Implementing Simple Regression Models in R

Vitthal Srinivasan
CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Build regression models in R

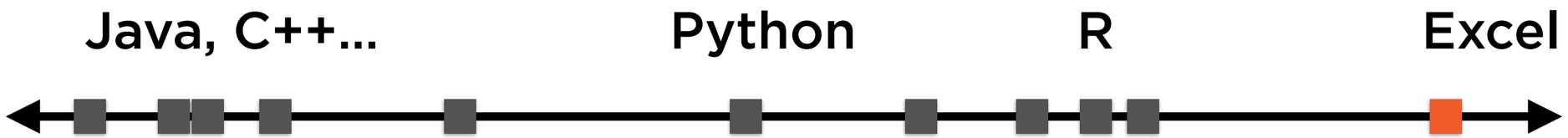
Understand and test the regression assumptions

Use simple regression models in R

- to explain variance
- to make forecasts

Avoid some common regression pitfalls

Ease of Prototyping



Excel is the fastest prototyping tool out there

Robustness and Reuse

No free lunches

“Make the common use-case easy and the difficult use-case possible.”

Regression: Excel, R or Python?

Excel

Create a regression slide for an important presentation

R

Create a regression case study for a seminar

Python

Build trading model that scrapes websites, combines sentiment analysis and regression

Regression: Excel, R or Python?

Excel

Presentations

R

Seminars

Python

Trading models

R for Regression

R

Presentations

R

Seminars

R

Trading models

Use **R for regression**: It makes sense whatever your use-case

Excel or R for Linear Regression?

