       rlc6       rlc7
18           675 -11068.32  -9530.82  -7993.320  -6455.820  -4918.320
19           540 -11510.17 -10280.17  -9050.175  -7820.175  -6590.175
20           405 -11952.03 -11029.53 -10107.030  -9184.530  -8262.030
21           270 -12393.88 -11778.88 -11163.885 -10548.885  -9933.885
22           135 -12835.74 -12528.24 -12220.740 -11913.240 -11605.740
23             0 -13277.59 -13277.59 -13277.595 -13277.595 -13277.595
         rlc8       rlc9
18  -3380.820  -1843.320
19  -5360.175  -4130.175
20  -7339.530  -6417.030
21  -9318.885  -8703.885
22 -11298.240 -10990.740
23 -13277.595 -13277.595
```

2.3 Squash

- Everything same as tomato and strawberry.
- Numbers 11 to 17 in names are price ranges for squash.

```
squash <- read_xlsx("Data/Parameters.xlsx",
                    sheet = "Squash",
                    start_row = 2,
                    start_col = 8,
                    skip_empty_rows = TRUE,
                    skip_empty_cols = TRUE,
                    col_names = TRUE) %>%
  rename(yield = Yield,
         yldvar = `Yield Variation (%)`)
dim(squash)
```

```
[1] 21 25
```

```
head(squash)
```

```
  yldvar yield Rev11 Rev12 Rev13 Rev14 Rev15 Rev16 Rev17 Total Cost   rolac11
3    2.0  2180 23980 26160 28340 30520 32700 34880 37060   13670.88 10309.117
```

```
4    1.9  2071 22781 24852 26923 28994 31065 33136 35207   13173.63  9607.367
5    1.8  1962 21582 23544 25506 27468 29430 31392 33354   12676.38  8905.617
6    1.7  1853 20383 22236 24089 25942 27795 29648 31501   12179.13  8203.867
7    1.6  1744 19184 20928 22672 24416 26160 27904 29648   11681.88  7502.117
8    1.5  1635 17985 19620 21255 22890 24525 26160 27795   11184.63  6800.367
    rolac12  rolac13  rolac14  rolac15  rolac16  rolac17 Operator Cost    rlc11
3 12489.117 14669.12 16849.12 19029.12 21209.12 23389.12          2700 7609.117
4 11678.367 13749.37 15820.37 17891.37 19962.37 22033.37          2565 7042.367
5 10867.617 12829.62 14791.62 16753.62 18715.62 20677.62          2430 6475.617
6 10056.867 11909.87 13762.87 15615.87 17468.87 19321.87          2295 5908.867
7  9246.117 10990.12 12734.12 14478.12 16222.12 17966.12          2160 5342.117
8  8435.367 10070.37 11705.37 13340.37 14975.37 16610.37          2025 4775.367
     rlc12     rlc13     rlc14    rlc15    rlc16    rlc17
3 9789.117 11969.117 14149.117 16329.12 18509.12 20689.12
4 9113.367 11184.367 13255.367 15326.37 17397.37 19468.37
5 8437.617 10399.617 12361.617 14323.62 16285.62 18247.62
6 7761.867  9614.867 11467.867 13320.87 15173.87 17026.87
7 7086.117  8830.117 10574.117 12318.12 14062.12 15806.12
8 6410.367  8045.367  9680.367 11315.37 12950.37 14585.37
```

```r
tail(squash)
```

```
   yldvar yield Rev11 Rev12 Rev13 Rev14 Rev15 Rev16 Rev17 Total Cost   rolac11
18    0.5   545  5995  6540  7085  7630  8175  8720  9265   6212.133  -217.133
19    0.4   436  4796  5232  5668  6104  6540  6976  7412   5714.883  -918.883
20    0.3   327  3597  3924  4251  4578  4905  5232  5559   5217.633 -1620.633
21    0.2   218  2398  2616  2834  3052  3270  3488  3706   4720.383 -2322.383
22    0.1   109  1199  1308  1417  1526  1635  1744  1853   4223.133 -3024.133
23    0.0     0     0     0     0     0     0     0     0   3725.883 -3725.883
     rolac12     rolac13   rolac14   rolac15     rolac16   rolac17
18   327.867   872.86702  1417.867  1962.867  2507.86702  3052.867
19  -482.883   -46.88298   389.117   825.117  1261.11702  1697.117
20 -1293.633  -966.63298  -639.633  -312.633    14.36702   341.367
21 -2104.383 -1886.38298 -1668.383 -1450.383 -1232.38298 -1014.383
22 -2915.133 -2806.13298 -2697.133 -2588.133 -2479.13298 -2370.133
23 -3725.883 -3725.88298 -3725.883 -3725.883 -3725.88298 -3725.883
   Operator Cost     rlc11     rlc12     rlc13     rlc14     rlc15     rlc16
18           675  -892.133  -347.133   197.867   742.867  1287.867  1832.867
19           540 -1458.883 -1022.883  -586.883  -150.883   285.117   721.117
20           405 -2025.633 -1698.633 -1371.633 -1044.633  -717.633  -390.633
21           270 -2592.383 -2374.383 -2156.383 -1938.383 -1720.383 -1502.383
22           135 -3159.133 -3050.133 -2941.133 -2832.133 -2723.133 -2614.133
```

```
23             0 -3725.883 -3725.883 -3725.883 -3725.883 -3725.883 -3725.883
         rlc17
18  2377.86702
19  1157.11702
20   -63.63298
21 -1284.38298
22 -2505.13298
23 -3725.88298
```

2.4 Electricity price

Electricity price ranges from 1 cents to 6 cents in 0.5 cent increment. Previously, I used AL retail electricity price as described below. It's no longer in use but I put description below for the record.

Electricity price ($/kWh) was retail electricity price range for Alabama based on retail electricity price in April 2023 and April 2024 taken from DOE Database. Retail electricity price range in Alabama was from 6.44 to 15.85 cents/kWh in April 2023 and April 2024 which represents industry, commercial, and residential prices.

```r
elec_price <- read_xlsx("Data/Parameters.xlsx",
                              sheet = "Electricity Price") %>%
  rename(epr_kwh = `Electricity Price ($/kWh)`)
dim(elec_price)
```

```
[1] 11  1
```

```r
str(elec_price)
```

```
'data.frame':   11 obs. of  1 variable:
 $ epr_kwh: num  0.01 0.015 0.02 0.025 0.03 0.035 0.04 0.045 0.05 0.055 ...
```

```r
elec_price
```

```
   epr_kwh
2    0.010
3    0.015
4    0.020
5    0.025
6    0.030
```

```
7     0.035
8     0.040
9     0.045
10    0.050
11    0.055
12    0.060
```

2.5 PV system cost

- Data taken from "Capital Costs for Dual-Use Photovoltaic Installations: 2020 Benchmark" Table 1 and Figure 3.
- This data was used to estimate CAPEX.
- avtyps = agrivoltaic types.
- item = itemized component of system.
- cost = cost of each item.
- height = ground to panel clearance height (ft.)
- tcost = Total cost is the sum of all itemized cost for AV system. See figure 3 and table 1 in above document for more detail.

```r
pvsc <- wb_read(file = "Data/Parameters.xlsx",
                sheet = "PV system Cost (NREL)",
                rows = c(1:109),
                cols = c(1:5),
                col_names = TRUE) %>%
  rename(avtyps = `AV Types`,
         item = Item,
         cost = `Cost ($/W)`,
         height = `Panel Height (ft.)`,
         tcost = `Total Cost ($/W)`
         )
dim(pvsc)
```

```
[1] 108   5
```

```r
str(pvsc)
```

```
'data.frame':   108 obs. of  5 variables:
 $ avtyps: chr  "Typical Fixed PV" "Typical Fixed PV" "Typical Fixed PV" "Typical Fixed PV"
 $ item  : chr  "EPC/Developer Net Profit" "Developer Overhead" "Contingency(3%)" "Interconne
 $ cost  : num  0.11 0.15 0.05 0.03 0.02 0.05 0.12 0.18 0.24 0.11 ...
```

```
 $ height: num  4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 ...
 $ tcost : num  1.53 1.53 1.53 1.53 1.53 1.53 1.53 1.53 1.53 1.53 ...
```

```r
head(pvsc)
```

```
            avtyps                      item cost height tcost
2 Typical Fixed PV EPC/Developer Net Profit 0.11    4.6  1.53
3 Typical Fixed PV       Developer Overhead 0.15    4.6  1.53
4 Typical Fixed PV          Contingency(3%) 0.05    4.6  1.53
5 Typical Fixed PV      Interconnection Fee 0.03    4.6  1.53
6 Typical Fixed PV  Permitting Fee (if any) 0.02    4.6  1.53
7 Typical Fixed PV        Sale Tax (if any) 0.05    4.6  1.53
```

```r
tail(pvsc)
```

```
                                     avtyps                           item cost
104 PV + Crops (Reinforced Regular Mount)                   EPC Overhead 0.25
105 PV + Crops (Reinforced Regular Mount) Installation and Labor Cost 0.32
106 PV + Crops (Reinforced Regular Mount)                 Electrical BOS 0.38
107 PV + Crops (Reinforced Regular Mount)                 Structural BOS 0.32
108 PV + Crops (Reinforced Regular Mount)                  Inverter Only 0.08
109 PV + Crops (Reinforced Regular Mount)                         Module 0.40
    height tcost
104    8.2  2.33
105    8.2  2.33
106    8.2  2.33
107    8.2  2.33
108    8.2  2.33
109    8.2  2.33
```

2.6 Capex (NREL)

Variable Descriptions:

- Capex: Capital investment cost ($/W) to develop solar energy system. Capex includes cost of physical structure, developer's overhead and EPC/Developer's net profit.
- capex estimated as f(height, tracker) using OLS for 6.4 ft Tracking system.
- Height = ground to panel clearance in ft.

- array: Solar array. Tracker = Single axis sun tracking panels; Fixed = Non-tracking panels.
- Source: Horowitz, 2020. CAPEX AV.

```
capex <- read.table(file = "Data/CAPEX.txt",
                    header = TRUE,
                   sep = "\t") %>%
  rename(capex = cost,
         height = pheight,
         array = tracker)
dim(capex)
```

```
[1] 6 3
```

```
str(capex)
```

```
'data.frame':   6 obs. of  3 variables:
 $ height: num  4.6 4.6 6.4 8.2 8.2 6.4
 $ capex : num  1.59 1.73 1.85 2.33 2.11 ...
 $ array : chr  "Fixed" "Tracking" "Fixed" "Fixed" ...
```

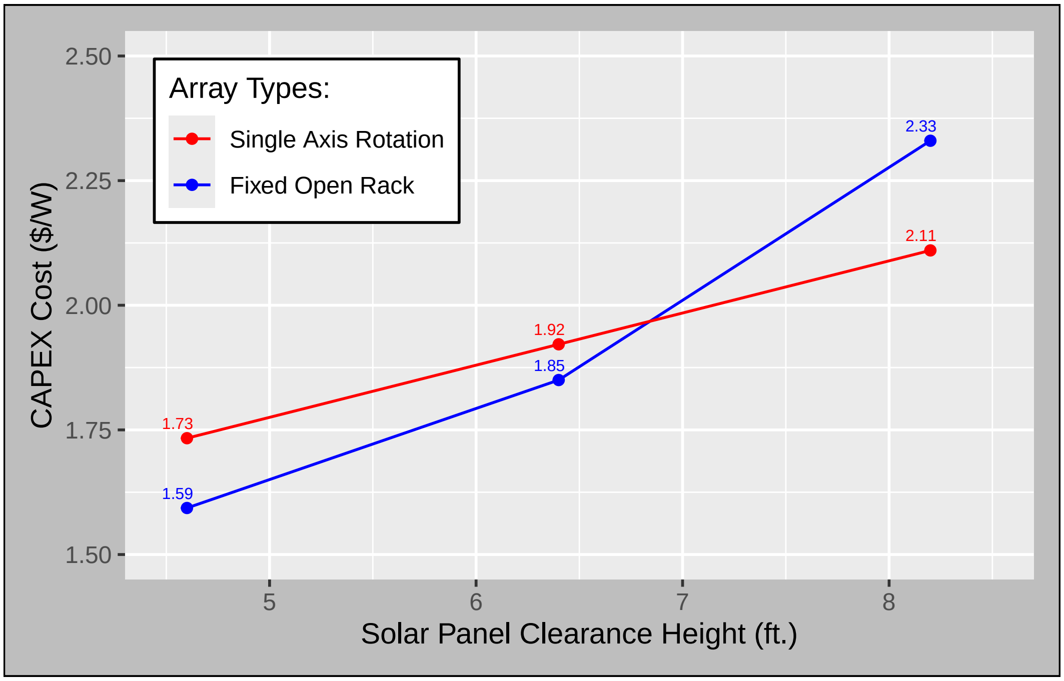
```
capex
```

```
  height    capex    array
1    4.6 1.593333    Fixed
2    4.6 1.733333 Tracking
3    6.4 1.850000    Fixed
4    8.2 2.330000    Fixed
5    8.2 2.110000 Tracking
6    6.4 1.921667 Tracking
```

2.6.1 Plotting capex

```
capex %>%
  ggplot(aes(
    x = height,
    y = capex,
    color = array,
```

```r
  group = array
)) +
geom_point() +
geom_line() +
# Display the rounded capex values
geom_text(aes(label = sprintf("%.2f", capex)),
          vjust = -0.8,
          hjust = 0.8,
          size = 2,
          check_overlap = TRUE,
          show.legend = FALSE
          ) +
labs(
  #title = "CAPEX Cost by Solar Panel Height",
  x = "Solar Panel Clearance Height (ft.)",
  y = "CAPEX Cost ($/W)",
  color = "Array Types:"
  ) +
scale_x_continuous(limits = c(4.5, 8.5)) +
scale_y_continuous(limits = c(1.5, 2.5)) +
guides(color = guide_legend(reverse = TRUE)) +
theme(
  plot.background = element_rect(
    fill = "grey",
    color = "black"
    ),
  legend.position = c(0.2, 0.8),
  legend.background = element_rect(
    fill = "white",
    color = "black"
    ),
  plot.margin = margin(10, 10, 10, 10)
) +
scale_color_manual(
  values = c("Fixed" = "blue",
             "Tracking" = "red"),
  labels = c("Fixed Open Rack",
             "Single Axis Rotation")
)
```




```
# Save the plot
ggsave(
  filename = "Plots/CAPEX Solar Panels.png",
  width = 8,
  height = 6,
  units = "in"
  )
```

2.7 Panel Configuration

- Panel configuration and DV system output (W).

```
panconf <- wb_read(file = "Data/Parameters.xlsx",
                   sheet = "Panel Spacing",
                   start_row = 2,
                   start_col = 1,
                   skip_empty_rows = TRUE,
                   skip_empty_cols = TRUE,
                   col_names = TRUE)
  # rename(avtyps = `AV Types`,
  #        item = Item,
  #        cost = `Cost ($/W)`,
```

```r
  #        height = `Panel Height (ft.)`,
  #        tcost = `Total Cost ($/W)`)
dim(panconf)
```

```
[1] 21 21
```

```r
str(panconf)
```

```
'data.frame':   21 obs. of  21 variables:
 $ Total Area (Acre)              : num  1 1 1 1 1 1 1 1 1 1 ...
 $ Total Area (Sq. Ft.)           : num  43560 43560 43560 43560 43560 ...
 $ Solar Proportion               : num  1 0.95 0.9 0.85 0.8 0.75 0.7 0.65 0.6 0.55 ...
 $ Solar Proportion Area (Sq. Ft.): num  43560 41382 39204 37026 34848 ...
 $ Solar Proportion Area (Sq.M.)  : num  4047 3845 3642 3440 3237 ...
 $ Side Length (ft.)              : num  209 209 209 209 209 ...
 $ YSide Length (ft.)             : num  209 209 209 209 209 ...
 $ XSide length (ft.)             : num  209 198 188 177 167 ...
 $ Panel Length (ft.)             : num  7.75 7.75 7.75 7.75 7.75 7.75 7.75 7.75 7.75 7.75 .
 $ Row Seperator (ft.)            : num  6 6 6 6 6 6 6 6 6 6 ...
 $ Panel Width(ft.)               : num  3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 ...
 $ Panel Area (Sq. ft.)           : num  27.1 27.1 27.1 27.1 27.1 ...
 $ Panels/Row                     : num  59 59 59 59 59 59 59 59 59 59 ...
 $ Total Rows                     : num  15 14 13 12 12 11 10 9 9 8 ...
 $ Total Panels                   : num  885 826 767 708 708 649 590 531 531 472 ...
 $ Array Area (Sq. Ft.)           : num  24006 22405 20805 19204 19204 ...
 $ Array Area (Sq. M.)            : num  2230 2082 1933 1784 1784 ...
 $ XSide Open Length (ft)         : num  92 100 107 115 115 123 131 138 138 146 ...
 $ Inter Panel Spacing (ft)       : num  6 7 8 10 10 12 14 17 17 20 ...
 $ Panel Efficienfy               : num  0.19 0.19 0.19 0.19 0.19 0.19 0.19 0.19 0.19 0.19 .
 $ DC System Size (kW)            : num  424 395 367 339 339 ...
```

```r
head(panconf)
```

```
  Total Area (Acre) Total Area (Sq. Ft.) Solar Proportion
3                 1                43560             1.00
4                 1                43560             0.95
5                 1                43560             0.90
6                 1                43560             0.85
7                 1                43560             0.80
8                 1                43560             0.75
```

	Solar Proportion Area (Sq. Ft.)	Solar Proportion Area (Sq.M.)
3	43560	4046.856
4	41382	3844.513
5	39204	3642.170
6	37026	3439.828
7	34848	3237.485
8	32670	3035.142

	Side Length (ft.)	YSide Length (ft.)	XSide length (ft.)	Panel Length (ft.)
3	208.7103	208.7103	208.7103	7.75
4	208.7103	208.7103	198.2748	7.75
5	208.7103	208.7103	187.8393	7.75
6	208.7103	208.7103	177.4038	7.75
7	208.7103	208.7103	166.9683	7.75
8	208.7103	208.7103	156.5327	7.75

	Row Seperator (ft.)	Panel Width(ft.)	Panel Area (Sq. ft.)	Panels/Row
3	6	3.5	27.125	59
4	6	3.5	27.125	59
5	6	3.5	27.125	59
6	6	3.5	27.125	59
7	6	3.5	27.125	59
8	6	3.5	27.125	59

	Total Rows	Total Panels	Array Area (Sq. Ft.)	Array Area (Sq. M.)
3	15	885	24005.62	2230.195
4	14	826	22405.25	2081.516
5	13	767	20804.88	1932.836
6	12	708	19204.50	1784.156
7	12	708	19204.50	1784.156
8	11	649	17604.12	1635.477

	XSide Open Length (ft)	Inter Panel Spacing (ft)	Panel Efficienfy
3	92	6	0.19
4	100	7	0.19
5	107	8	0.19
6	115	10	0.19
7	115	10	0.19
8	123	12	0.19

	DC System Size (kW)
3	423.7371
4	395.4880
5	367.2388
6	338.9897
7	338.9897
8	310.7405

```r
tail(panconf)
```

```
   Total Area (Acre) Total Area (Sq. Ft.) Solar Proportion
18                 1                43560             0.25
19                 1                43560             0.20
20                 1                43560             0.15
21                 1                43560             0.10
22                 1                43560             0.05
23                 1                43560             0.00
   Solar Proportion Area (Sq. Ft.) Solar Proportion Area (Sq.M.)
18                           10890                     1011.7140
19                            8712                      809.3712
20                            6534                      607.0284
21                            4356                      404.6856
22                            2178                      202.3428
23                               0                        0.0000
   Side Length (ft.) YSide Length (ft.) XSide length (ft.) Panel Length (ft.)
18          208.7103           208.7103           52.17758               7.75
19          208.7103           208.7103           41.74207               7.75
20          208.7103           208.7103           31.30655               7.75
21          208.7103           208.7103           20.87103               7.75
22          208.7103           208.7103           10.43552               7.75
23          208.7103           208.7103            0.00000               7.75
   Row Seperator (ft.) Panel Width(ft.) Panel Area (Sq. ft.) Panels/Row
18                   6              3.5               27.125         59
19                   6              3.5               27.125         59
20                   6              3.5               27.125         59
21                   6              3.5               27.125         59
22                   6              3.5               27.125         59
23                   6              3.5               27.125         59
   Total Rows Total Panels Array Area (Sq. Ft.) Array Area (Sq. M.)
18          3          177             4801.125            446.0391
19          3          177             4801.125            446.0391
20          2          118             3200.750            297.3594
21          1           59             1600.375            148.6797
22          0            0                0.000              0.0000
23          0            0                0.000              0.0000
   XSide Open Length (ft) Inter Panel Spacing (ft) Panel Efficienfy
18                    185                       92             0.19
19                    185                       92             0.19
20                    193                      193             0.19
21                    200                       NA             0.19
```

```
22                    208                          NA             0.19
23                    208                          NA             0.19
   DC System Size (kW)
18            84.74742
19            84.74742
20            56.49828
21            28.24914
22             0.00000
23             0.00000
```

2.8 Energy output

Energy output was simulated using NREL PV Watts Calculator.

- sprop = land proportion covered by solar in 1 acres. Value ranges from 0 to 1.
- Panels = Total number of panels in 1 acres of land.
- datalot: 1 = first simulation done for four regions of AL; 2 = second simulation done for four regions of AL. Two simulations have two unique zipcodes for each simulated region.
- al_regs = regions of Alabama
- zips = zipcodes selected from each region of AL for simulation.
- array = Fixed (open rack); 1AxisRot = 1 Axis Tracking. See above NREL tool for more detail.
- dc_kw = DC system size, calculated for each solar panel heights considering solar panels efficiency and area covered by solar panels.
- energy = total energy output (kWh/Year) considering system parameters. Total hours considered by the model is 8,760 (See PV Watts Calculator Results > help (below the result) > results > download monthly or hourly results).

