FINANCIAL ANALYTICS IN R

Putting It All Together

Emily Riederer
Instructor

A Quick Recap

- Business Models
- Cashflow Calculations
- Profitability Metrics

A Caffeinated Case Study

Project Valuations

- Treat project as a mini-business
- Think about *incremental* cashflows
- Ignore sunk costs
- Remember to value the side effects

Coffee-nomics

Investment: Nitro kegerator (dispenser)

Incremental Revenue:

- Nitro coffee sales
- Sales of additional items by incremental customers

Coffee-nomics

Investment: Nitro kegerator (dispenser)

Incremental Expenses:

- Coffee kegs
- Additional labor
- Machine maintenance
- Cannibalization

The balancing act

- Equal parts art and science
- Many possible levels of detail

FINANCIAL ANALYTICS IN R

Let's practice!

FINANCIAL ANALYTICS IN R

Asking What If?

Emily Riederer
Instructor

But what if...?

...we have other project ideas?

- Expand to offer brunch menu
- Open a new location
- Build better loyalty program

...things happen that are out of our control?

- Cost of coffee goes up
- Competition drives down prices
- Only sales are from cannibalization

Scenario (What-If) Analysis

- Alter model assumptions to compare different outcomes
  - Alternative projects
  - Exogenous circumstances (optimistic, realistic, pessimistic)

The mechanical route

```
scenario1 <- mutate(assumptions, var1 = 1.2 * var1)
cashflow1 <- calc_model(scenario1)
calc_npv(cashflow1)
```

```
scenario2 <- mutate(assumptions, var1 = 1.5 * var1, var2 = 0.8 * var2)
cashflow2 <- calc_model(scenario2)
calc_npv(cashflow2)
```

```
# etc...
```

Tidying up

```
library(purrr)
library(tidyr)

all_scenarios
```

scenario	var1	var2	var3
'scenario1'	1	5	7
'scenario1'	2	4	8
'scenario1'	3	10	12
'scenario2'	1	15	14
'scenario2'	2	14	16
'scenario2'	3	20	24

Tidying up

```
library(purrr)
library(tidyr)

all_scenarios %>%
    nest(-scenario)
```

scenario	data
'scenario1'	<tibble [3x3]>
'scenario2'	<tibble [3x3]>

Tidying up

```
library(purrr)
library(tidyr)

all_scenarios %>%
    nest(-scenario) %>%
    mutate( cashflow = map_df( data, calc_model) )
```

scenario	data	cashflow
'scenario1'	<tibble [3x3]>	calc_model(scenario1 data)
'scenario2'	<tibble [3x3]>	calc_model(scenario2 data)

Tidying up

```
library(purrr)
library(tidyr)

all_scenarios %>%
    nest(-scenario) %>%
    mutate( cashflow = map_df( data, calc_model) )  %>%
    mutate( npv = map_dbl( cashflow, calc_npv) )
```

scenario	data	cashflow	npv
'scenario1'	<tibble [3x3]>	<tibble [3x3]>	calc_npv(scenario1 cashflow)
'scenario2'	<tibble [3x3]>	<tibble [3x3]>	calc_npv(scenario2 cashflow)

FINANCIAL ANALYTICS IN R

Your Turn!

FINANCIAL ANALYTICS IN R

Sensitivity Analysis

Emily Riederer
Instructor

Sensitivity analysis in R

```
sensitivity <-
    expand.grid(
      factor = c(0.5, 1, 1.5),
      metric = c("vbl1", "vbl2")
    )
```

factor	metric
0.5	'vbl1'
1	'vbl1'
1.5	'vbl1'
0.5	'vbl2'
1	'vbl2'
1.5	'vbl2'

Sensitivity analysis in R

```
sensitivity <-
    expand.grid(
      factor = c(0.5, 1, 1.5),
      metric = c("vbl1", "vbl2")
  )
```