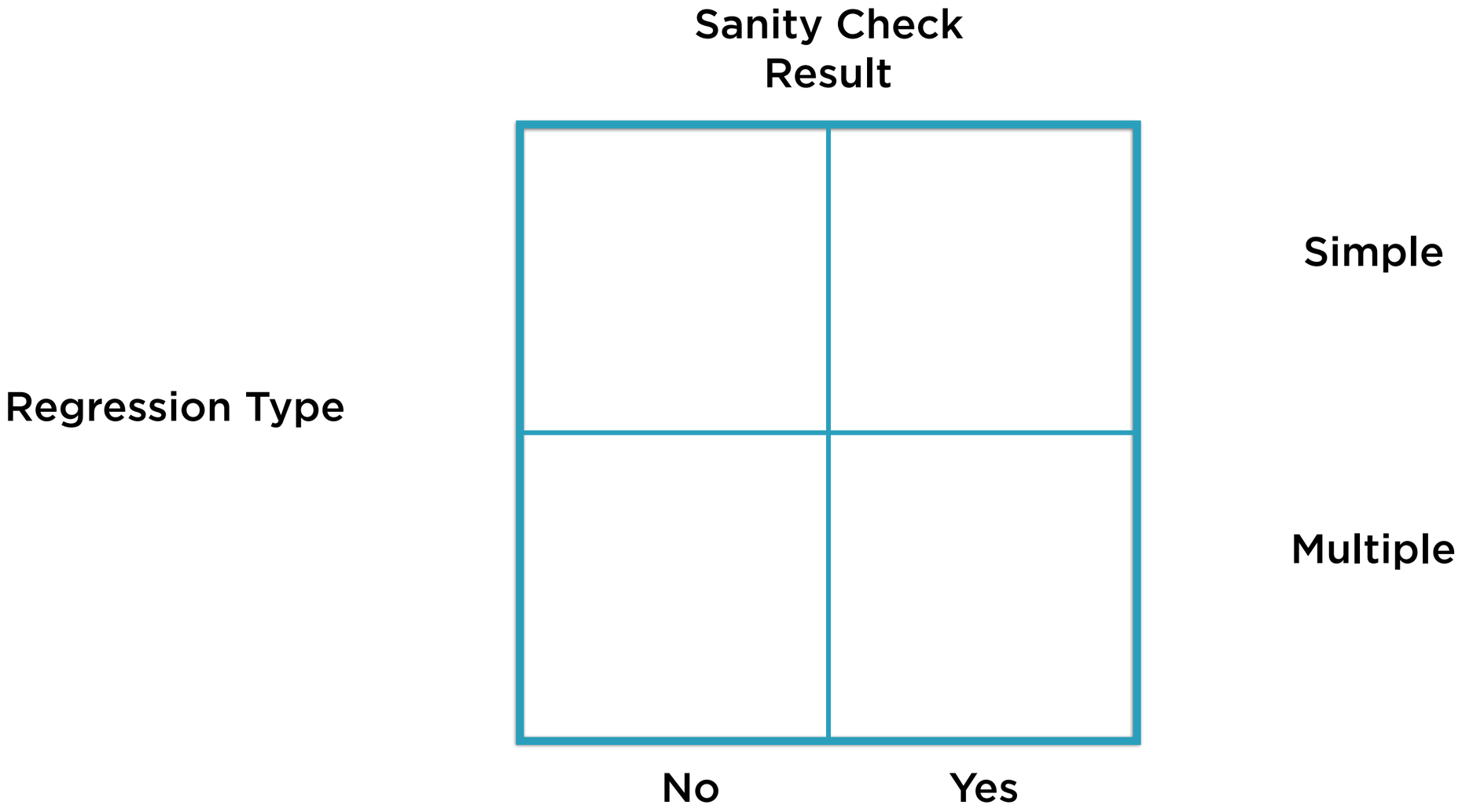
Number of independent variables

Simple or Multiple Regression?

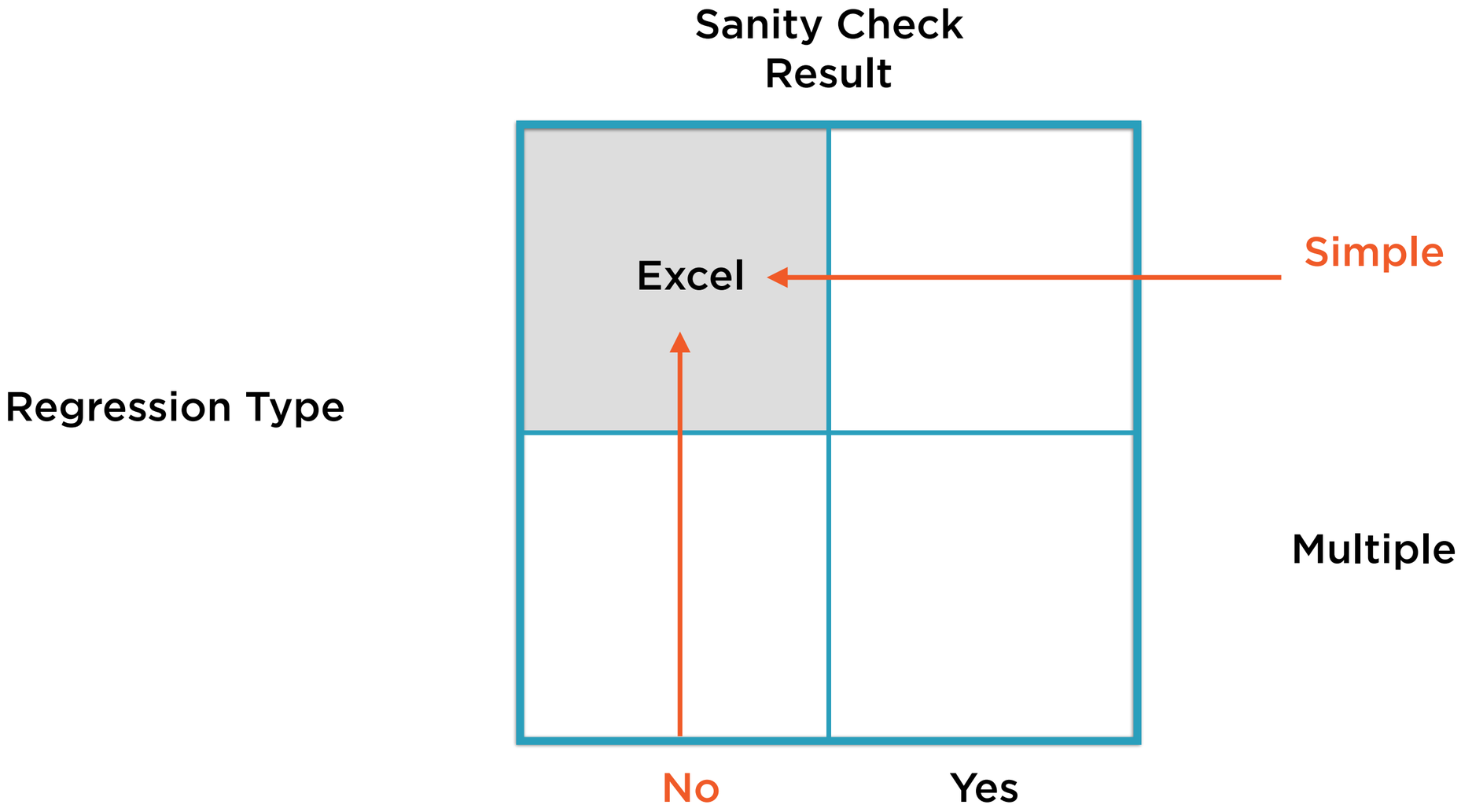
Sanity check regression model

Accept model as-is or analyse and tweak?