```
energy_output <- read_xlsx("Data/Parameters.xlsx",
                           sheet = "Energy Output",
                           start_row = 1,
                           start_col = 1,
                           skip_empty_rows = TRUE,
                           skip_empty_cols = TRUE,
                           col_names = TRUE) %>%
  rename(sprop = `Solar Proportion`,
         panels = `Total Panels`,
         datalot = DataLot,
```

```r
         al_regs = `Region of AL`,
         zips = ZIPCODE,
         array = `Array Type`,
         dc_kw = `DC System Size (kW)`,
         energy = `Energy (kWh/Year)`) %>%
  mutate(
    dc_kw = round(dc_kw, 2),
    #energy = energy*1000,  #kWh/Year converted to Wh/Year.
    array = case_when(
      array == "1AxisRot" ~ "Tracking",
      array == "FixedOpen" ~ "Fixed",
      TRUE ~ array)
    )

dim(energy_output)
```

```
[1] 336   8
```

```r
str(energy_output)
```

```
'data.frame':   336 obs. of  8 variables:
 $ sprop  : num  1 1 1 1 1 1 1 1 0.95 0.95 ...
 $ panels : num  885 885 885 885 885 885 885 885 826 826 ...
 $ datalot: num  1 1 1 1 1 1 1 1 1 1 ...
 $ al_regs: chr  "Northern" "Northern" "Central" "Central" ...
 $ zips   : num  35801 35801 35223 35223 36117 ...
 $ array  : chr  "Tracking" "Fixed" "Tracking" "Fixed" ...
 $ dc_kw  : num  424 424 424 424 424 ...
 $ energy : num  672887 585225 668895 579758 728181 ...
```

```r
head(energy_output)
```

```
  sprop panels datalot    al_regs  zips    array  dc_kw energy
2     1    885       1   Northern 35801 Tracking 423.74 672887
3     1    885       1   Northern 35801    Fixed 423.74 585225
4     1    885       1    Central 35223 Tracking 423.74 668895
5     1    885       1    Central 35223    Fixed 423.74 579758
6     1    885       1 Black Belt 36117 Tracking 423.74 728181
7     1    885       1 Black Belt 36117    Fixed 423.74 629523
```

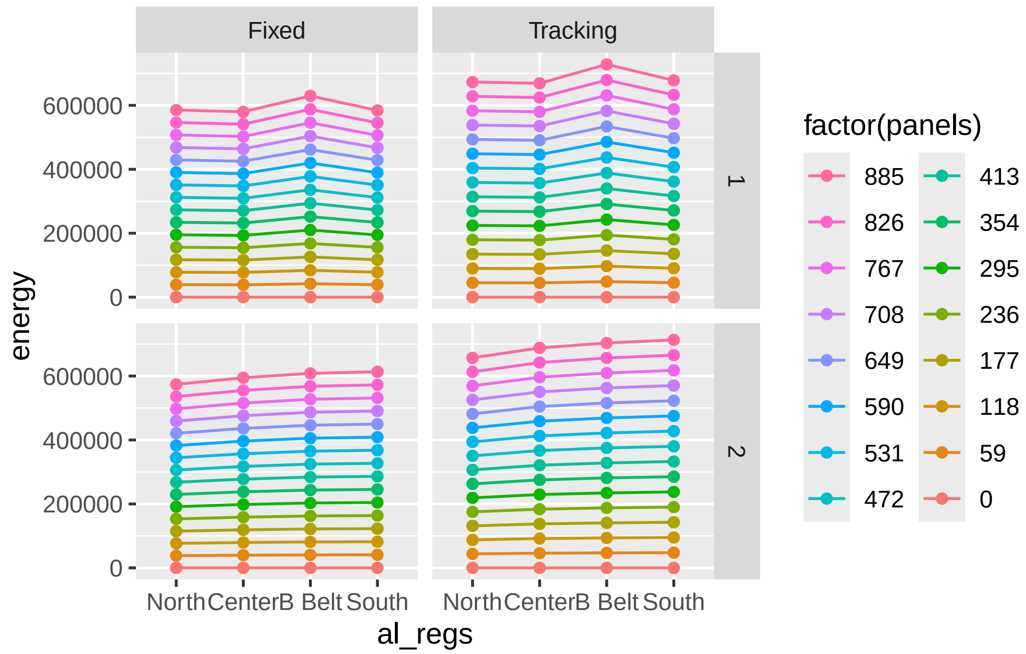
```r
tail(energy_output)
```

```
    sprop panels datalot    al_regs  zips    array dc_kw energy
332     0      0       2    Central 35136 Tracking     0      0
333     0      0       2    Central 35136    Fixed     0      0
334     0      0       2 Black Belt 36040 Tracking     0      0
335     0      0       2 Black Belt 36040    Fixed     0      0
336     0      0       2   Southern 36507 Tracking     0      0
337     0      0       2   Southern 36507    Fixed     0      0
```

2.8.1 Energy output by solar panels counts

Plotting Energy output by number of solar panels in one acres of AV system from fixed and single axis rotation system for two zipcodes (1, 2) within each of the four regions of AL.

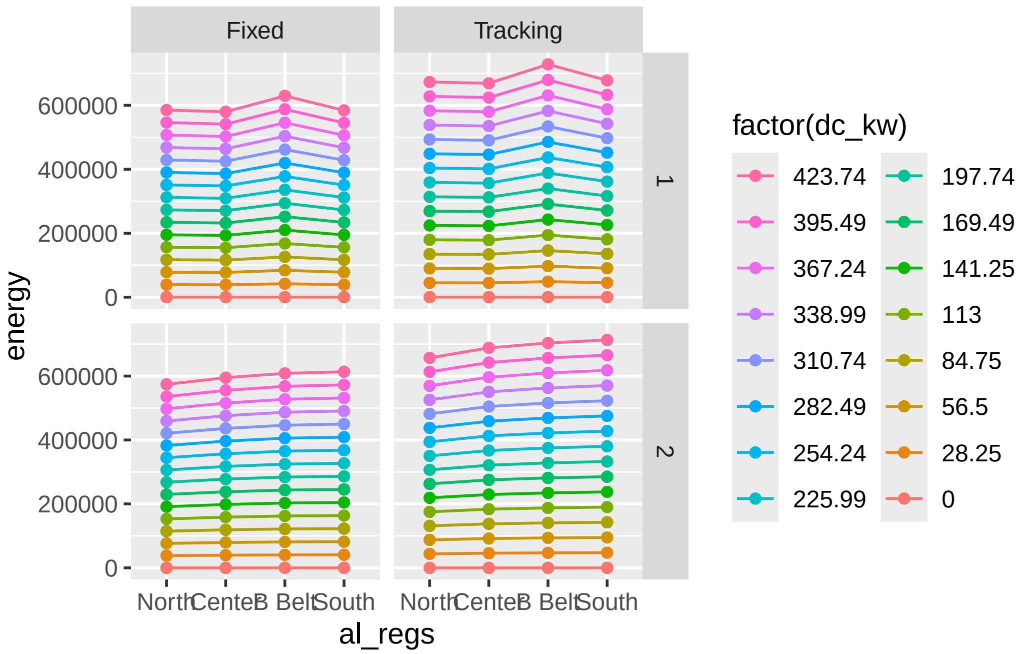
```r
lox <- c("Northern", "Central", "Black Belt", "Southern")
array_levs = c("Single Axis Rotation", "Fixed Open Rack")
datalot_levs = c("Location 1", "Location 2")
ggplot(data = energy_output,
         mapping = aes(x = al_regs,
                       y = energy,
                       #fill = energy,
                       color = factor(panels),
                       group = factor(panels))) +
  geom_line()+
  geom_point() +
  facet_grid(datalot~array) +
  scale_x_discrete(limits = lox,
                   labels = c("North", "Center", "B Belt", "South")) +
  guides(color = guide_legend(ncol = 2, reverse = TRUE))
```



2.8.2 Energy output by DC System Size

Plotting Energy output by DC System Size from fixed and single axis rotation system for two zipcodes (1, 2) within each of the four regions of AL.

```
lox <- c("Northern", "Central", "Black Belt", "Southern")
ggplot(data = energy_output,
         mapping = aes(x = al_regs,
                       y = energy,
                       #fill = energy,
                       color = factor(dc_kw),
                       group = factor(dc_kw))) +
  geom_line()+
  geom_point() +
  facet_grid(datalot~array) +
  scale_x_discrete(limits = lox,
                   labels = c("North", "Center", "B Belt", "South")) +
  guides(color = guide_legend(ncol =2, reverse = TRUE))
```



3 Solar Energy Calculation

3.1 Simulation 1 for energy revenue

- elcprc = electricity price. See Electricity price data for more detail.
- elcrev = Revenue from electricity for given electricity prices. See "energy output" and "electricity price" dataset for more details.
- I filtered datalot 2–I did not take average of "energy" from datalot 1 and datalot 2– to minimize computation time.

```
# Convert to data frames if they are not already
matrix1 <- energy_output  %>%
  group_by(sprop, al_regs, array, dc_kw, panels) %>%
  filter(datalot == 2) %>%
  # Compute mean of datalot 1 and datalot 2:
  summarise(
    energy = mean(energy),
    .groups = 'drop'
    ) # dimension of matrix is 168*6
matrix2 <- elec_price # dimension of matrix is 11*1
```

```r
# Initialize the result data frame
# energy_revenue <- data.frame(matrix(nrow = 1848, ncol = 9))
energy_revenue <- data.frame(
  matrix(nrow = nrow(matrix2)*nrow(matrix1),
         ncol = ncol(matrix2)+ncol(matrix1)+1))

# Variable to keep track of the row index in the result matrix
row_index <- 1

# Loop through each value of the second matrix
for (i in 1:nrow(matrix2)) {
  # Loop through each value of the second matrix
  for (j in 1:nrow(matrix1)) {
    # First matrix, second matrix, combined two matrices.
    new_row <- c(matrix1[j, ],
                 matrix2[i, ],
                 matrix1$energy[j] * matrix2$epr_kwh[i])
    # Assign the new row to the result matrix
    energy_revenue[row_index, ] <- new_row
    # Increment the row index
    row_index <- row_index + 1
  }
}
# Name the columns
colnames(energy_revenue) <- c(colnames(matrix1), "elcprc", "elcrev")

# Display the result
dim(energy_revenue)
```

```
[1] 1848    8
```

```r
head(energy_revenue); tail(energy_revenue)
```

```
  sprop    al_regs    array dc_kw panels energy elcprc elcrev
1     0 Black Belt    Fixed     0      0      0   0.01      0
2     0 Black Belt Tracking     0      0      0   0.01      0
3     0    Central    Fixed     0      0      0   0.01      0
4     0    Central Tracking     0      0      0   0.01      0
5     0   Northern    Fixed     0      0      0   0.01      0
6     0   Northern Tracking     0      0      0   0.01      0
```

```
     sprop  al_regs    array  dc_kw panels energy elcprc   elcrev
1843     1  Central    Fixed 423.74    885 594824   0.06 35689.44
1844     1  Central Tracking 423.74    885 688037   0.06 41282.22
1845     1 Northern    Fixed 423.74    885 574020   0.06 34441.20
1846     1 Northern Tracking 423.74    885 656889   0.06 39413.34
1847     1 Southern    Fixed 423.74    885 613342   0.06 36800.52
1848     1 Southern Tracking 423.74    885 712873   0.06 42772.38
```

```r
# Check for any NAs in the result
if(any(is.na(energy_revenue))) {
  na_indices <- which(is.na(energy_revenue), arr.ind = TRUE)
  print(paste("NAs found at rows:", unique(na_indices[, 1])))
} else {
  print("No NAs found in the result data frame.")
}
```

```
[1] "No NAs found in the result data frame."
```

```r
dim(energy_revenue)
```

```
[1] 1848    8
```

```r
str(energy_revenue)
```

```
'data.frame':   1848 obs. of  8 variables:
 $ sprop  : num  0 0 0 0 0 0 0 0 0.05 0.05 ...
 $ al_regs: chr  "Black Belt" "Black Belt" "Central" "Central" ...
 $ array  : chr  "Fixed" "Tracking" "Fixed" "Tracking" ...
 $ dc_kw  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ panels : num  0 0 0 0 0 0 0 0 0 0 ...
 $ energy : num  0 0 0 0 0 0 0 0 0 0 ...
 $ elcprc : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...
 $ elcrev : num  0 0 0 0 0 0 0 0 0 0 ...
```

```r
head(energy_revenue)
```

```
  sprop    al_regs    array dc_kw panels energy elcprc elcrev
1     0 Black Belt    Fixed     0      0      0   0.01      0
2     0 Black Belt Tracking     0      0      0   0.01      0
```

```
3      0    Central     Fixed     0       0       0   0.01       0
4      0    Central Tracking     0       0       0   0.01       0
5      0   Northern     Fixed     0       0       0   0.01       0
6      0   Northern Tracking     0       0       0   0.01       0
```

```
tail(energy_revenue)
```

```
     sprop  al_regs    array  dc_kw panels energy elcprc   elcrev
1843     1  Central    Fixed 423.74    885 594824   0.06 35689.44
1844     1  Central Tracking 423.74    885 688037   0.06 41282.22
1845     1 Northern    Fixed 423.74    885 574020   0.06 34441.20
1846     1 Northern Tracking 423.74    885 656889   0.06 39413.34
1847     1 Southern    Fixed 423.74    885 613342   0.06 36800.52
1848     1 Southern Tracking 423.74    885 712873   0.06 42772.38
```

3.2 Simulation 2 for energy revenue

This simulation has same result as above (Cross checking above code and output). Results are suppressed but errors and warnings are not. No error and no warnings means code is working as it should.

3.3 Plotting revenue from energy production

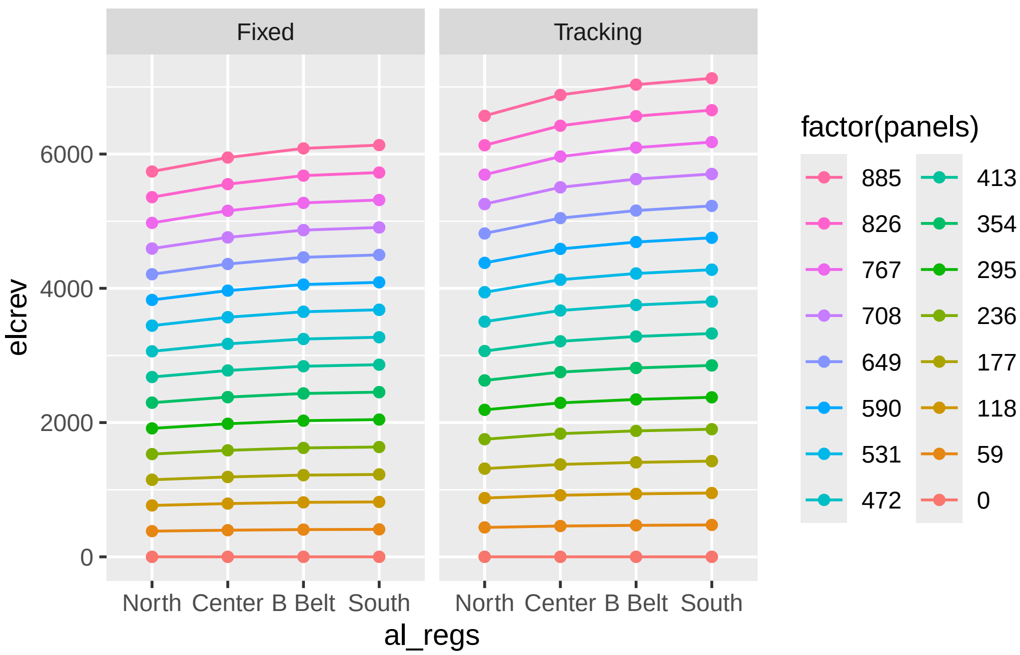
3.3.1 Breakdown by number of solar panels

I am using data from simulation 1 for this visualization. This code plots one chart per electricity cost. There are 11 electricity cost resulting into 11 charts. Electricity revenue is average revenue of first and second lots of simulation.

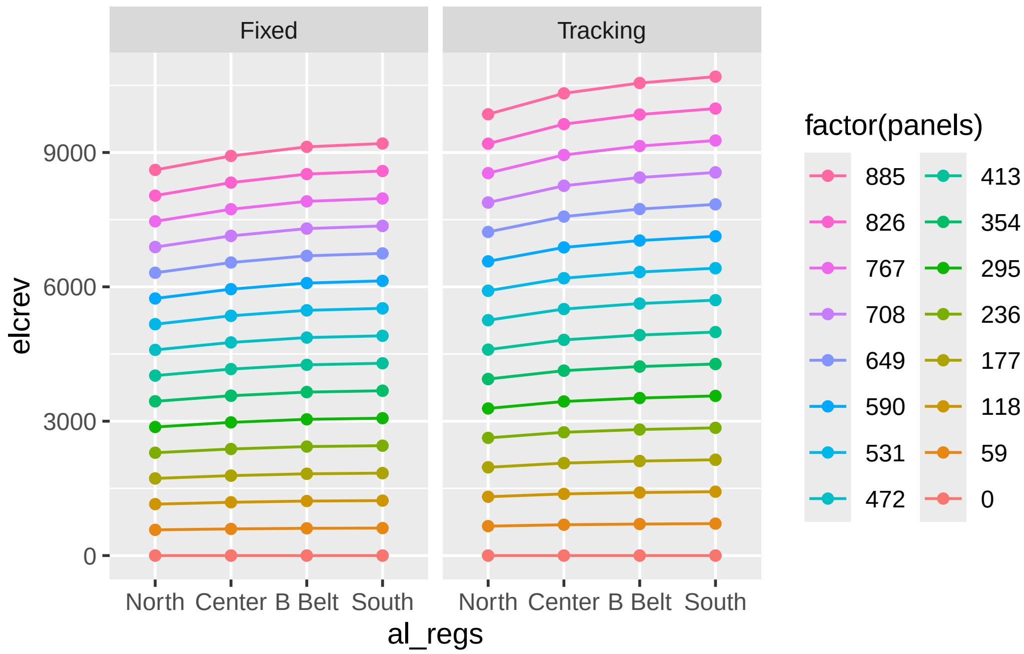
```
lox <- c("Northern", "Central", "Black Belt", "Southern")
array_levs = c("Single Axis Rotation", "Fixed Open Rack")
datalot_levs = c("Location 1", "Location 2")
for (i in unique(energy_revenue$elcprc)) {
 a = ggplot(data = (energy_revenue %>%
  filter(elcprc == i)),
         mapping = aes(x =al_regs,
                       y = elcrev,
                       #fill = energy,
                       color = factor(panels),
                       group = factor(panels)))+
```