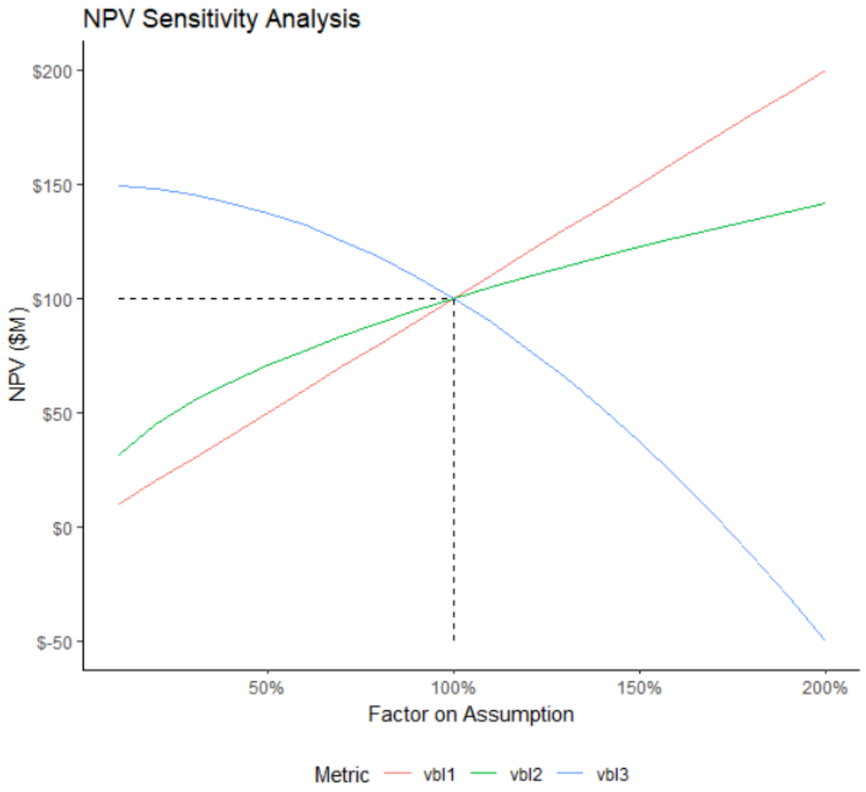
factor	metric	<what we want is...>
0.5	'vbl1'	valuation after assumption 'vbl1' is multiplied by 0.5
1	'vbl1'	valuation after assumption 'vbl1' is multiplied by 1
1.5	'vbl1'	valuation after assumption 'vbl1' is multiplied by 1.5
0.5	'vbl2'	valuation after assumption 'vbl2' is multiplied by 0.5
1	'vbl2'	valuation after assumption 'vbl2' is multiplied by 1
1.5	'vbl2'	valuation after assumption 'vbl2' is multiplied by 1.5

Sensitivity analysis in R

```
sensitivity <-
    expand.grid(
      factor = c(0.5, 1, 1.5),
      metric = c("vbl1", "vbl2")
    ) %>%
    mutate(scenario = map2(metric,factor,~factor_data(assumptions, .x, .y)))
```

factor	metric	scenario
0.5	'vbl1'	factor_data(assumptions, metric, factor)
1	'vbl1'	factor_data(assumptions, metric, factor)
1.5	'vbl1'	factor_data(assumptions, metric, factor)
0.5	'vbl2'	factor_data(assumptions, metric, factor)
1	'vbl2'	factor_data(assumptions, metric, factor)
1.5	'vbl2'	factor_data(assumptions, metric, factor)

Visualizing sensitivity



Sensitivity Plots

- summarize "information overload"
- highlight relative magnitudes
- reveal non-linearities
- emphasize univariate nature of analysis

Cautions with sensitivity analysis

- Only looking at univariate changes, but errors are often correlated
- Not considering variance of estimates or different likelihoods of being off by a certain percent

FINANCIAL ANALYTICS IN R

Let's practice!

FINANCIAL ANALYTICS IN R

Cashflow Visualization & Communication

Emily Riederer
Instructor

Long versus wide cashflows

Long data is tidy data

- 1 column per metric
- 1 row per observation

Month	Received	Spent	Net
1	100	150	-50
2	200	175	25
3	300	200	100
4	400	225	175
5	500	250	250
6	500	250	250

Cashflows are wide data

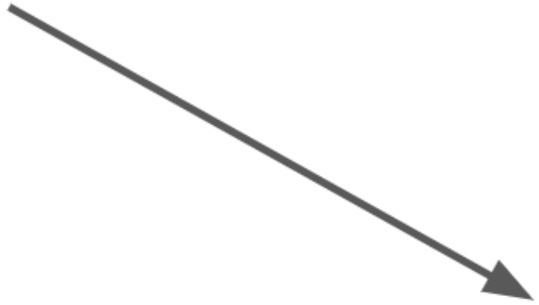
- 1 column per unit of time
- 1 row per metric