```
  geom_line()+
  geom_point()+
  facet_grid(.~array) +
  scale_x_discrete(limits = lox,
                   labels = c("North", "Center", "B Belt", "South")) +
   guides(color = guide_legend(ncol =2, reverse = TRUE))
 cat("Electricity Price = ", i)
 print(a)
}
```

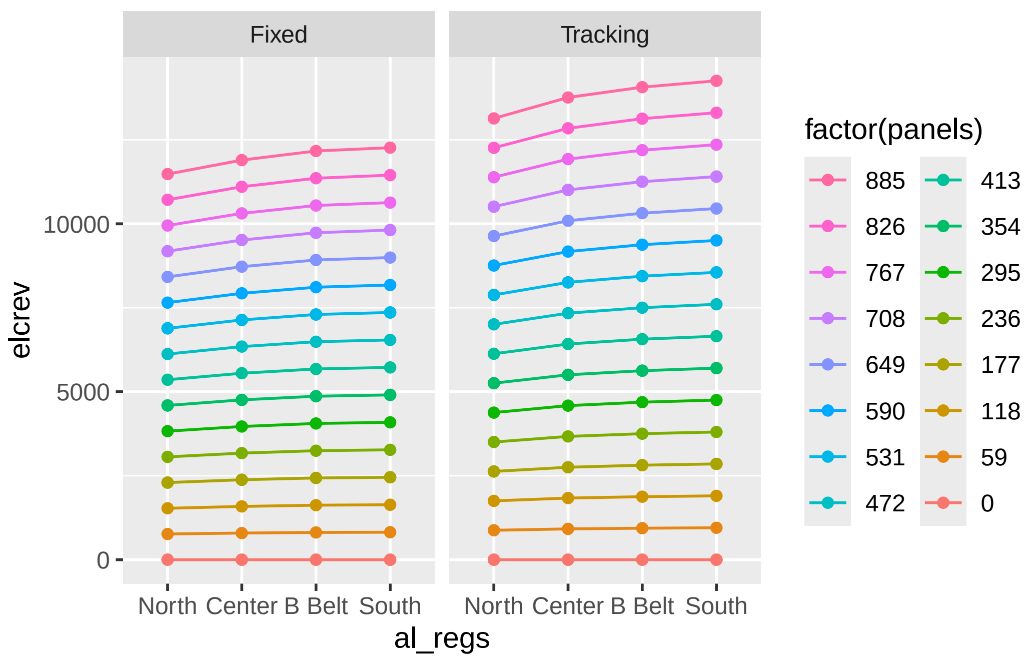
```
Electricity Price =  0.01
```



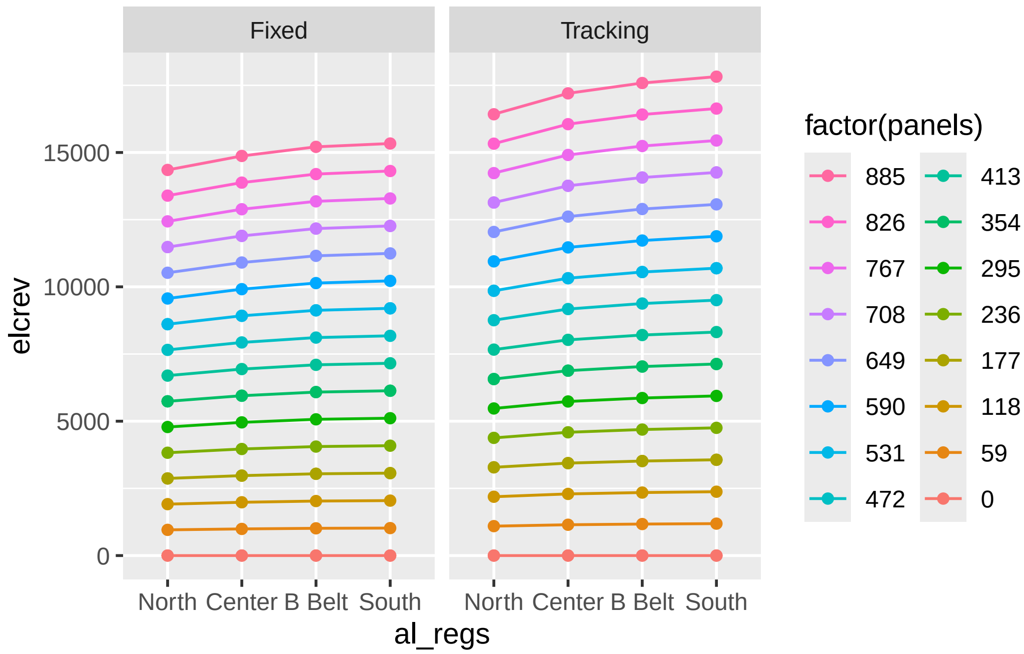

```
Electricity Price =  0.015
```



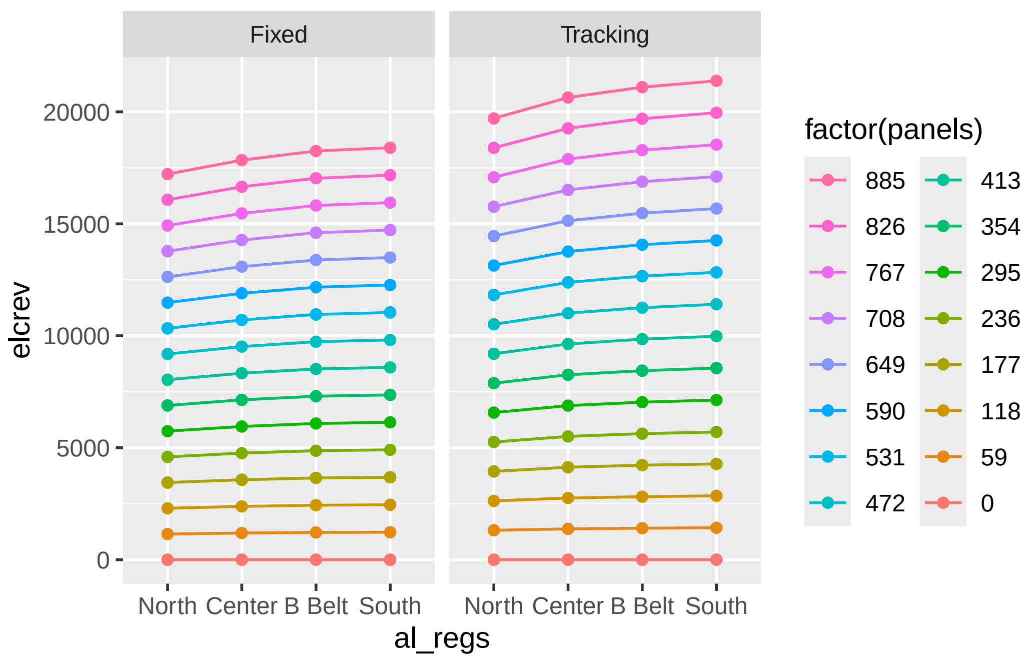
Electricity Price = 0.02



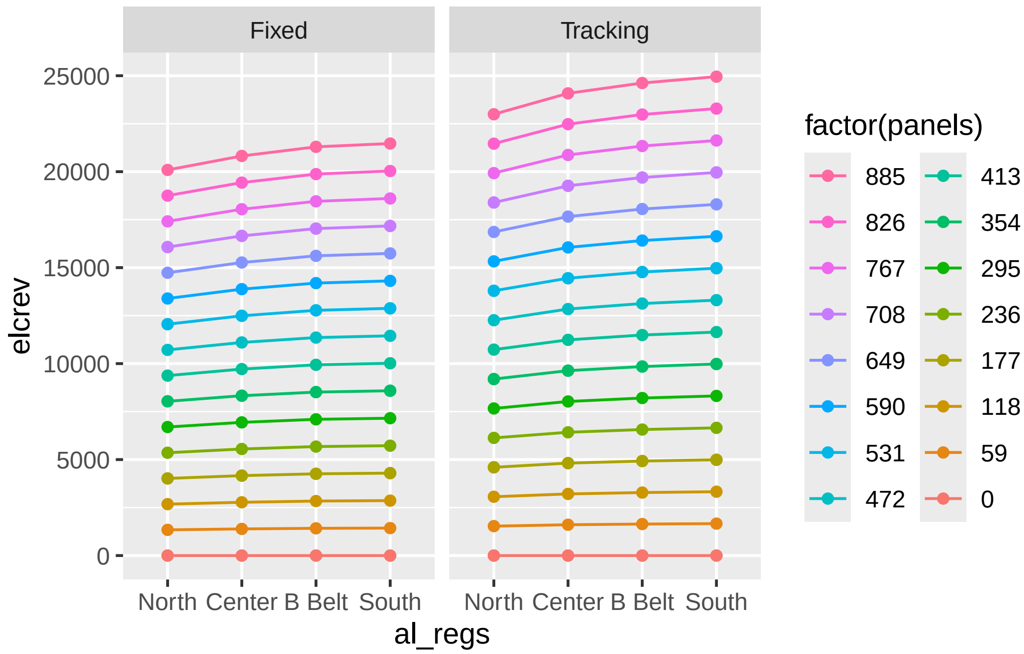
Electricity Price = 0.025



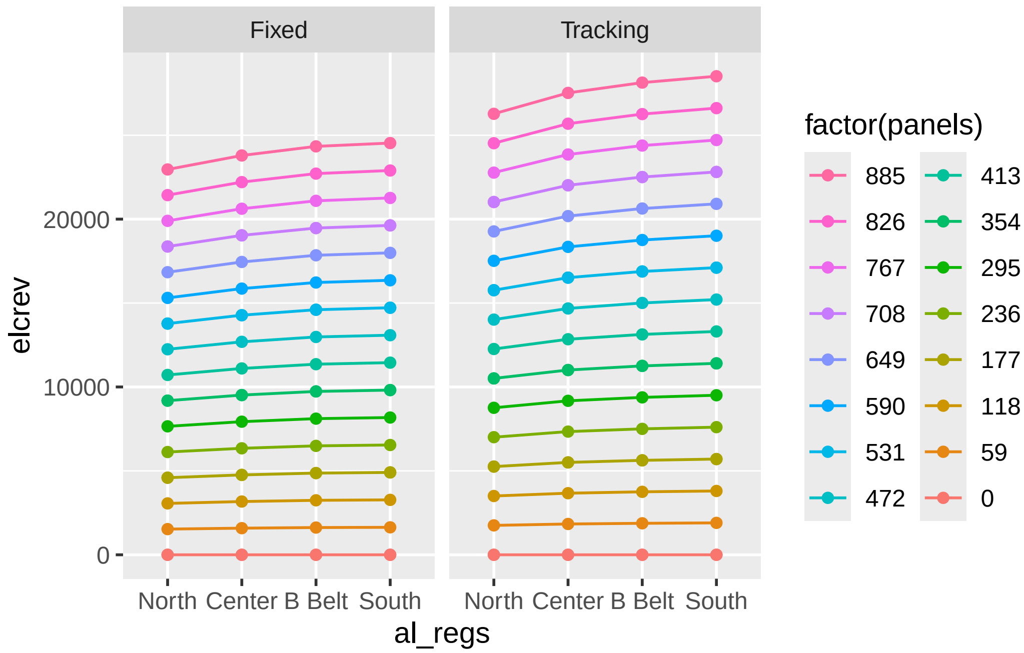
Electricity Price = 0.03



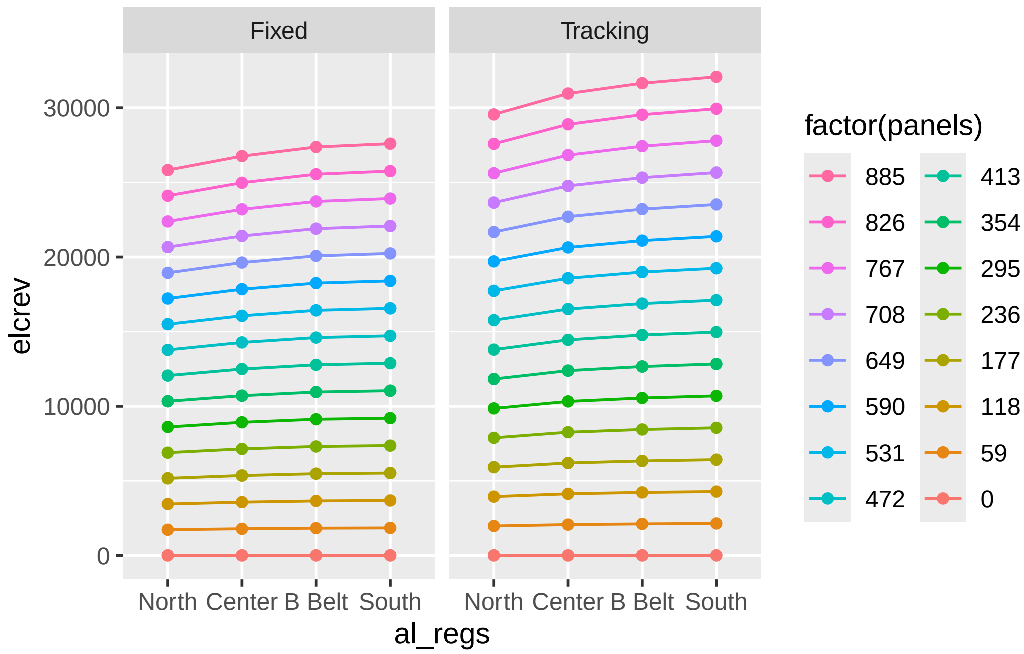
Electricity Price = 0.035



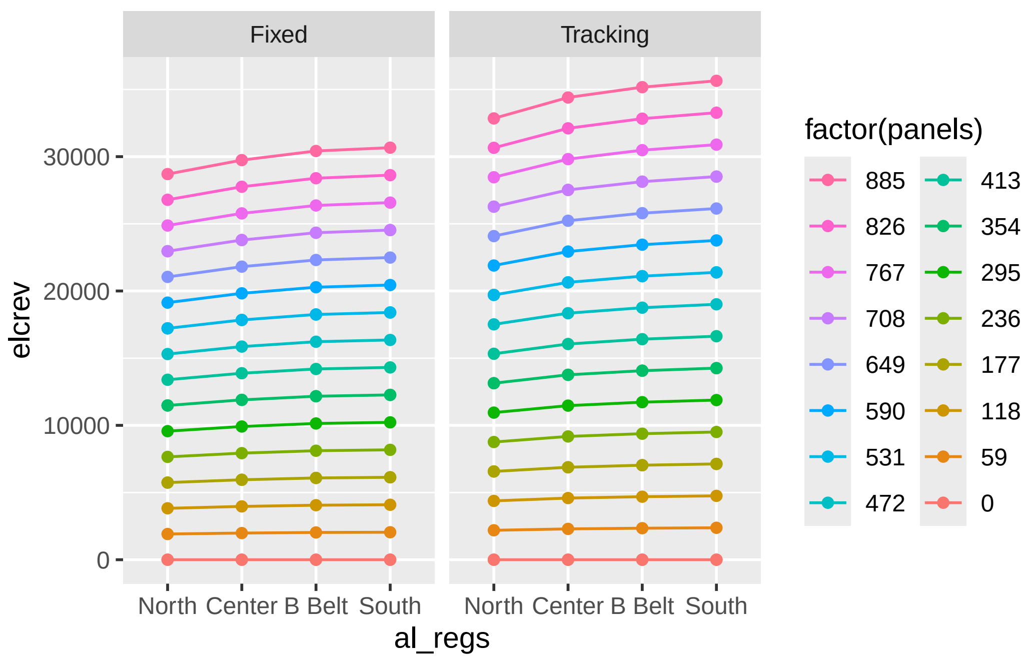

```
Electricity Price =  0.04
```



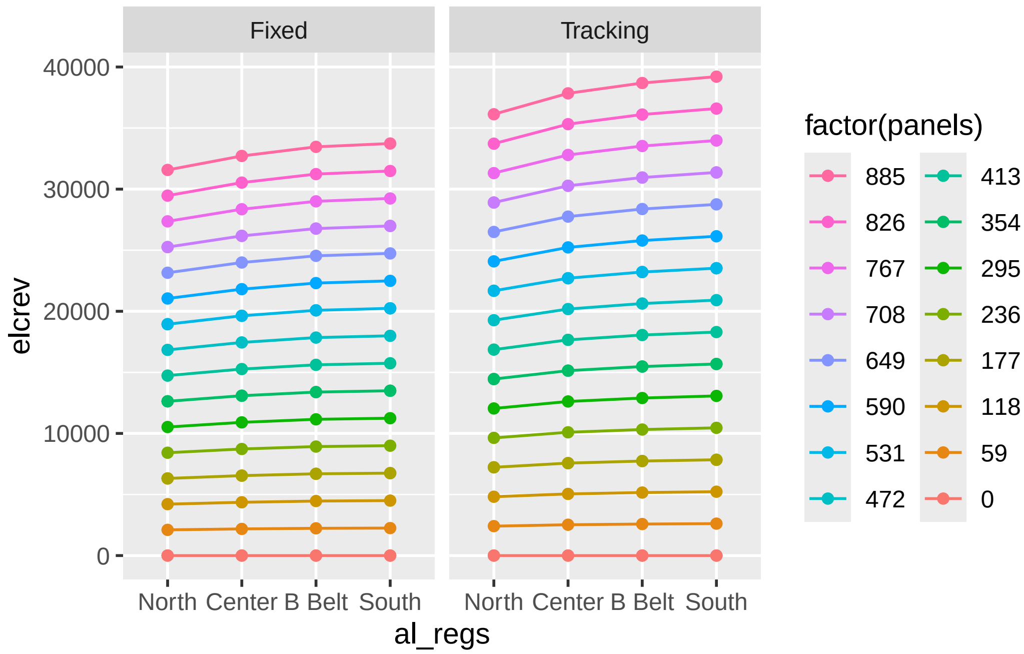

```
Electricity Price =  0.045
```



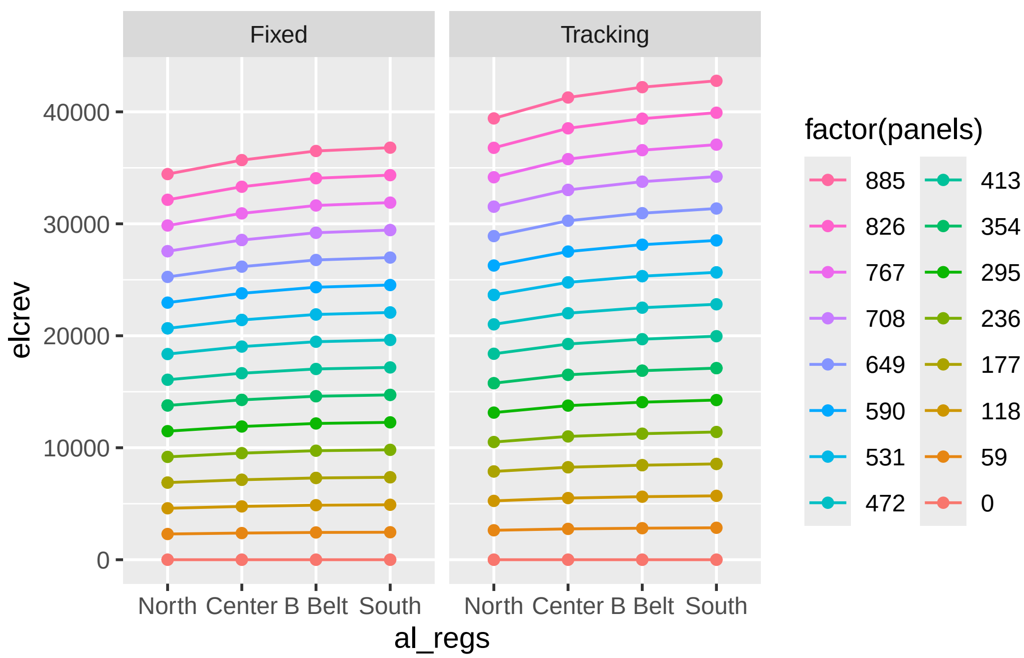

```
Electricity Price =  0.05
```




```
Electricity Price =  0.055
```




```
Electricity Price =  0.06
```



3.3.2 Breakdown by proportion of land under solar panels

- Two proportions may have same number of solar panels (Eg. 0.80 and 0.85, 0.20 and 0.25). So, total lines in the chart may not match with total number of legend levels. Some proportions are overlapping in the chart. See panel configuration for more detail.

```
lox <- c("Northern", "Central", "Black Belt", "Southern")
array_levs = c("Single Axis Rotation", "Fixed Open Rack")
datalot_levs = c("Location 1", "Location 2")
for (i in unique(energy_revenue$elcprc)) {
 a = ggplot(data = (energy_revenue %>%
  filter(elcprc == i)),
         mapping = aes(x =al_regs,
                       y = elcrev,
                       #fill = energy,
                       color = factor(sprop),
                       group = factor(sprop)))+
  geom_line()+
  geom_point()+
  facet_grid(.~array) +
  scale_x_discrete(limits = lox,
                   labels = c("North", "Center", "B Belt", "South")) +
   guides(color = guide_legend(ncol = 2, reverse = TRUE))
 cat("Electricity Price = ", i)
 print(a)
}
```

3.4 Solar system cost

- Cost of solar energy system in agrivoltaic setting.
- I used energy output per 7.75 ft.*3.5 ft. panel (545 w), capex ($/w), and total number of panels to get total cost for each height and panel tracking system.
- height = height of solar panels; see capex dataset for details.
- capex = capex from capex table; see capex dataset for details.
- opex = Operational cost ($15/kW/Year); Source: Ramasamy, 2022. PV Cost Benchmark
- ttlcost = Total cost for given DC system size.
- anncost = Annual payment to repay loan $(P_{ann}) = \frac{P_o(i(1+i)^t)}{(1+i)^t-1)}$, where P_o = CAPEX loan burrowed to repay in t years; $t = 25$, and i = annual interest rate at 5%.

- moncost = Monthly payment to repay loan (P_{mon}) = $\frac{P_o((i/12)(1+(i/12))^{t*12})}{(1+(i/12))^{t*12}-1)}$, where P_o = CAPEX loan burrowed to repay in t years; $t = 25$, and i = annual interest rate at 5%.

```
cat("energy_revenue dimension = ", dim(energy_revenue)) # 1936*9
```

```
energy_revenue dimension =  1848 8
```

```
names(energy_revenue)
```

```
[1] "sprop"   "al_regs" "array"   "dc_kw"   "panels"  "energy"  "elcprc"
[8] "elcrev"
```

```
cat("capex = ", dim(capex)) # 6*3
```

```
capex =  6 3
```

```
names(capex)
```

```
[1] "height" "capex"  "array"
```

```
r = 0.04 # Discount/interest Rate
n = 30 # Life Span of solar panels (Years)
expanded_data <- energy_revenue %>%
  slice(rep(1:n(),
            each = 3))
capex_height <- rep(unique(capex$height),
                    length.out = nrow(energy_revenue))

energy_cost = cbind(expanded_data, capex_height) %>%
  rename(height = capex_height)

energy_cost <- left_join(energy_cost,
                         capex,
                         by = c("array", "height")) %>%
  mutate(
    landlease = 1000, #$/Acre/Year
    # 7.75*3.5 sq.ft. panel energy output = 545 W.
    # Operational cost (OPEX) = $15/kW-yr; 1 kW = 1,000W.
    opex = 545*15/1000*panels,
```

```
    ttlcost = capex*545*panels,
    recredit = 6.60/1000*energy, # Renewable energy credit 6.60 $/MWh
    anncost = ttlcost*(r*(1+r)**n)/((1+r)**n-1)+opex,
    moncost = ttlcost*((r/12)*(1+(r/n))^(n*12))/((1+(r/12))^(n*12)-1)+opex/12
    )
dim(energy_cost)
```

```
[1] 5544   16
```

```
str(energy_cost)
```

```
'data.frame':   5544 obs. of  16 variables:
 $ sprop    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ al_regs  : chr  "Black Belt" "Black Belt" "Black Belt" "Black Belt" ...
 $ array    : chr  "Fixed" "Fixed" "Fixed" "Tracking" ...
 $ dc_kw    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ panels   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ energy   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ elcprc   : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...
 $ elcrev   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ height   : num  4.6 6.4 8.2 4.6 6.4 8.2 4.6 6.4 8.2 4.6 ...
 $ capex    : num  1.59 1.85 2.33 1.73 1.92 ...
 $ landlease: num  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...
 $ opex     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ttlcost  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ recredit : num  0 0 0 0 0 0 0 0 0 0 ...
 $ anncost  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ moncost  : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
head(energy_cost)
```

```
  sprop    al_regs    array dc_kw panels energy elcprc elcrev height    capex
1     0 Black Belt    Fixed     0      0      0   0.01      0    4.6 1.593333
2     0 Black Belt    Fixed     0      0      0   0.01      0    6.4 1.850000
3     0 Black Belt    Fixed     0      0      0   0.01      0    8.2 2.330000
4     0 Black Belt Tracking     0      0      0   0.01      0    4.6 1.733333
5     0 Black Belt Tracking     0      0      0   0.01      0    6.4 1.921667
6     0 Black Belt Tracking     0      0      0   0.01      0    8.2 2.110000
  landlease opex ttlcost recredit anncost moncost
1      1000    0       0        0       0       0
```

```
2       1000    0       0       0       0       0
3       1000    0       0       0       0       0
4       1000    0       0       0       0       0
5       1000    0       0       0       0       0
6       1000    0       0       0       0       0
```

```r
tail(energy_cost)
```

```
     sprop  al_regs    array  dc_kw panels energy elcprc   elcrev height
5539     1 Southern    Fixed 423.74    885 613342   0.06 36800.52    4.6
5540     1 Southern    Fixed 423.74    885 613342   0.06 36800.52    6.4
5541     1 Southern    Fixed 423.74    885 613342   0.06 36800.52    8.2
5542     1 Southern Tracking 423.74    885 712873   0.06 42772.38    4.6
5543     1 Southern Tracking 423.74    885 712873   0.06 42772.38    6.4
5544     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
        capex landlease     opex   ttlcost recredit  anncost  moncost
5539 1.593333      1000 7234.875  768504.5 4048.057 51677.57 2391.775
5540 1.850000      1000 7234.875  892301.2 4048.057 58836.74 2679.940
5541 2.330000      1000 7234.875 1123817.3 4048.057 72225.34 3218.846
5542 1.733333      1000 7234.875  836030.0 4704.962 55582.57 2548.956
5543 1.921667      1000 7234.875  926867.9 4704.962 60835.74 2760.402
5544 2.110000      1000 7234.875 1017705.8 4704.962 66088.90 2971.848
```

3.5 Profit from solar

Profit from solar energy system in agrivoltaic setting

- eprofit = profit from electricity after subtracting total cost (ttlcost) from total revenue (elcrev).
- eannprof = annual profit from solar after subtracting annual loan repayment distributed over 25 years.
- emonprof = monthly profit from solar after subtracting monthly loan repayment distributed over 25 years.

```r
solar_profit <- energy_cost %>%
  mutate(
    eprofit = elcrev - ttlcost,
    eannprof = elcrev - anncost + recredit,
    emonprof = elcrev/12 - moncost + recredit/12)
dim(solar_profit)
```

```
[1] 5544   19
```

```r
str(solar_profit)
```

```
'data.frame':   5544 obs. of  19 variables:
 $ sprop    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ al_regs  : chr  "Black Belt" "Black Belt" "Black Belt" "Black Belt" ...
 $ array    : chr  "Fixed" "Fixed" "Fixed" "Tracking" ...
 $ dc_kw    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ panels   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ energy   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ elcprc   : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...
 $ elcrev   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ height   : num  4.6 6.4 8.2 4.6 6.4 8.2 4.6 6.4 8.2 4.6 ...
 $ capex    : num  1.59 1.85 2.33 1.73 1.92 ...
 $ landlease: num  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...
 $ opex     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ttlcost  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ recredit : num  0 0 0 0 0 0 0 0 0 0 ...
 $ anncost  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ moncost  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eprofit  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eannprof : num  0 0 0 0 0 0 0 0 0 0 ...
 $ emonprof : num  0 0 0 0 0 0 0 0 0 0 ...
```

```r
head(solar_profit)
```

```
  sprop    al_regs    array dc_kw panels energy elcprc elcrev height    capex
1     0 Black Belt    Fixed     0      0      0   0.01      0    4.6 1.593333
2     0 Black Belt    Fixed     0      0      0   0.01      0    6.4 1.850000
3     0 Black Belt    Fixed     0      0      0   0.01      0    8.2 2.330000
4     0 Black Belt Tracking     0      0      0   0.01      0    4.6 1.733333
5     0 Black Belt Tracking     0      0      0   0.01      0    6.4 1.921667
6     0 Black Belt Tracking     0      0      0   0.01      0    8.2 2.110000
  landlease opex ttlcost recredit anncost moncost eprofit eannprof emonprof
1      1000    0       0        0       0       0       0        0        0
2      1000    0       0        0       0       0       0        0        0
3      1000    0       0        0       0       0       0        0        0
4      1000    0       0        0       0       0       0        0        0
5      1000    0       0        0       0       0       0        0        0
6      1000    0       0        0       0       0       0        0        0
```

```r
tail(solar_profit)
```

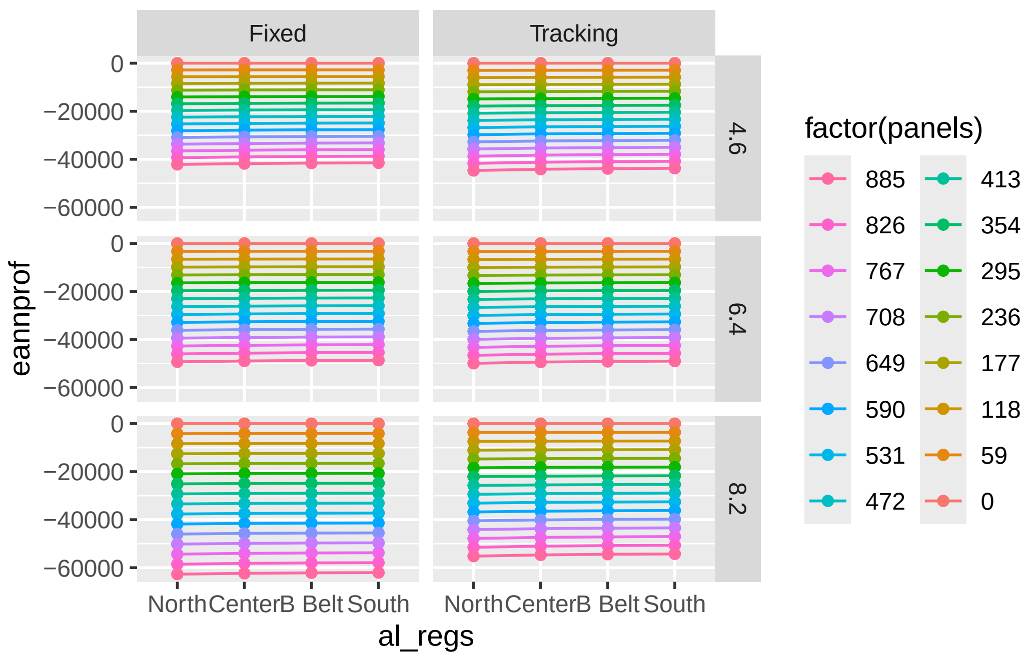
```
     sprop  al_regs    array  dc_kw panels energy elcprc   elcrev height
5539     1 Southern    Fixed 423.74    885 613342   0.06 36800.52    4.6
5540     1 Southern    Fixed 423.74    885 613342   0.06 36800.52    6.4
5541     1 Southern    Fixed 423.74    885 613342   0.06 36800.52    8.2
5542     1 Southern Tracking 423.74    885 712873   0.06 42772.38    4.6
5543     1 Southern Tracking 423.74    885 712873   0.06 42772.38    6.4
5544     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
        capex landlease     opex   ttlcost recredit  anncost  moncost
5539 1.593333      1000 7234.875  768504.5 4048.057 51677.57 2391.775
5540 1.850000      1000 7234.875  892301.2 4048.057 58836.74 2679.940
5541 2.330000      1000 7234.875 1123817.3 4048.057 72225.34 3218.846
5542 1.733333      1000 7234.875  836030.0 4704.962 55582.57 2548.956
5543 1.921667      1000 7234.875  926867.9 4704.962 60835.74 2760.402
5544 2.110000      1000 7234.875 1017705.8 4704.962 66088.90 2971.848
        eprofit   eannprof   emonprof
5539  -731704.0 -10828.989 1012.2729
5540  -855500.7 -17988.168  724.1078
5541 -1087016.7 -31376.761  185.2017
5542  -793257.6  -8105.231 1407.4890
5543  -884095.5 -13358.394 1196.0432
5544  -974933.4 -18611.558  984.5974
```

3.5.1 Plot Solar profit

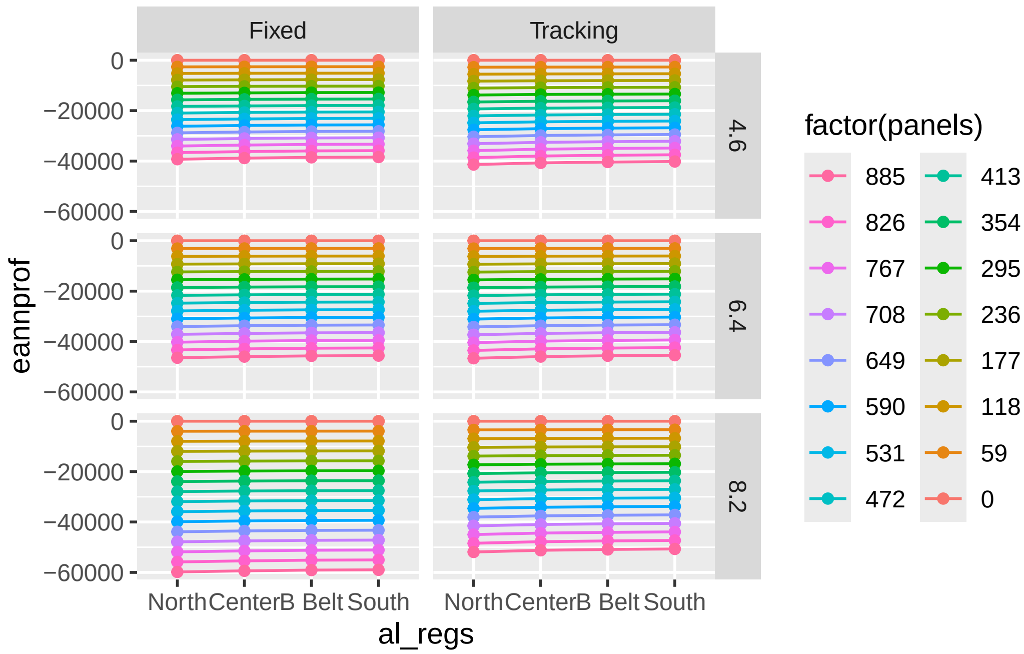
```r
lox <- c("Northern", "Central", "Black Belt", "Southern")
array_levs = c("Single Axis Rotation", "Fixed Open Rack")
datalot_levs = c("Location 1", "Location 2")
  for (i in unique(solar_profit$elcprc)) {
    b = ggplot(
      data = (solar_profit %>%
                filter(elcprc == i)),
      mapping = aes(
        x = al_regs,
        y = eannprof, #Annual Profit
        #fill = energy,
        color = factor(panels),
        group = factor(panels)
      )
```

```
    ) +
      geom_line() +
      geom_point() +
      facet_grid(height ~ array) +
      scale_x_discrete(limits = lox,
                       labels = c("North", "Center",
                                  "B Belt", "South")) +
      guides(color = guide_legend(ncol = 2,
                                  reverse = TRUE))
    cat("Electricity Price = ", i)
    print(b)
  }
```

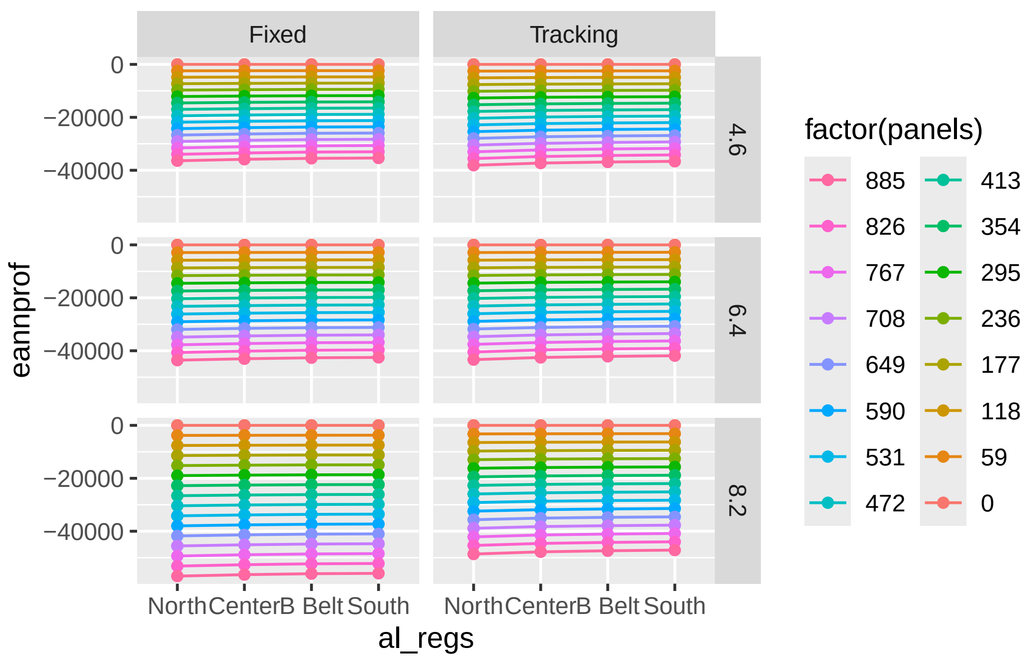
```
Electricity Price =  0.01
```



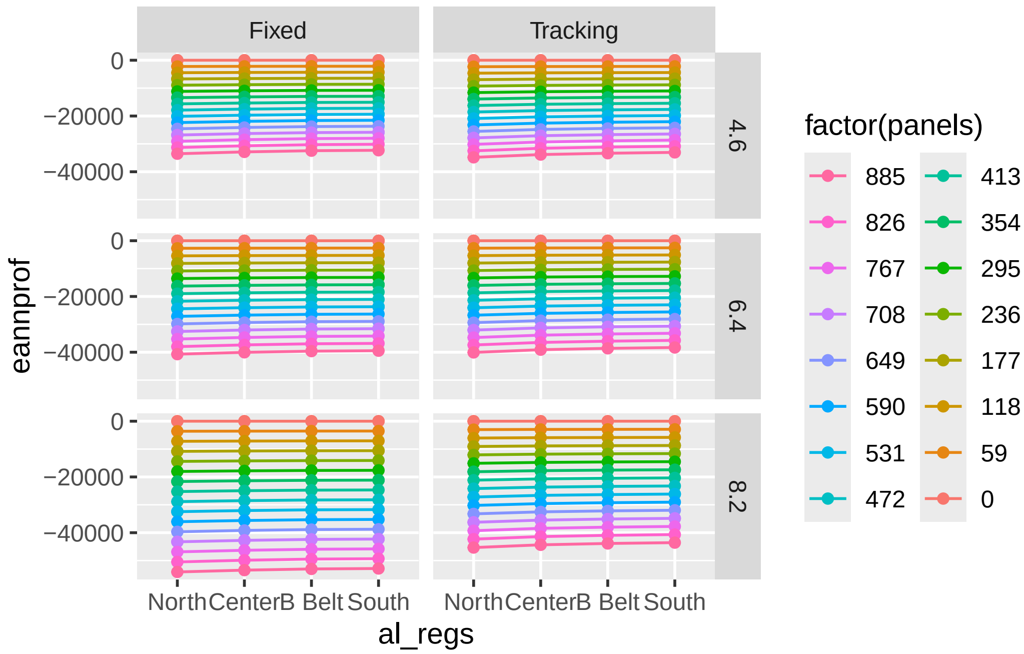