Month	1	2	3	4	5	6
Cash Received	100.00	200.00	300.00	400.00	500.00	500.00
Cash Spent	150.00	175.00	200.00	225.00	250.00	250.00
Net Cash	(50.00)	25.00	100.00	175.00	250.00	250.00

Tidying a cashflow (wide to long)

```
library(tidyr)
long_cashflow <- gather(cashflow, key = Month, value = Value, -Metric)
```

Metric	1	2	3	4	5	6
Received	100.00	200.00	300.00	400.00	500.00	500.00
Spent	150.00	175.00	200.00	225.00	250.00	250.00



Metric	Month	Value
Received	1	100
Received	2	200
Received	3	300
Received	4	400
Received	5	500
Received	6	500
Spent	1	150
Spent	2	175
Spent	3	200
Spent	4	225
Spent	5	250
Spent	6	250

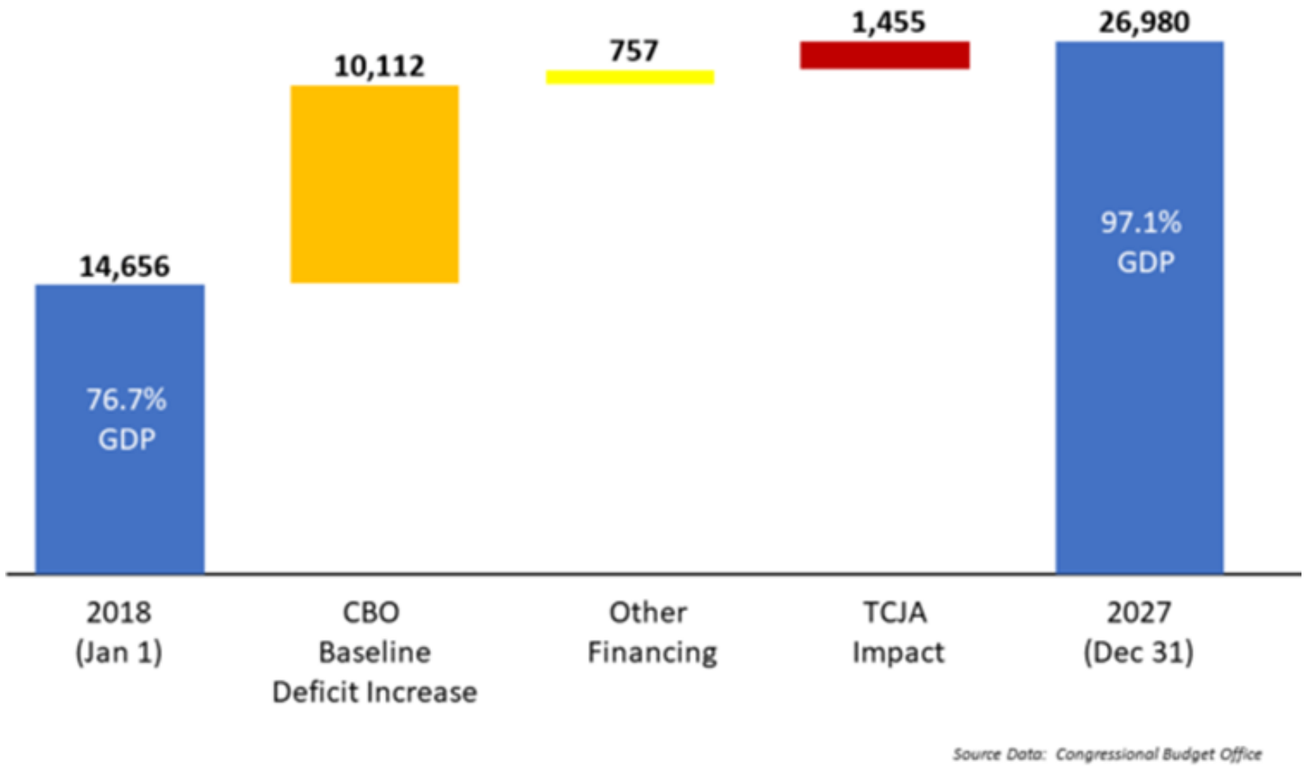
Tidying a cashflow (wide to long)

```
tidy_cashflow <- spread(long_cashflow, key = Metric, value = Value, -Metric)
```