```
Electricity Price =  0.015
```



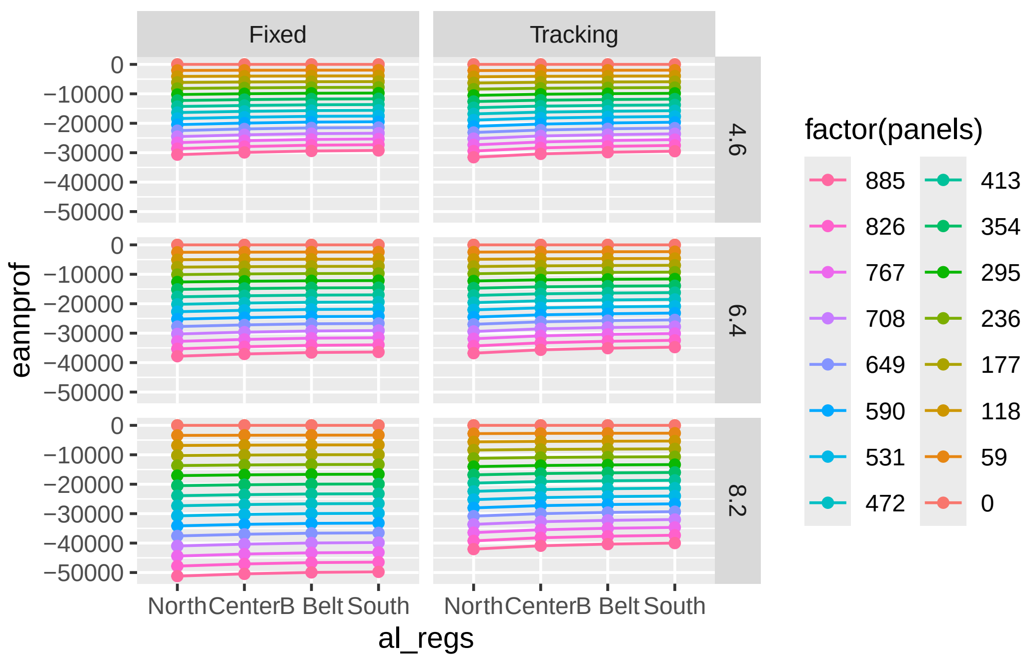
Electricity Price = 0.02



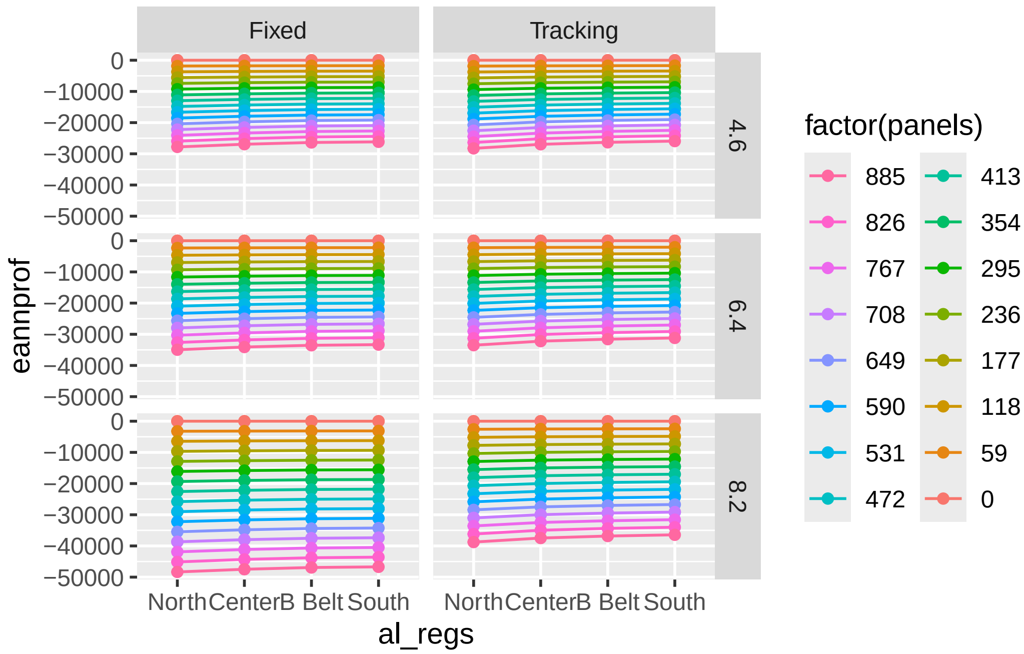
Electricity Price = 0.025



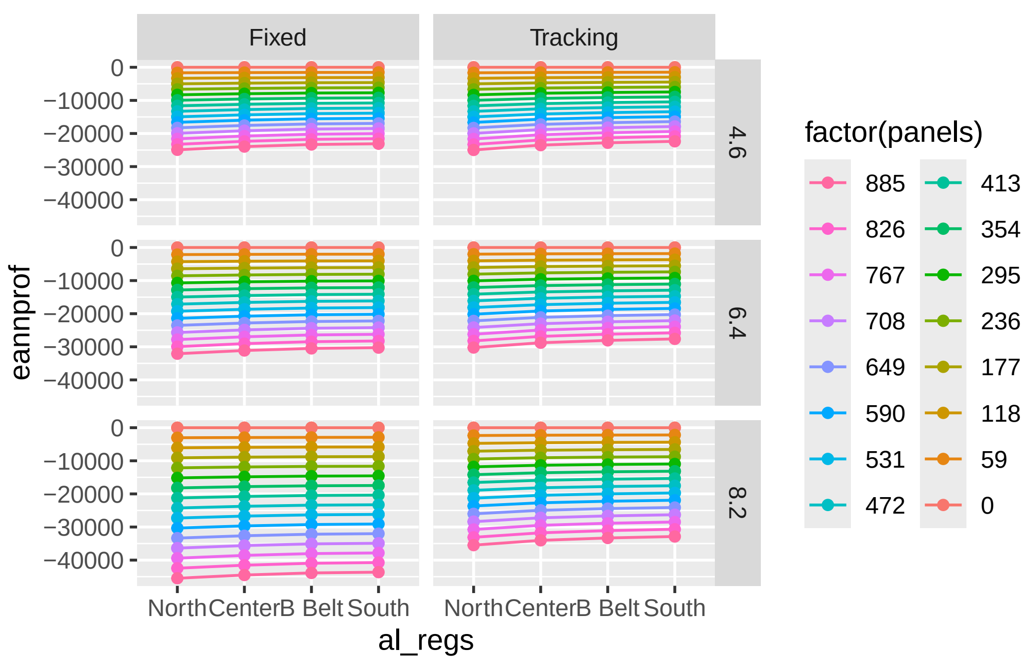
`Electricity Price =  0.03`



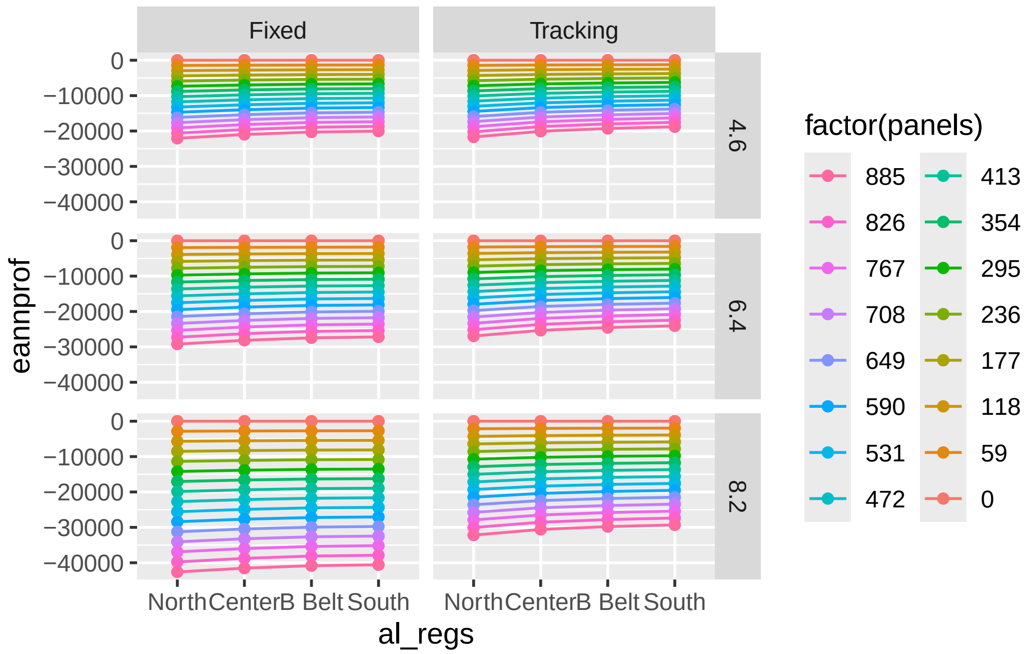
`Electricity Price =  0.035`



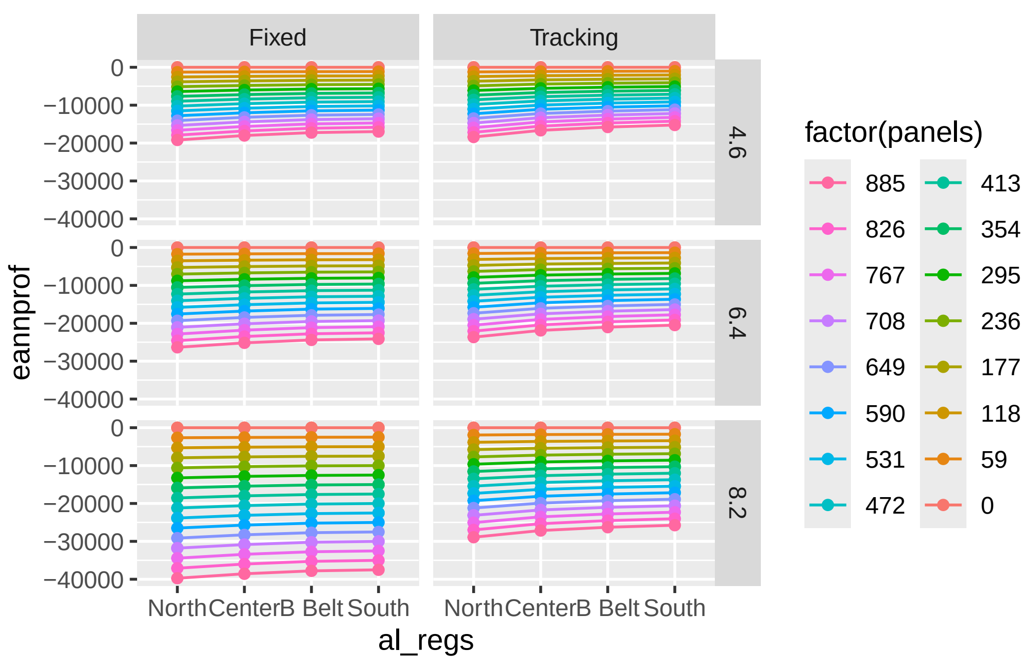

```
Electricity Price =  0.04
```



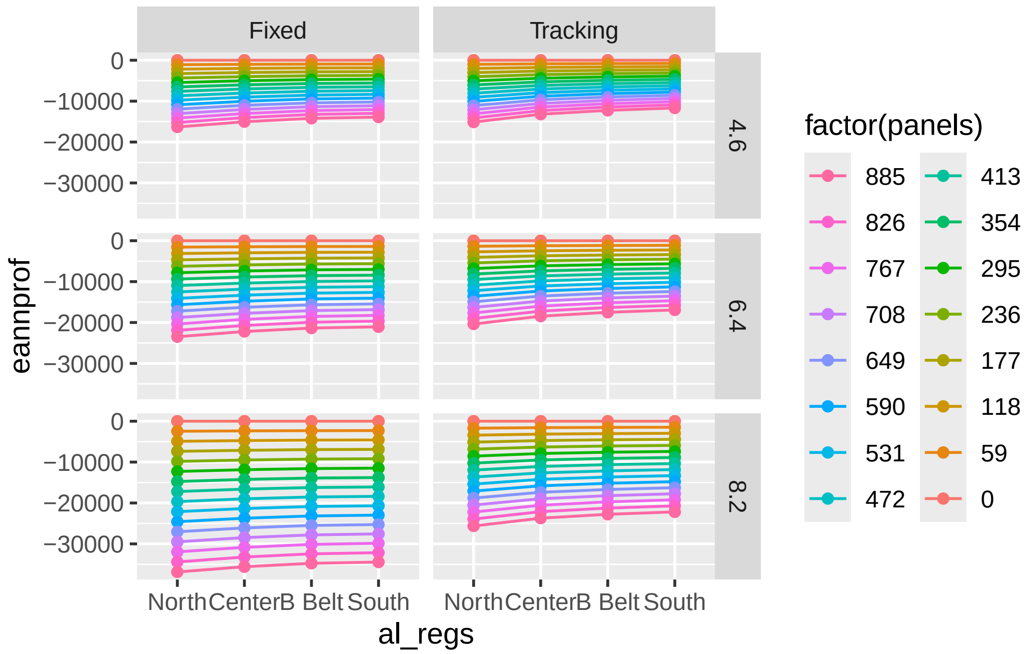

```
Electricity Price =  0.045
```



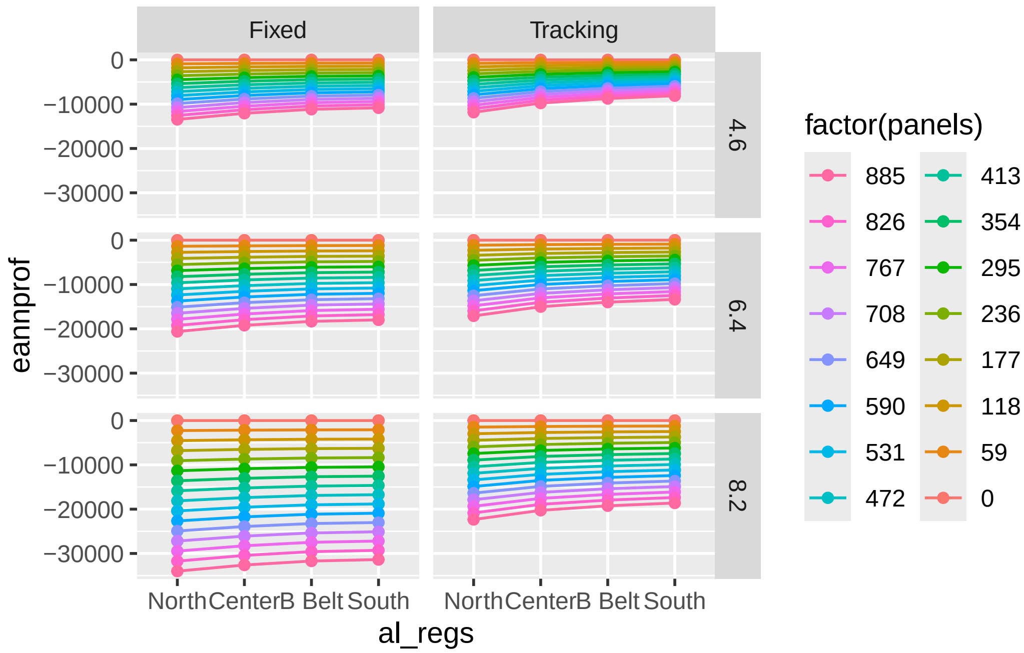
Electricity Price = 0.05



Electricity Price = 0.055

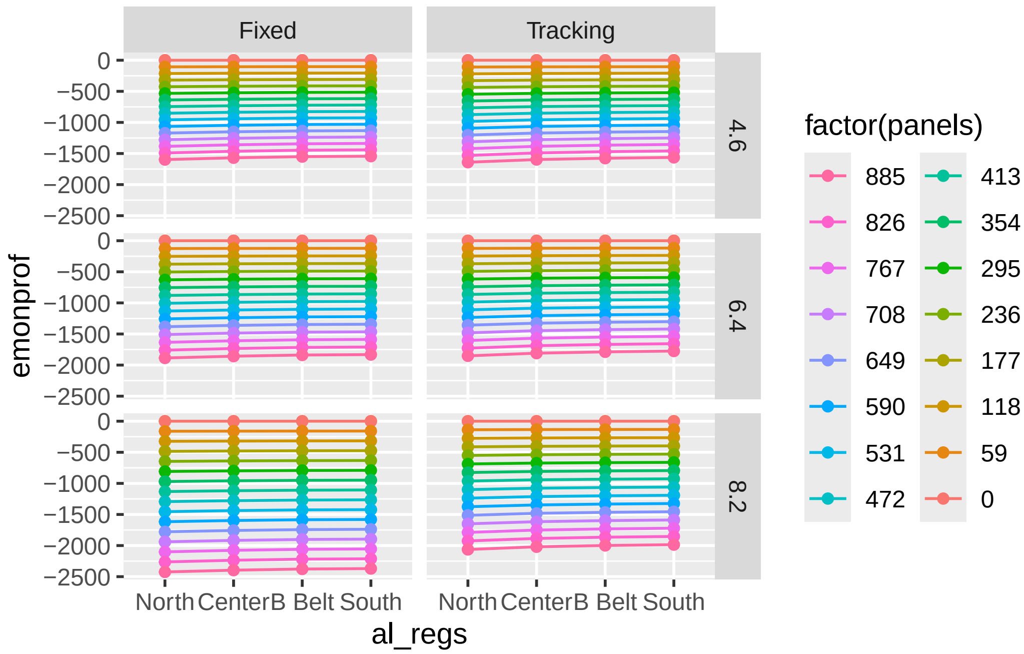


Electricity Price = 0.06

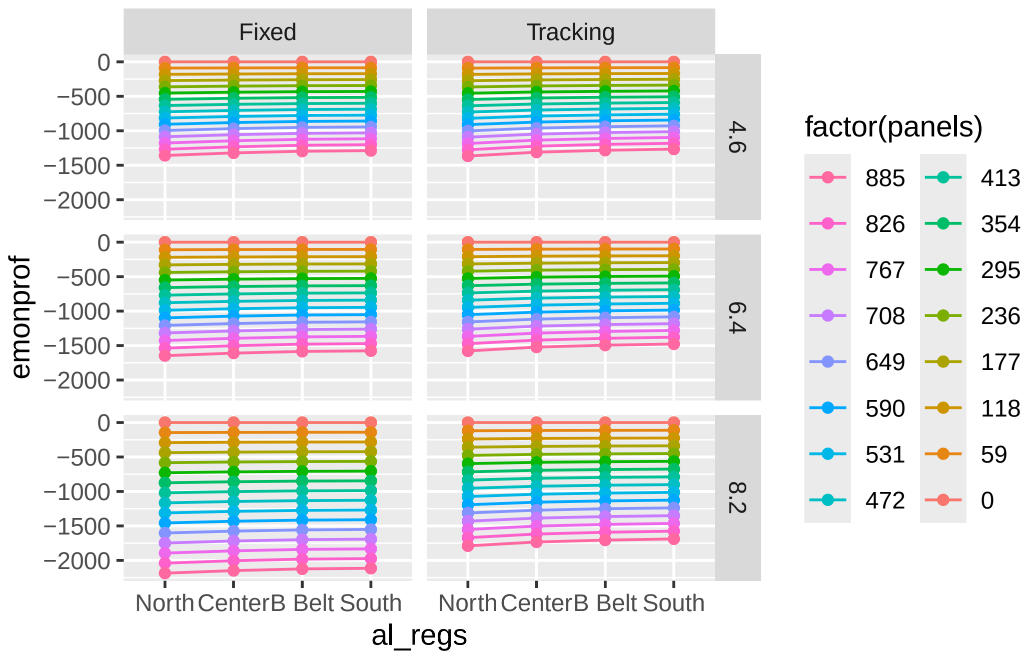


```r
lox <- c("Northern", "Central", "Black Belt", "Southern")
array_levs = c("Single Axis Rotation", "Fixed Open Rack")
datalot_levs = c("Location 1", "Location 2")
  for (i in unique(solar_profit$elcprc)) {
    b = ggplot(
      data = (solar_profit %>%
                filter(elcprc == i)),
      mapping = aes(
        x = al_regs,
        y = emonprof, #Monthly Profit.
        #fill = energy,
        color = factor(panels),
        group = factor(panels)
      )
    ) +
      geom_line() +
      geom_point() +
      facet_grid(height ~ array) +
      scale_x_discrete(limits = lox,
                       labels = c("North", "Center",
                                  "B Belt", "South")) +
      guides(color = guide_legend(ncol = 2,
                                  reverse = TRUE))
    cat("Electricity Price = ", i)
    print(b)
  }
```

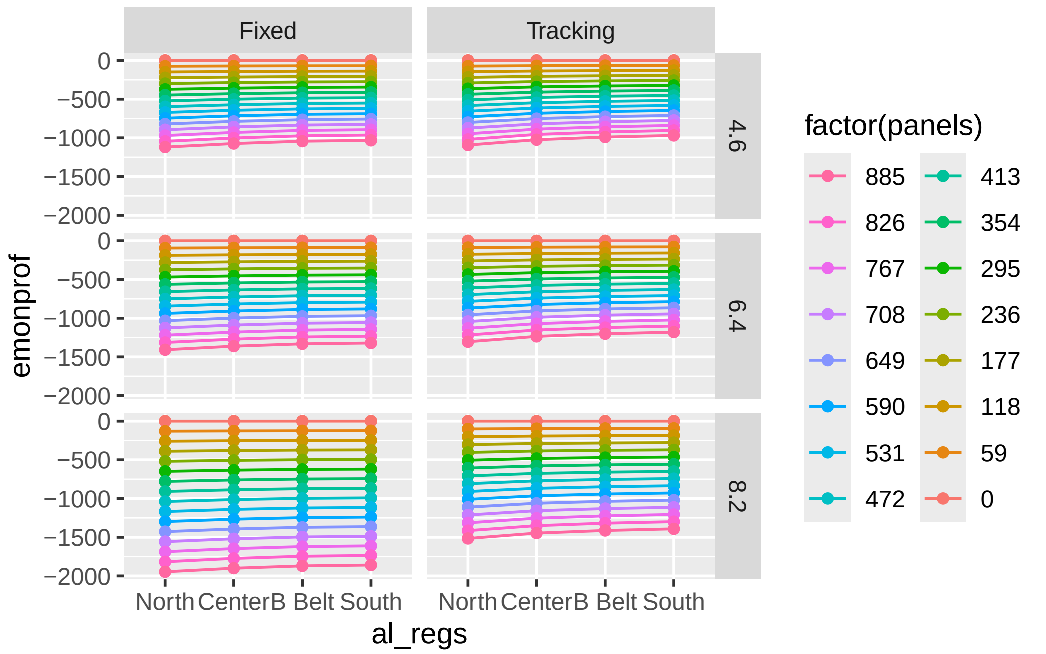
```
Electricity Price =  0.01
```



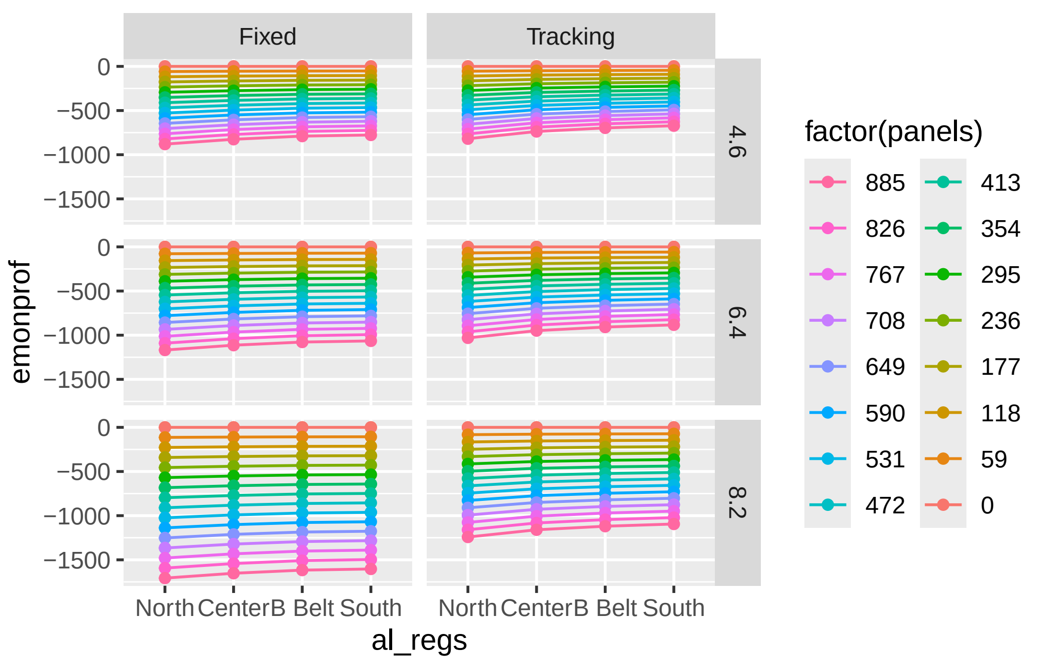

```
Electricity Price =  0.015
```



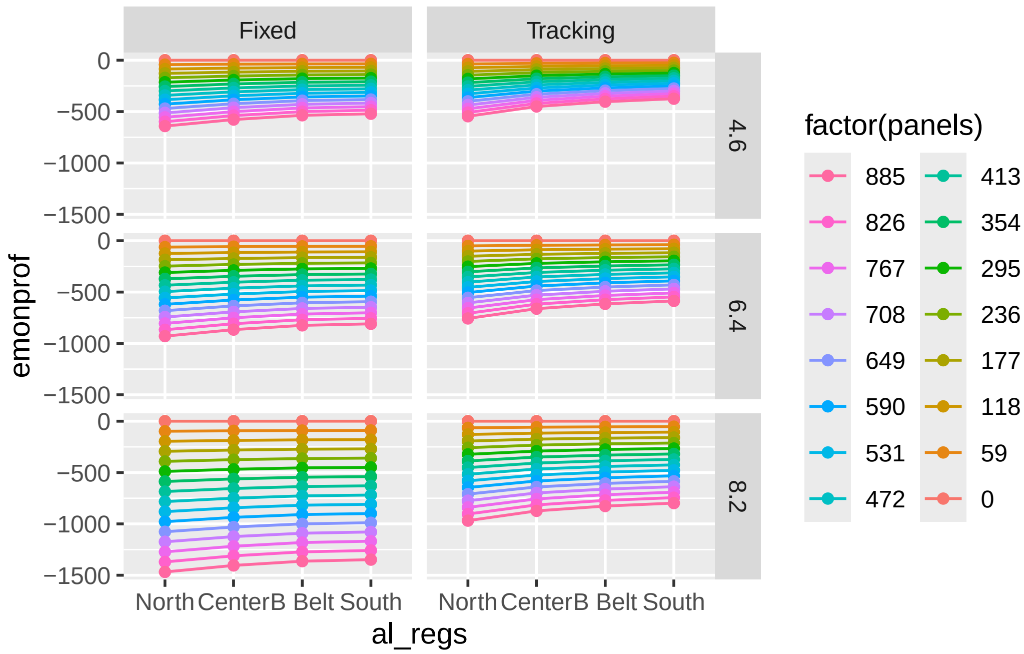

```
Electricity Price =  0.02
```



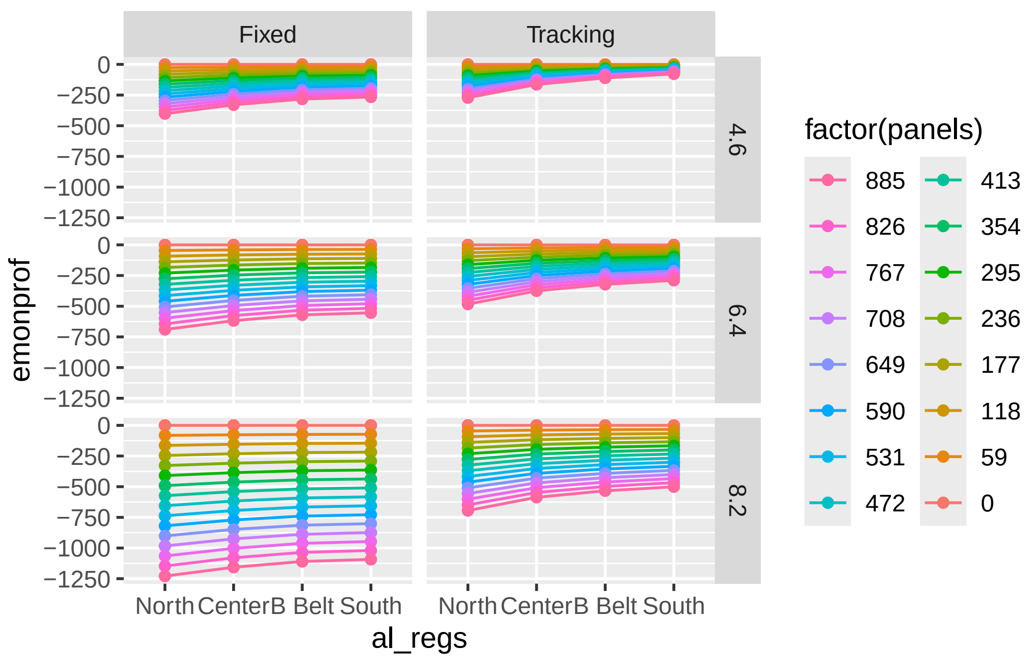

```
Electricity Price =  0.025
```



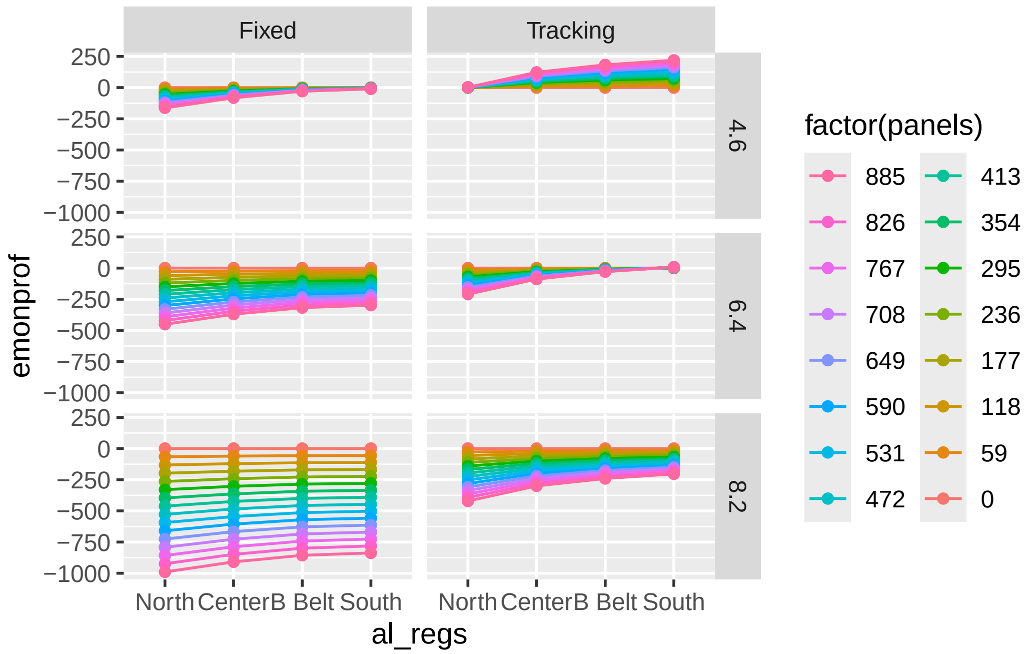

```
Electricity Price =  0.03
```



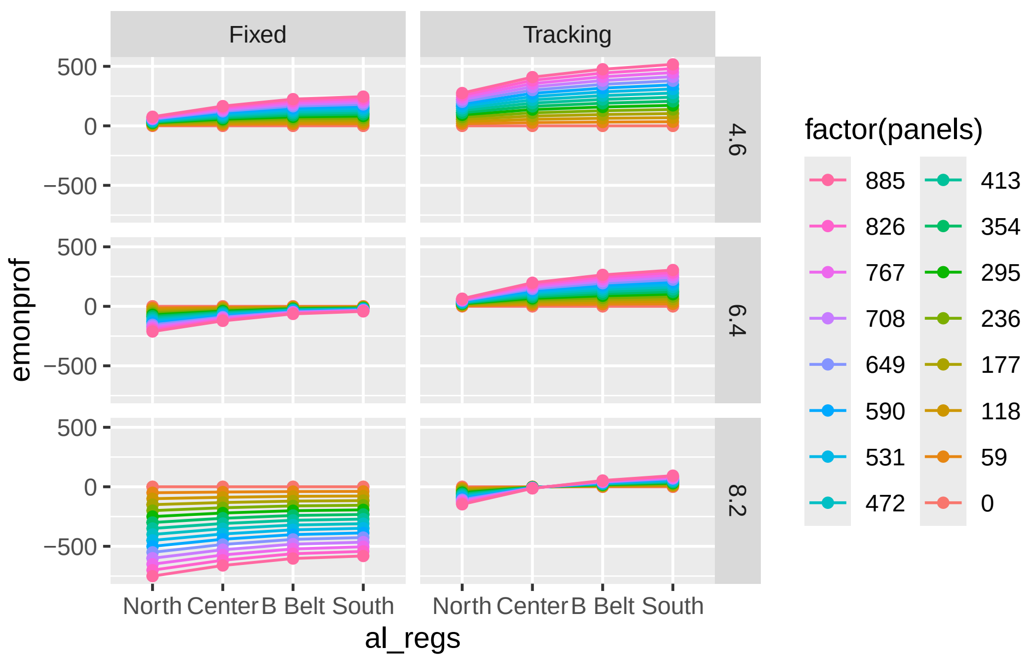

```
Electricity Price =  0.035
```



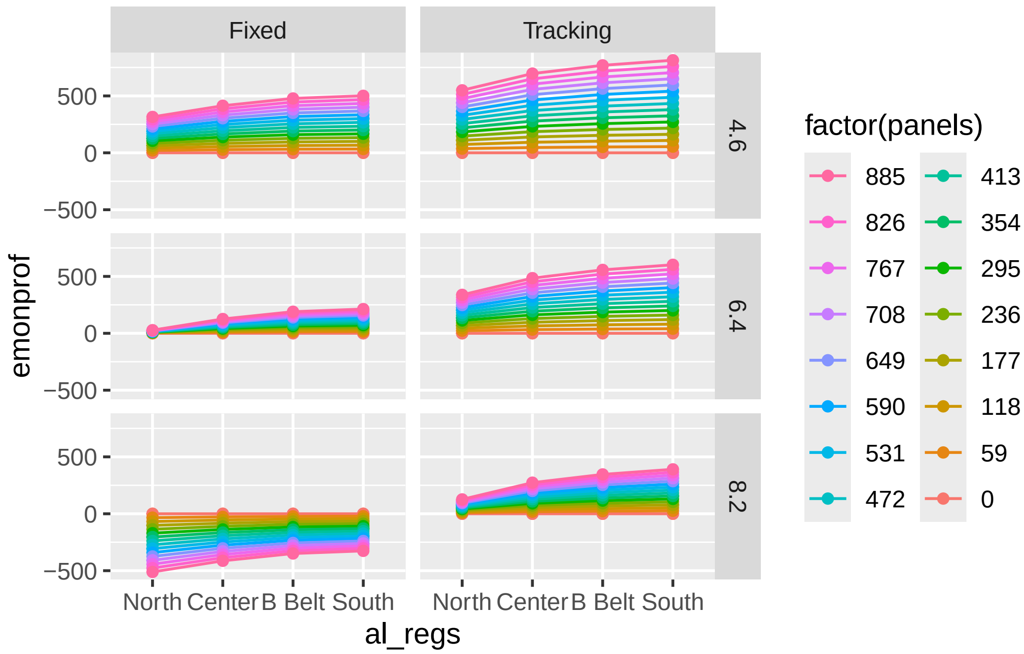

```
Electricity Price =  0.04
```



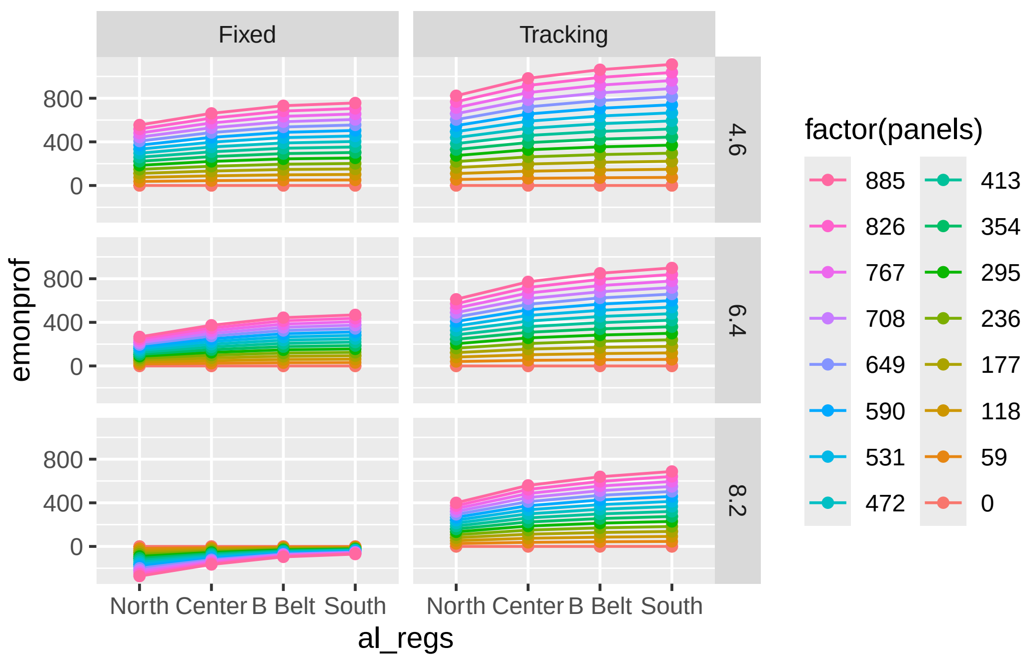

```
Electricity Price =  0.045
```



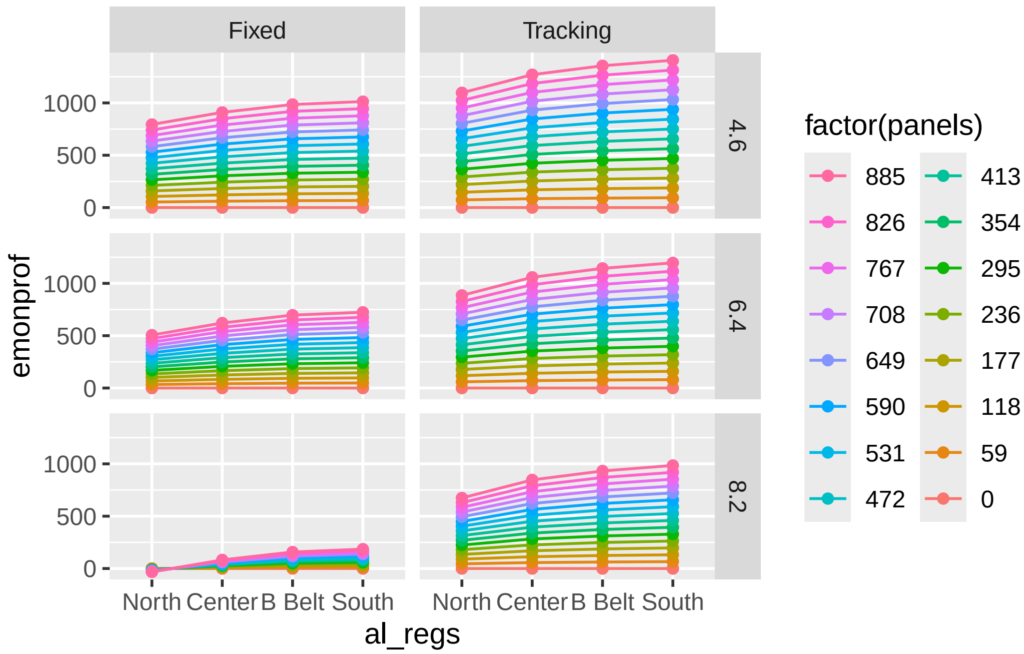

```
Electricity Price =  0.05
```



Electricity Price = 0.055



Electricity Price = 0.06



4 Profit from crops

4.1 Tomato

Filter return to operator, land and capital profit from Tomato:

```
tomato_profit = tomato %>%
  select(yldvar, yield,
         rolac17, rolac18, rolac19, rolac20,
         rolac21, rolac22, rolac23)
dim(tomato_profit)
```

```
[1] 21  9
```

```
tomato_profit
```

```
   yldvar yield     rolac17     rolac18     rolac19     rolac20     rolac21
3     2.0  2720  21679.3826  24399.3826  27119.3826  29839.3826  32559.3826
4     1.9  2584  20065.3826  22649.3826  25233.3826  27817.3826  30401.3826
5     1.8  2448  18451.3826  20899.3826  23347.3826  25795.3826  28243.3826
6     1.7  2312  16837.3826  19149.3826  21461.3826  23773.3826  26085.3826
```