Metric	Month	Value
Received	1	100
Received	2	200
Received	3	300
Received	4	400
Received	5	500
Received	6	500
Spent	1	150
Spent	2	175
Spent	3	200
Spent	4	225
Spent	5	250
Spent	6	250

Month	Received	Spent
1	100	150
2	200	175
3	300	200
4	400	225
5	500	250
6	500	250

Waterfall diagrams

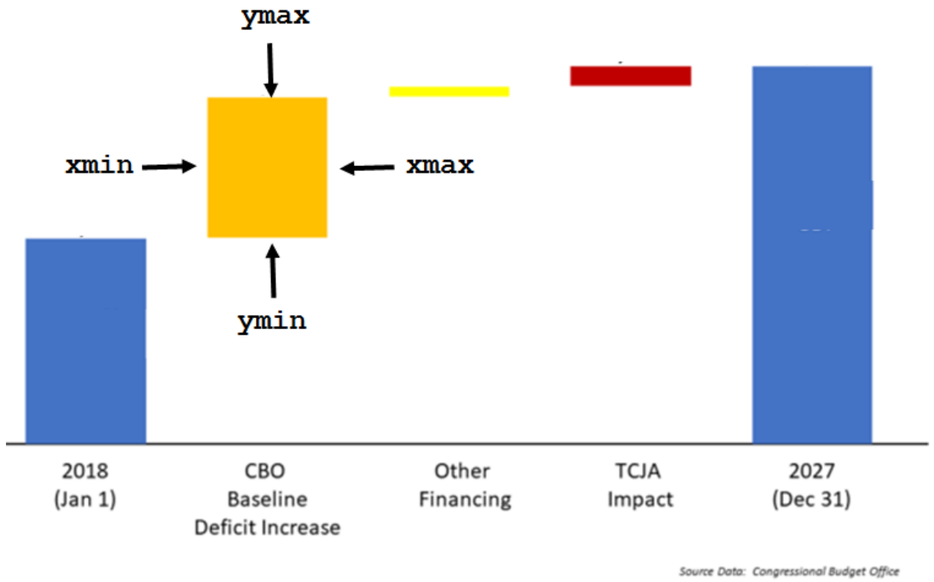


Source: US Congressional Budget Office. https://www.cbo.gov/publication/53348

Waterfall diagrams in ggplot2

`ggplot2`'s `geom_rect` lets us control bar height and orientation:

```
library(ggplot2)

ggplot(data) +
geom_rect(
    aes( xmin = , xmax = ,
         ymin = , ymax =
    )
  )
```