```
7     1.6  2176  15223.3826  17399.3826  19575.3826  21751.3826  23927.3826
8     1.5  2040  13609.3826  15649.3826  17689.3826  19729.3826  21769.3826
9     1.4  1904  11995.3826  13899.3826  15803.3826  17707.3826  19611.3826
10    1.3  1768  10381.3826  12149.3826  13917.3826  15685.3826  17453.3826
11    1.2  1632   8767.3826  10399.3826  12031.3826  13663.3826  15295.3826
12    1.1  1496   7153.3826   8649.3826  10145.3826  11641.3826  13137.3826
13    1.0  1360   5539.3826   6899.3826   8259.3826   9619.3826  10979.3826
14    0.9  1224   3925.3826   5149.3826   6373.3826   7597.3826   8821.3826
15    0.8  1088   2311.3826   3399.3826   4487.3826   5575.3826   6663.3826
16    0.7   952    697.3826   1649.3826   2601.3826   3553.3826   4505.3826
17    0.6   816   -916.6174   -100.6174    715.3826   1531.3826   2347.3826
18    0.5   680  -2530.6174  -1850.6174  -1170.6174   -490.6174    189.3826
19    0.4   544  -4144.6174  -3600.6174  -3056.6174  -2512.6174  -1968.6174
20    0.3   408  -5758.6174  -5350.6174  -4942.6174  -4534.6174  -4126.6174
21    0.2   272  -7372.6174  -7100.6174  -6828.6174  -6556.6174  -6284.6174
22    0.1   136  -8986.6174  -8850.6174  -8714.6174  -8578.6174  -8442.6174
23    0.0     0 -10600.6174 -10600.6174 -10600.6174 -10600.6174 -10600.6174
        rolac22     rolac23
3   35279.3826  37999.3826
4   32985.3826  35569.3826
5   30691.3826  33139.3826
6   28397.3826  30709.3826
7   26103.3826  28279.3826
8   23809.3826  25849.3826
9   21515.3826  23419.3826
10  19221.3826  20989.3826
11  16927.3826  18559.3826
12  14633.3826  16129.3826
13  12339.3826  13699.3826
14  10045.3826  11269.3826
15   7751.3826   8839.3826
16   5457.3826   6409.3826
17   3163.3826   3979.3826
18    869.3826   1549.3826
19  -1424.6174   -880.6174
20  -3718.6174  -3310.6174
21  -6012.6174  -5740.6174
22  -8306.6174  -8170.6174
23 -10600.6174 -10600.6174
```

Convert data to long format:

```r
# Assign column names for clarity
colnames(tomato_profit) <- c("yldvar", "yield",
                  "rolac17", "rolac18", "rolac19",
                  "rolac20", "rolac21", "rolac22",
                  "rolac23")

# Reshape the data frame from wide to long format
tomato_long <- melt(tomato_profit,
                id.vars = c("yldvar", "yield"),
                measure.vars = c("rolac17", "rolac18", "rolac19",
                                 "rolac20", "rolac21", "rolac22",
                                 "rolac23"),
                variable.name = "price",
                value.name = "profit")

# Convert the 'Price' column to numeric by extracting the number
tomato_long$price <- as.numeric(gsub("rolac", "", tomato_long$price))

# View the resulting data frame
dim(tomato_long)
```

```
[1] 147   4
```

```r
str(tomato_long)
```

```
'data.frame':	147 obs. of  4 variables:
 $ yldvar: num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield : num  2720 2584 2448 2312 2176 ...
 $ price : num  17 17 17 17 17 17 17 17 17 17 ...
 $ profit: num  21679 20065 18451 16837 15223 ...
```

```r
head(tomato_long)
```

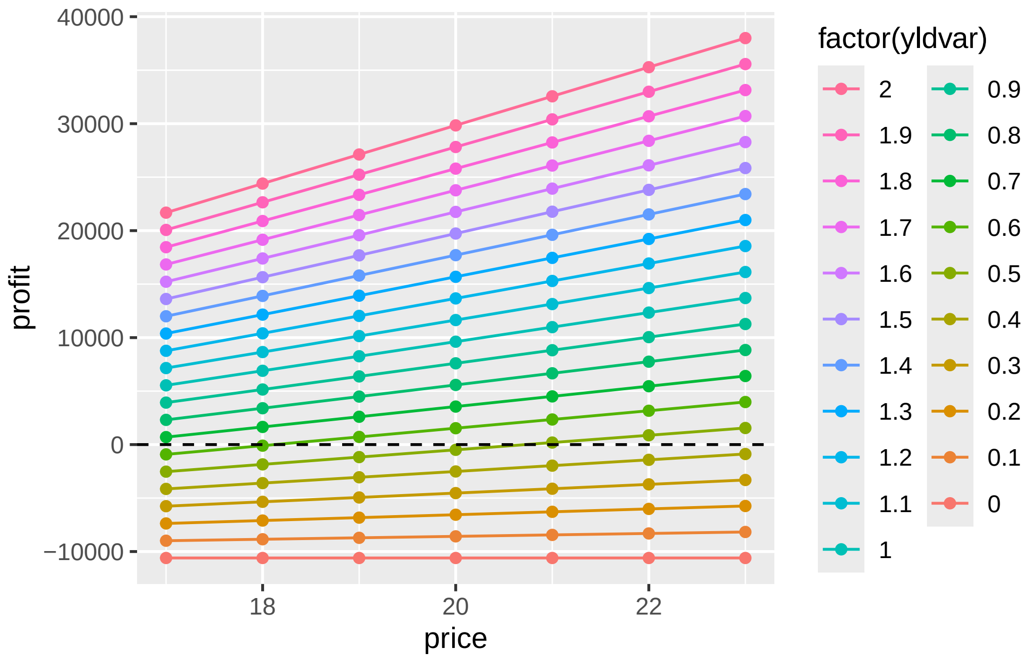
```
  yldvar yield price   profit
1    2.0  2720    17 21679.38
2    1.9  2584    17 20065.38
3    1.8  2448    17 18451.38
4    1.7  2312    17 16837.38
5    1.6  2176    17 15223.38
6    1.5  2040    17 13609.38
```

```r
tail(tomato_long)
```

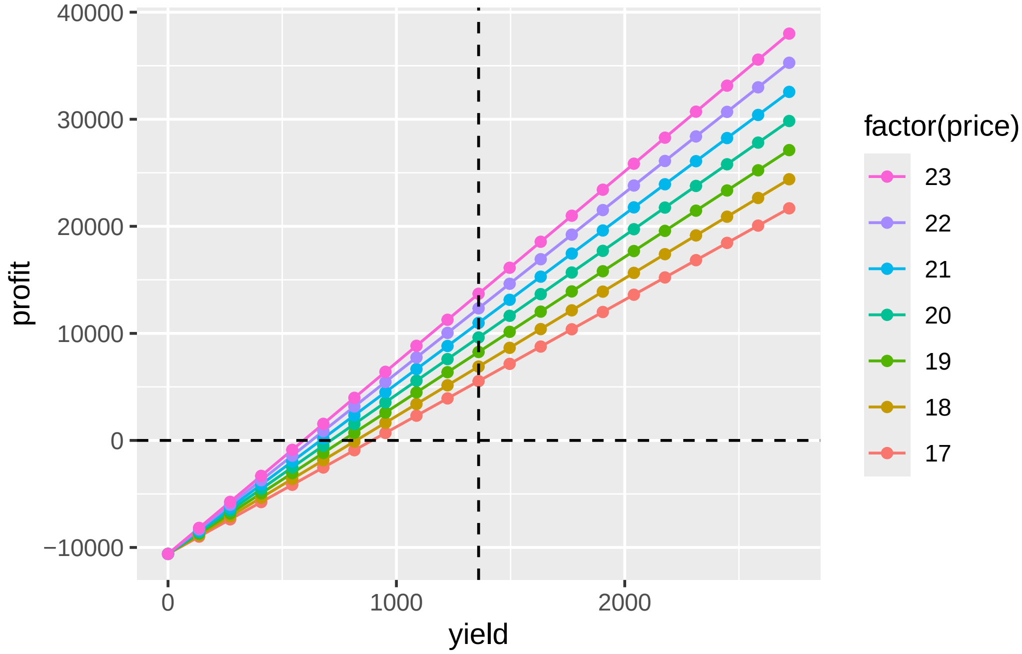
```
    yldvar yield price      profit
142    0.5   680    23   1549.3826
143    0.4   544    23   -880.6174
144    0.3   408    23  -3310.6174
145    0.2   272    23  -5740.6174
146    0.1   136    23  -8170.6174
147    0.0     0    23 -10600.6174
```

4.1.1 Plot Tomato Profit

```r
ggplot(data = tomato_long,
       mapping = aes(x = price,
                     y = profit,
                     color = factor(yldvar),
                     group = factor(yield))) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  guides(color = guide_legend(ncol = 2,
                              reverse = TRUE))
```




```
ggplot(data = tomato_long,
       mapping = aes(x = yield,
                     y = profit,
                     #fill = yield,
                     color = factor(price),
                     group = factor(price))) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  # Vertical dashed line is 100% yield
  geom_vline(xintercept = tomato_long$yield[11],
             linetype = "dashed",
             color = "black") +
guides(color = guide_legend(reverse = TRUE))
```



4.2 Strawberry

Filter return to operator, land and capital profit from strawberry

```
strawberry_profit = strawberry %>%
  select(yldvar, yield,
         rolac3, rolac4, rolac5, rolac6,
         rolac7, rolac8, rolac9)
dim(strawberry_profit)
```

```
[1] 21  9
```

```
strawberry_profit
```

```
   yldvar  yield     rolac3      rolac4     rolac5     rolac6     rolac7
3     2.0 6150.0  -1740.495  4409.50503  10559.505  16709.505  22859.505
4     1.9 5842.5  -2317.350  3525.15003   9367.650  15210.150  21052.650
5     1.8 5535.0  -2894.205  2640.79503   8175.795  13710.795  19245.795
6     1.7 5227.5  -3471.060  1756.44003   6983.940  12211.440  17438.940
7     1.6 4920.0  -4047.915   872.08503   5792.085  10712.085  15632.085
8     1.5 4612.5  -4624.770   -12.26997   4600.230   9212.730  13825.230
```

```
9     1.4 4305.0  -5201.625   -896.62497   3408.375   7713.375  12018.375
10    1.3 3997.5  -5778.480  -1780.97997   2216.520   6214.020  10211.520
11    1.2 3690.0  -6355.335  -2665.33497   1024.665   4714.665   8404.665
12    1.1 3382.5  -6932.190  -3549.68997   -167.190   3215.310   6597.810
13    1.0 3075.0  -7509.045  -4434.04497  -1359.045   1715.955   4790.955
14    0.9 2767.5  -8085.900  -5318.39997  -2550.900    216.600   2984.100
15    0.8 2460.0  -8662.755  -6202.75497  -3742.755  -1282.755   1177.245
16    0.7 2152.5  -9239.610  -7087.10997  -4934.610  -2782.110   -629.610
17    0.6 1845.0  -9816.465  -7971.46497  -6126.465  -4281.465  -2436.465
18    0.5 1537.5 -10393.320  -8855.81997  -7318.320  -5780.820  -4243.320
19    0.4 1230.0 -10970.175  -9740.17497  -8510.175  -7280.175  -6050.175
20    0.3  922.5 -11547.030 -10624.52997  -9702.030  -8779.530  -7857.030
21    0.2  615.0 -12123.885 -11508.88497 -10893.885 -10278.885  -9663.885
22    0.1  307.5 -12700.740 -12393.23997 -12085.740 -11778.240 -11470.740
23    0.0    0.0 -13277.595 -13277.59497 -13277.595 -13277.595 -13277.595
       rolac8     rolac9
3   29009.505  35159.505
4   26895.150  32737.650
5   24780.795  30315.795
6   22666.440  27893.940
7   20552.085  25472.085
8   18437.730  23050.230
9   16323.375  20628.375
10  14209.020  18206.520
11  12094.665  15784.665
12   9980.310  13362.810
13   7865.955  10940.955
14   5751.600   8519.100
15   3637.245   6097.245
16   1522.890   3675.390
17   -591.465   1253.535
18  -2705.820  -1168.320
19  -4820.175  -3590.175
20  -6934.530  -6012.030
21  -9048.885  -8433.885
22 -11163.240 -10855.740
23 -13277.595 -13277.595
```

Convert data to long format:

```r
# Assign column names for clarity
colnames(strawberry_profit) <- c("yldvar", "yield",
```

```r
                  "rolac3", "rolac4", "rolac5",
                  "rolac6", "rolac7", "rolac8",
                  "rolac9")
# Reshape the data frame from wide to long format
stberry_long <- melt(strawberry_profit,
                id.vars = c("yldvar", "yield"),
                measure.vars = c("rolac3", "rolac4", "rolac5",
                                 "rolac6", "rolac7", "rolac8",
                                 "rolac9"),
                variable.name = "price",
                value.name = "profit")

# Convert the 'Price' column to numeric by extracting the number
stberry_long$price <- as.numeric(gsub("rolac", "", stberry_long$price))

# View the resulting data frame
dim(stberry_long)
```

```
[1] 147   4
```

```r
str(stberry_long)
```

```
'data.frame':	147 obs. of  4 variables:
 $ yldvar: num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield : num  6150 5842 5535 5228 4920 ...
 $ price : num  3 3 3 3 3 3 3 3 3 3 ...
 $ profit: num  -1740 -2317 -2894 -3471 -4048 ...
```

```r
head(stberry_long)
```

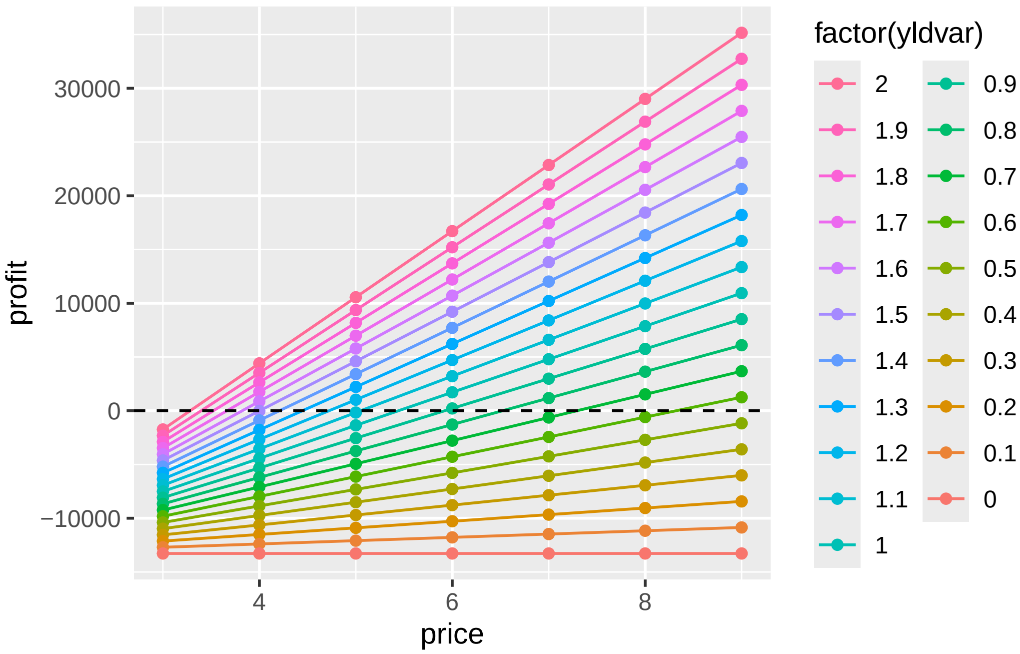
```
  yldvar  yield price    profit
1    2.0 6150.0     3 -1740.495
2    1.9 5842.5     3 -2317.350
3    1.8 5535.0     3 -2894.205
4    1.7 5227.5     3 -3471.060
5    1.6 4920.0     3 -4047.915
6    1.5 4612.5     3 -4624.770
```

```r
tail(stberry_long)
```

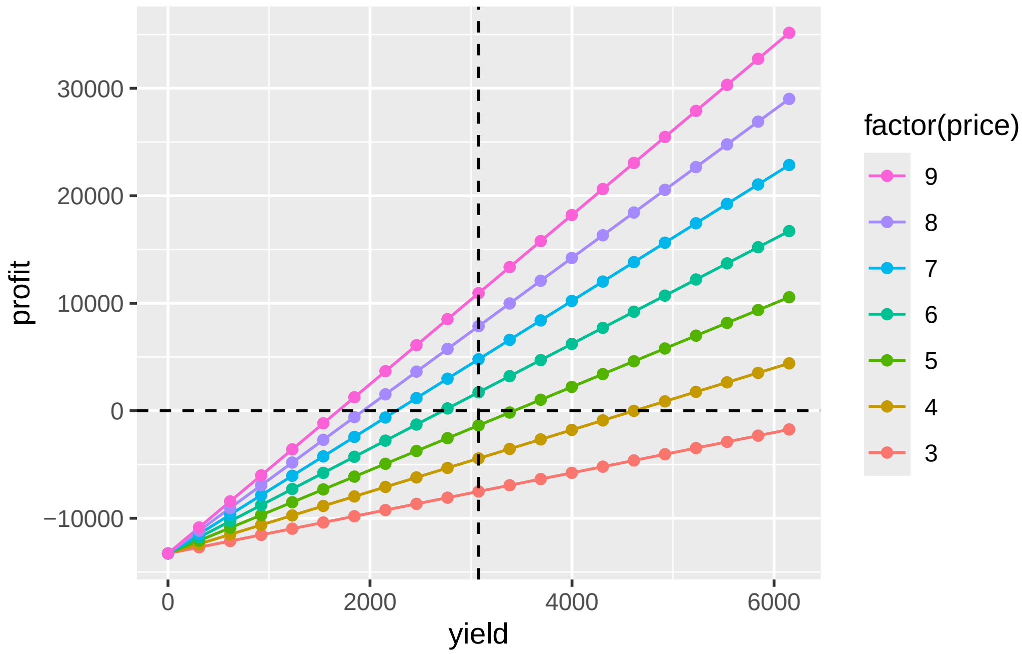
```
    yldvar  yield price     profit
142    0.5 1537.5     9  -1168.320
143    0.4 1230.0     9  -3590.175
144    0.3  922.5     9  -6012.030
145    0.2  615.0     9  -8433.885
146    0.1  307.5     9 -10855.740
147    0.0    0.0     9 -13277.595
```

4.2.1 Plot Strawberry Profit

```r
ggplot(data = stberry_long,
       mapping = aes(x = price,
                     y = profit,
                     color = factor(yldvar),
                     group = factor(yield))) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  guides(color = guide_legend(ncol = 2,
                              reverse = TRUE))
```




```
ggplot(data = stberry_long,
       mapping = aes(x = yield,
                     y = profit,
                     color = factor(price),
                     group = factor(price))) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  #Vertical dashed line is 100% yield
  geom_vline(xintercept = stberry_long$yield[11],
             linetype = "dashed",
             color = "black") +
  guides(color = guide_legend(reverse = TRUE))
```



4.3 Squash

Filter return to operator, land and capital profit from squash

```
squash_profit = squash %>%
  select(yldvar, yield,
         rolac11, rolac12, rolac13, rolac14,
         rolac15, rolac16, rolac17)
squash_profit
```

```
   yldvar yield   rolac11   rolac12     rolac13   rolac14   rolac15     rolac16
3     2.0  2180 10309.117 12489.117 14669.11702 16849.117 19029.117 21209.11702
4     1.9  2071  9607.367 11678.367 13749.36702 15820.367 17891.367 19962.36702
5     1.8  1962  8905.617 10867.617 12829.61702 14791.617 16753.617 18715.61702
6     1.7  1853  8203.867 10056.867 11909.86702 13762.867 15615.867 17468.86702
7     1.6  1744  7502.117  9246.117 10990.11702 12734.117 14478.117 16222.11702
8     1.5  1635  6800.367  8435.367 10070.36702 11705.367 13340.367 14975.36702
9     1.4  1526  6098.617  7624.617  9150.61702 10676.617 12202.617 13728.61702
10    1.3  1417  5396.867  6813.867  8230.86702  9647.867 11064.867 12481.86702
11    1.2  1308  4695.117  6003.117  7311.11702  8619.117  9927.117 11235.11702
12    1.1  1199  3993.367  5192.367  6391.36702  7590.367  8789.367  9988.36702
13    1.0  1090  3291.617  4381.617  5471.61702  6561.617  7651.617  8741.61702
```

```
14    0.9   981  2589.867  3570.867  4551.86702  5532.867  6513.867  7494.86702
15    0.8   872  1888.117  2760.117  3632.11702  4504.117  5376.117  6248.11702
16    0.7   763  1186.367  1949.367  2712.36702  3475.367  4238.367  5001.36702
17    0.6   654   484.617  1138.617  1792.61702  2446.617  3100.617  3754.61702
18    0.5   545  -217.133   327.867   872.86702  1417.867  1962.867  2507.86702
19    0.4   436  -918.883  -482.883   -46.88298   389.117   825.117  1261.11702
20    0.3   327 -1620.633 -1293.633  -966.63298  -639.633  -312.633    14.36702
21    0.2   218 -2322.383 -2104.383 -1886.38298 -1668.383 -1450.383 -1232.38298
22    0.1   109 -3024.133 -2915.133 -2806.13298 -2697.133 -2588.133 -2479.13298
23    0.0     0 -3725.883 -3725.883 -3725.88298 -3725.883 -3725.883 -3725.88298
     rolac17
3  23389.117
4  22033.367
5  20677.617
6  19321.867
7  17966.117
8  16610.367
9  15254.617
10 13898.867
11 12543.117
12 11187.367
13  9831.617
14  8475.867
15  7120.117
16  5764.367
17  4408.617
18  3052.867
19  1697.117
20   341.367
21 -1014.383
22 -2370.133
23 -3725.883
```

Convert data to long format:

```r
# Assign column names for clarity
colnames(squash_profit) <- c("yldvar", "yield",
                  "rolac11", "rolac12", "rolac13",
                  "rolac14", "rolac15", "rolac16",
                  "rolac17")

# Reshape the data frame from wide to long format
```

```r
squash_long <- melt(squash_profit,
                id.vars = c("yldvar", "yield"),
                measure.vars = c("rolac11", "rolac12", "rolac13",
                                 "rolac14", "rolac15", "rolac16",
                                 "rolac17"),
                variable.name = "price",
                value.name = "profit")

# Convert the 'Price' column to numeric by extracting the number
squash_long$price <- as.numeric(gsub("rolac", "", squash_long$price))

# View the resulting data frame
dim(squash_long)
```

```
[1] 147   4
```

```r
str(squash_long)
```

```
'data.frame':   147 obs. of  4 variables:
 $ yldvar: num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield : num  2180 2071 1962 1853 1744 ...
 $ price : num  11 11 11 11 11 11 11 11 11 11 ...
 $ profit: num  10309 9607 8906 8204 7502 ...
```

```r
head(squash_long)
```

```
  yldvar yield price    profit
1    2.0  2180    11 10309.117
2    1.9  2071    11  9607.367
3    1.8  1962    11  8905.617
4    1.7  1853    11  8203.867
5    1.6  1744    11  7502.117
6    1.5  1635    11  6800.367
```

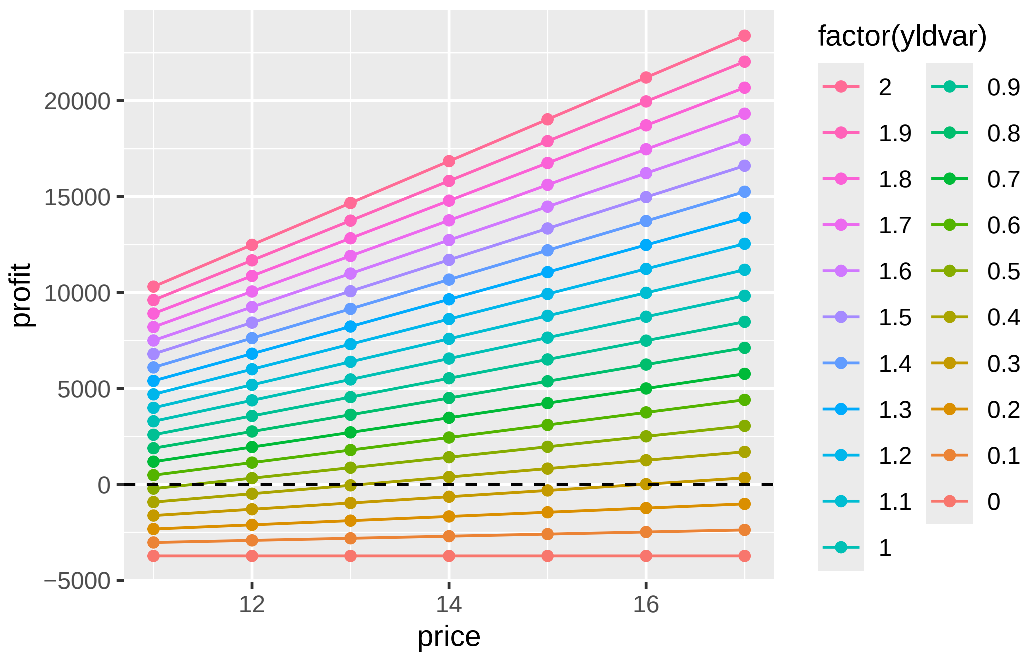
```r
tail(squash_long)
```

```
    yldvar yield price    profit
142    0.5   545    17  3052.867
143    0.4   436    17  1697.117
```

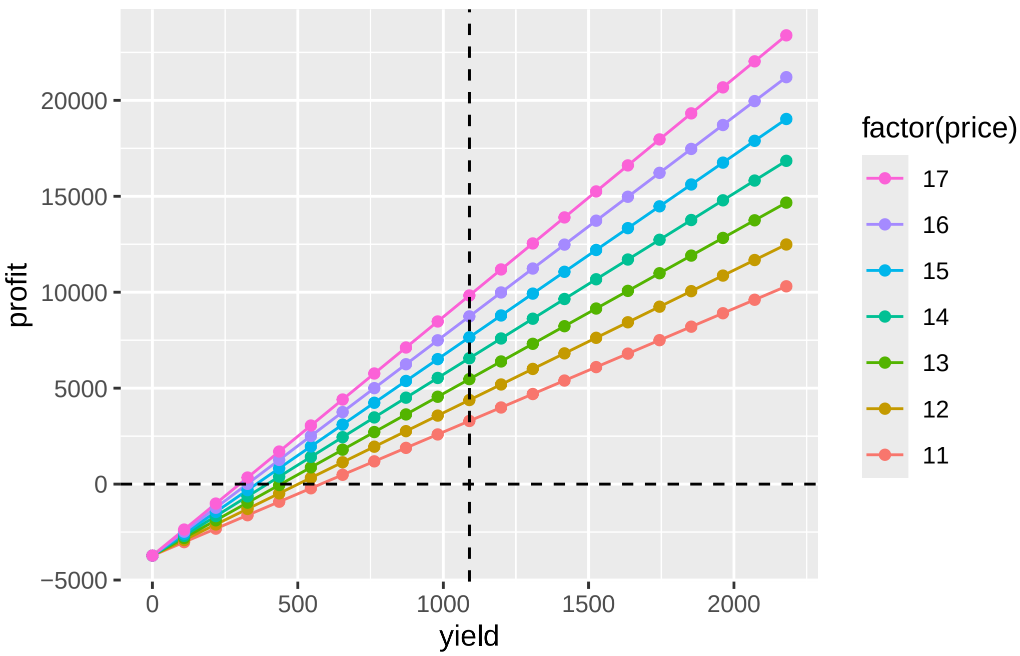
```
144     0.3    327     17    341.367
145     0.2    218     17 -1014.383
146     0.1    109     17 -2370.133
147     0.0      0     17 -3725.883
```

4.3.1 Profit from squash:

```
ggplot(data = squash_long,
       mapping = aes(x = price,
                     y = profit,
                     color = factor(yldvar),
                     group = factor(yield))) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  guides(color = guide_legend(ncol = 2,
                              reverse = TRUE))
```



```
ggplot(data = squash_long,
       mapping = aes(x = yield,
                     y = profit,
                     color = factor(price),
                     group = factor(price))) +
  geom_line() +
  geom_point() +
  geom_hline(yintercept = 0,
             linetype = "dashed",
             color = "black") +
  # Vertical dashed line is 100% yield
    geom_vline(xintercept = squash_long$yield[11],
             linetype = "dashed",
             color = "black") +
  guides(color = guide_legend(reverse = TRUE))
```



5 Profit from agrivoltaics

Total profit from solar and crops for all combinations of AVs simulated.

5.1 Profit from tomato agrivoltaic system

- Joint profit from tomato (tomato_long) and solar energy production (solar_profit) from 1 acre of land.
- The last variable (tav_profit) is the final profit from tomato agrivoltaic system which is the result of our interest.

```
# Calculate all combinations of rows from both matrices in a vectorized way
solar_expanded <- solar_profit[rep(1:nrow(solar_profit),
                                   each = nrow(tomato_long)), ]
tomato_expanded <- tomato_long[rep(1:nrow(tomato_long),
                                   times = nrow(solar_profit)), ]

# Calculate the new column for tav_profit directly
tav_profit_values <- solar_expanded$eannprof + tomato_expanded$profit

# Combine the matrices and the calculated tav_profit column
tav_profit <- cbind(solar_expanded,
                    tomato_expanded,
                    tav_profit = tav_profit_values)

# Convert to a data frame and ensure the correct format
tav_profit <- as.data.frame(tav_profit)
tav_profit <- data.frame(lapply(tav_profit, unlist))

# Inspect the structure and data
dim(tav_profit)
```

```
[1] 814968     24
```

```
str(tav_profit)
```

```
'data.frame':   814968 obs. of  24 variables:
 $ sprop     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ al_regs   : chr  "Black Belt" "Black Belt" "Black Belt" "Black Belt" ...
 $ array     : chr  "Fixed" "Fixed" "Fixed" "Fixed" ...
 $ dc_kw     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ panels    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ energy    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ elcprc    : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...
 $ elcrev    : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
 $ height    : num  4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 ...
 $ capex     : num  1.59 1.59 1.59 1.59 1.59 ...
 $ landlease : num  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...
 $ opex      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ttlcost   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ recredit  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ anncost   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ moncost   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eprofit   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eannprof  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ emonprof  : num  0 0 0 0 0 0 0 0 0 0 ...
 $ yldvar    : num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield     : num  2720 2584 2448 2312 2176 ...
 $ price     : num  17 17 17 17 17 17 17 17 17 17 ...
 $ profit    : num  21679 20065 18451 16837 15223 ...
 $ tav_profit: num  21679 20065 18451 16837 15223 ...
```

```r
head(tav_profit)
```

```
  sprop    al_regs array dc_kw panels energy elcprc elcrev height    capex
1     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
2     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
3     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
4     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
5     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
6     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
  landlease opex ttlcost recredit anncost moncost eprofit eannprof emonprof
1      1000    0       0        0       0       0       0        0        0
2      1000    0       0        0       0       0       0        0        0
3      1000    0       0        0       0       0       0        0        0
4      1000    0       0        0       0       0       0        0        0
5      1000    0       0        0       0       0       0        0        0
6      1000    0       0        0       0       0       0        0        0
  yldvar yield price   profit tav_profit
1    2.0  2720    17 21679.38   21679.38
2    1.9  2584    17 20065.38   20065.38
3    1.8  2448    17 18451.38   18451.38
4    1.7  2312    17 16837.38   16837.38
5    1.6  2176    17 15223.38   15223.38
6    1.5  2040    17 13609.38   13609.38
```

```r
tail(tav_profit)
```

```
       sprop  al_regs    array  dc_kw panels energy elcprc   elcrev height
814963     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814964     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814965     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814966     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814967     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814968     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
        capex landlease     opex ttlcost recredit anncost  moncost   eprofit
814963  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814964  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814965  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814966  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814967  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814968  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
        eannprof emonprof yldvar yield price      profit tav_profit
814963 -18611.56 984.5974    0.5   680    23   1549.3826  -17062.17
814964 -18611.56 984.5974    0.4   544    23   -880.6174  -19492.17
814965 -18611.56 984.5974    0.3   408    23  -3310.6174  -21922.17
814966 -18611.56 984.5974    0.2   272    23  -5740.6174  -24352.17
814967 -18611.56 984.5974    0.1   136    23  -8170.6174  -26782.17
814968 -18611.56 984.5974    0.0     0    23 -10600.6174  -29212.17
```

5.1.1 Saving results locally

```r
#write_csv(tav_profit, "tav_profit.csv")
write_feather(tav_profit,
  sink = "Data/tav_profit.feather",
  version = 2,
  chunk_size = 65536L,
  compression = c("default"),
  #compression = c("default", "lz4", "lz4_frame", "uncompressed", "zstd"),
  compression_level = NULL
)
```

5.2 Profit from strawberry agrivoltaic system

- Joint profit from strawberry (stberry_long) and solar energy production (solar_profit) from 1 acre of land.

- The last variable (sbav_profit) is the final profit from strawberry agrivoltaic system which is the result of our interest.

```
# Generate all combinations of rows from both matrices in a vectorized way
solar_expanded <- solar_profit[rep(1:nrow(solar_profit),
                                   each = nrow(stberry_long)), ]
stberry_expanded <- stberry_long[rep(1:nrow(stberry_long),
                                     times = nrow(solar_profit)), ]

# Calculate the new column for sbav_profit directly
sbav_profit_values <- solar_expanded$eannprof + stberry_expanded$profit

# Combine the matrices and the calculated sbav_profit column
sbav_profit <- cbind(solar_expanded,
                     stberry_expanded,
                     sbav_profit = sbav_profit_values)

# Convert to a data frame and ensure the correct format
sbav_profit <- as.data.frame(sbav_profit)
sbav_profit <- data.frame(lapply(sbav_profit, unlist))

# Inspect the structure and data
dim(sbav_profit)
```

```
[1] 814968     24
```

```
str(sbav_profit)
```

```
'data.frame':   814968 obs. of  24 variables:
 $ sprop      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ al_regs    : chr  "Black Belt" "Black Belt" "Black Belt" "Black Belt" ...
 $ array      : chr  "Fixed" "Fixed" "Fixed" "Fixed" ...
 $ dc_kw      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ panels     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ energy     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ elcprc     : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...
 $ elcrev     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ height     : num  4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 ...
 $ capex      : num  1.59 1.59 1.59 1.59 1.59 ...
 $ landlease  : num  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...
 $ opex       : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
 $ ttlcost    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ recredit   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ anncost    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ moncost    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eprofit    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eannprof   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ emonprof   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ yldvar     : num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield      : num  6150 5842 5535 5228 4920 ...
 $ price      : num  3 3 3 3 3 3 3 3 3 3 ...
 $ profit     : num  -1740 -2317 -2894 -3471 -4048 ...
 $ sbav_profit: num  -1740 -2317 -2894 -3471 -4048 ...
```

```
head(sbav_profit)
```

```
  sprop    al_regs array dc_kw panels energy elcprc elcrev height    capex
1     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
2     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
3     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
4     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
5     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
6     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
  landlease opex ttlcost recredit anncost moncost eprofit eannprof emonprof
1      1000    0       0        0       0       0       0        0        0
2      1000    0       0        0       0       0       0        0        0
3      1000    0       0        0       0       0       0        0        0
4      1000    0       0        0       0       0       0        0        0
5      1000    0       0        0       0       0       0        0        0
6      1000    0       0        0       0       0       0        0        0
  yldvar  yield price    profit sbav_profit
1    2.0 6150.0     3 -1740.495   -1740.495
2    1.9 5842.5     3 -2317.350   -2317.350
3    1.8 5535.0     3 -2894.205   -2894.205
4    1.7 5227.5     3 -3471.060   -3471.060
5    1.6 4920.0     3 -4047.915   -4047.915
6    1.5 4612.5     3 -4624.770   -4624.770
```

```
tail(sbav_profit)
```

```
       sprop  al_regs    array  dc_kw panels energy elcprc   elcrev height
814963     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
```

```
814964      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814965      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814966      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814967      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814968      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
       capex landlease     opex ttlcost recredit anncost  moncost   eprofit
814963  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814964  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814965  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814966  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814967  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814968  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
        eannprof emonprof yldvar  yield price     profit sbav_profit
814963 -18611.56 984.5974    0.5 1537.5     9  -1168.320   -19779.88
814964 -18611.56 984.5974    0.4 1230.0     9  -3590.175   -22201.73
814965 -18611.56 984.5974    0.3  922.5     9  -6012.030   -24623.59
814966 -18611.56 984.5974    0.2  615.0     9  -8433.885   -27045.44
814967 -18611.56 984.5974    0.1  307.5     9 -10855.740   -29467.30
814968 -18611.56 984.5974    0.0    0.0     9 -13277.595   -31889.15
```

5.2.1 Saving results locally

```r
#write_csv(sbav_profit, "tav_profit.csv")
write_feather(sbav_profit,
  sink = "Data/sbav_profit.feather",
  version = 2,
  chunk_size = 65536L,
  compression = c("default"),
  #compression = c("default", "lz4", "lz4_frame", "uncompressed", "zstd"),
  compression_level = NULL
)
```

5.3 Profit from squash agrivoltaic system

- Joint profit from squash (squash_long) and solar energy production (solar_profit) from 1 acre of land.
- The last variable (sqav_profit) is the final profit from squash agrivoltaic system which is the result of our interest.

```r
# Efficient calculation of all combinations of rows from both matrices
solar_expanded <- solar_profit[rep(1:nrow(solar_profit),
                                   each = nrow(squash_long)), ]
squash_expanded <- squash_long[rep(1:nrow(squash_long),
                                   times = nrow(solar_profit)), ]

# Calculate the new column for sqav_profit directly
sqav_profit_values <- solar_expanded$eannprof + squash_expanded$profit

# Combine the matrices and the calculated sqav_profit column
sqav_profit <- cbind(solar_expanded,
                     squash_expanded,
                     sqav_profit = sqav_profit_values)

# Convert to a data frame and ensure the correct format
sqav_profit <- as.data.frame(sqav_profit)
sqav_profit <- data.frame(lapply(sqav_profit, unlist))

# Inspect the structure and data
dim(sqav_profit)
```

```
[1] 814968     24
```

```r
str(sqav_profit)
```

```
'data.frame':	814968 obs. of  24 variables:
 $ sprop      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ al_regs    : chr  "Black Belt" "Black Belt" "Black Belt" "Black Belt" ...
 $ array      : chr  "Fixed" "Fixed" "Fixed" "Fixed" ...
 $ dc_kw      : num  0 0 0 0 0 0 0 0 0 0 ...
 $ panels     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ energy     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ elcprc     : num  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 ...
 $ elcrev     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ height     : num  4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 4.6 ...
 $ capex      : num  1.59 1.59 1.59 1.59 1.59 ...
 $ landlease  : num  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...
 $ opex       : num  0 0 0 0 0 0 0 0 0 0 ...
 $ ttlcost    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ recredit   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ anncost    : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
 $ moncost    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eprofit    : num  0 0 0 0 0 0 0 0 0 0 ...
 $ eannprof   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ emonprof   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ yldvar     : num  2 1.9 1.8 1.7 1.6 1.5 1.4 1.3 1.2 1.1 ...
 $ yield      : num  2180 2071 1962 1853 1744 ...
 $ price      : num  11 11 11 11 11 11 11 11 11 11 ...
 $ profit     : num  10309 9607 8906 8204 7502 ...
 $ sqav_profit: num  10309 9607 8906 8204 7502 ...
```

```
head(sqav_profit)
```

```
  sprop    al_regs array dc_kw panels energy elcprc elcrev height    capex
1     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
2     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
3     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
4     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
5     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
6     0 Black Belt Fixed     0      0      0   0.01      0    4.6 1.593333
  landlease opex ttlcost recredit anncost moncost eprofit eannprof emonprof
1      1000    0       0        0       0       0       0        0        0
2      1000    0       0        0       0       0       0        0        0
3      1000    0       0        0       0       0       0        0        0
4      1000    0       0        0       0       0       0        0        0
5      1000    0       0        0       0       0       0        0        0
6      1000    0       0        0       0       0       0        0        0
  yldvar yield price    profit sqav_profit
1    2.0  2180    11 10309.117   10309.117
2    1.9  2071    11  9607.367    9607.367
3    1.8  1962    11  8905.617    8905.617
4    1.7  1853    11  8203.867    8203.867
5    1.6  1744    11  7502.117    7502.117
6    1.5  1635    11  6800.367    6800.367
```

```
tail(sqav_profit)
```

```
       sprop  al_regs    array  dc_kw panels energy elcprc    elcrev height
814963     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814964     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814965     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814966     1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
```

```
814967      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
814968      1 Southern Tracking 423.74    885 712873   0.06 42772.38    8.2
       capex landlease      opex ttlcost recredit anncost  moncost   eprofit
814963  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814964  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814965  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814966  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814967  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
814968  2.11      1000 7234.875 1017706 4704.962 66088.9 2971.848 -974933.4
        eannprof emonprof yldvar yield price    profit sqav_profit
814963 -18611.56 984.5974    0.5   545    17  3052.867   -15558.69
814964 -18611.56 984.5974    0.4   436    17  1697.117   -16914.44
814965 -18611.56 984.5974    0.3   327    17   341.367   -18270.19
814966 -18611.56 984.5974    0.2   218    17 -1014.383   -19625.94
814967 -18611.56 984.5974    0.1   109    17 -2370.133   -20981.69
814968 -18611.56 984.5974    0.0     0    17 -3725.883   -22337.44
```

5.3.1 Saving results locally

```r
#write_csv(sqav_profit, "tav_profit.csv")
write_feather(sqav_profit,
  sink = "Data/sqav_profit.feather",
  version = 2,
  chunk_size = 65536L,
  compression = c("default"),
  #compression = c("default", "lz4", "lz4_frame", "uncompressed", "zstd"),
  compression_level = NULL
)
```