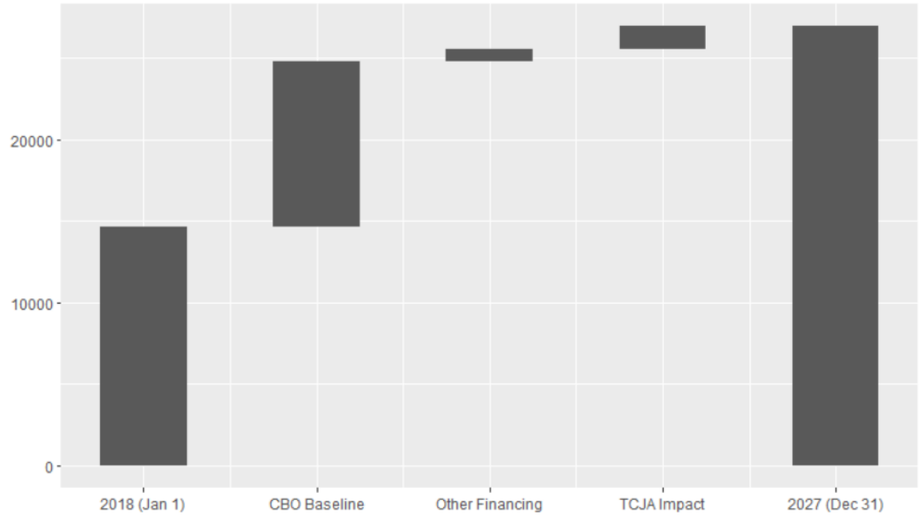
Waterfall diagrams in ggplot2

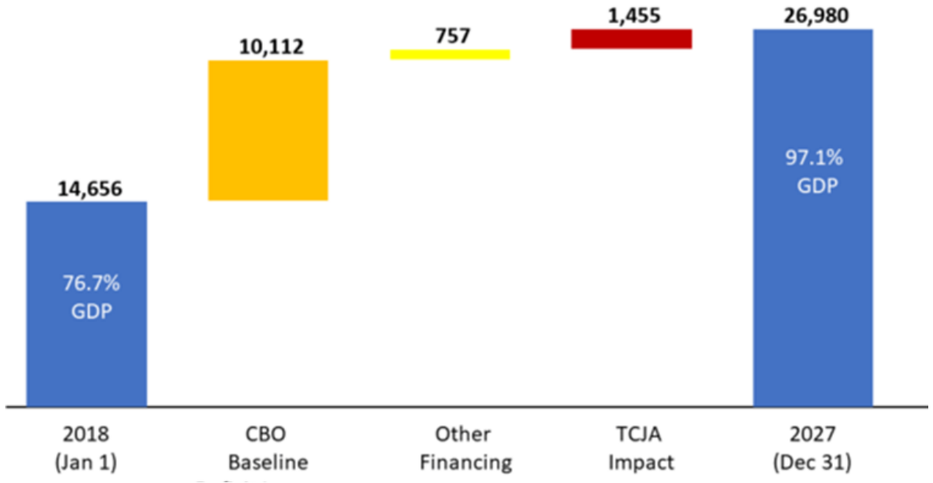
```
waterfall_data
```

rn	category	amount	start	end
1	2018	14656	0	14656
2	Baseline	10112	14656	24768
3	Other	757	24768	25525
4	TCJA	1455	25525	26980
5	2027	26980	0	26980

Waterfall diagrams in ggplot2

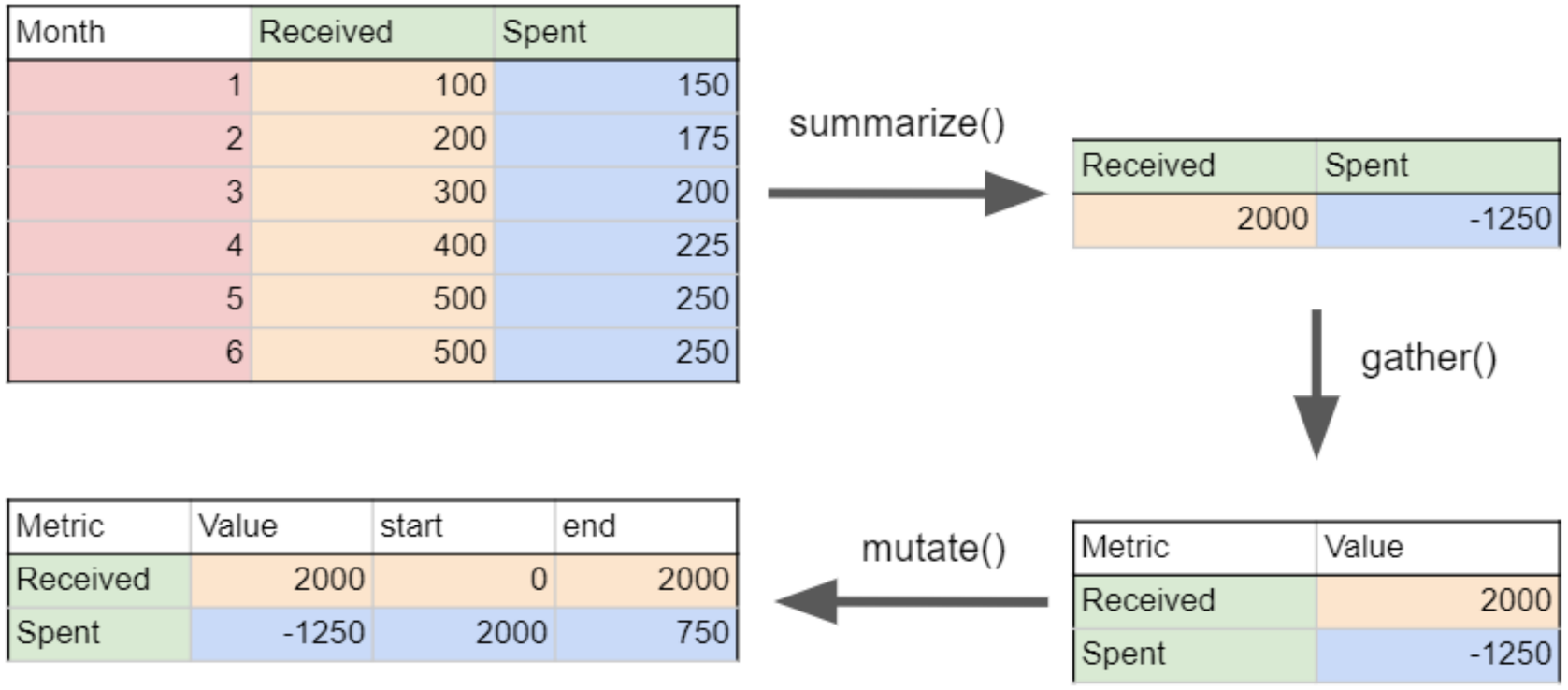
```
ggplot(waterfall_data,
  aes(
    xmin =  rn - 0.25,
    xmax = rn + 0.25,
    ymin = start, ymax = end)
  ) +
  geom_rect() +

  scale_x_continuous(
    breaks = waterfall_data$rn,
    labels = waterfall_data$category
    )
```





Wrangling data for waterfall diagrams

- Need to derive `ymin` and `ymax` from cashflow output



FINANCIAL ANALYTICS IN R

One last time...

FINANCIAL ANALYTICS IN R

Wrapping Up

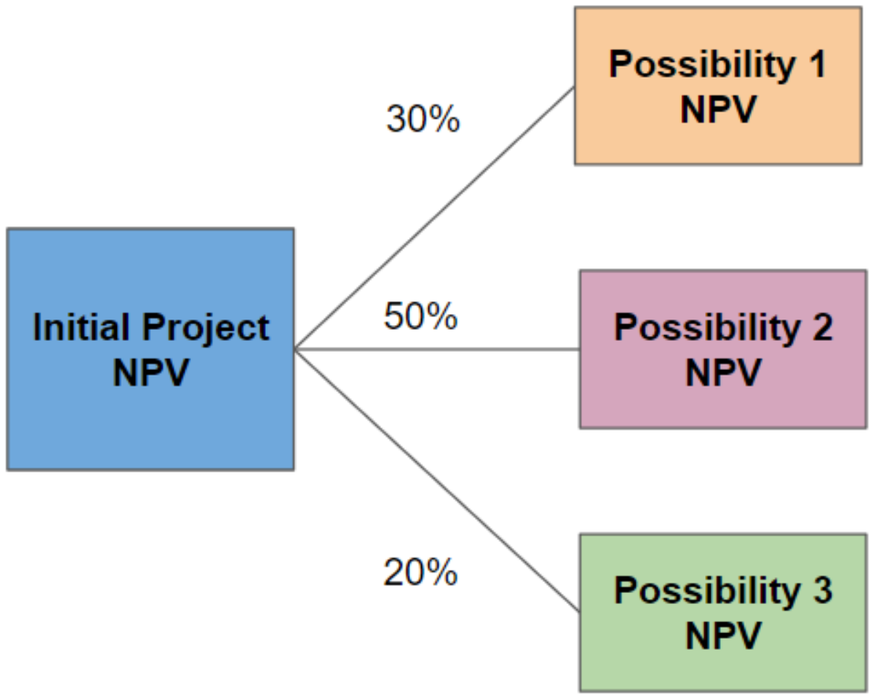
Emily Riederer
Instructor

Capital Structure

- How you fund your investment
- Many nuanced options all based on some combination of:
  - Debt: loan requiring repayment
  - Equity: firm "stock" or ownership

Valuing Future Options/Decisions

One project opens the door to others




```
total_npv <-
    init_npv +
        0.3 * npv1 +
        0.5 * npv2 +
        0.2 * npv3
```

Probabilistic Simulation

Where's the (admission of) uncertainty?

Deterministic:

```
assumptions$sales <- 5000
```

Stochastic:

```
assumptions$sales <- rnorm(n = 10, mean = 5000, sd = 200)
```

FINANCIAL ANALYTICS IN R

Congratulations!