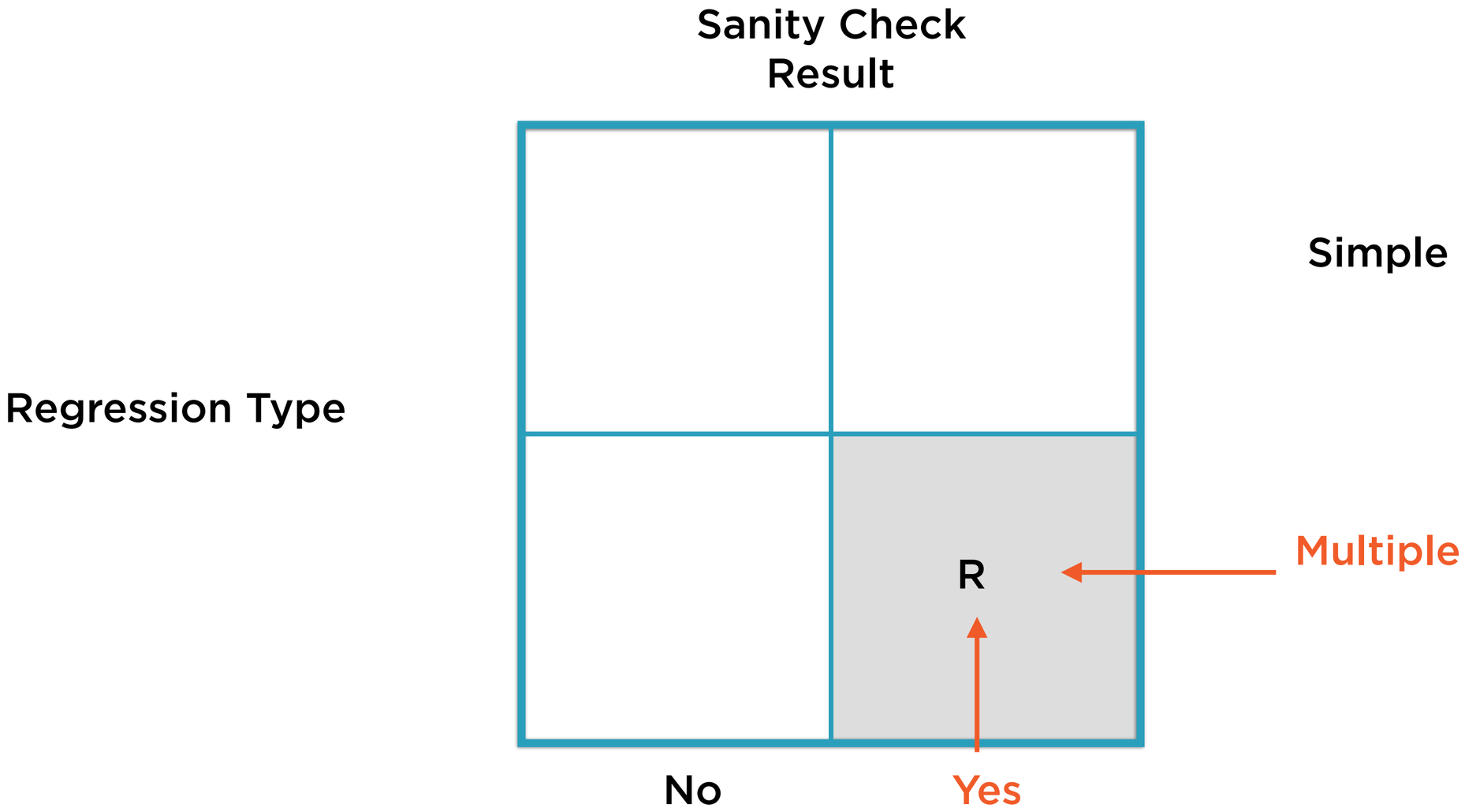
Excel or R for Linear Regression?



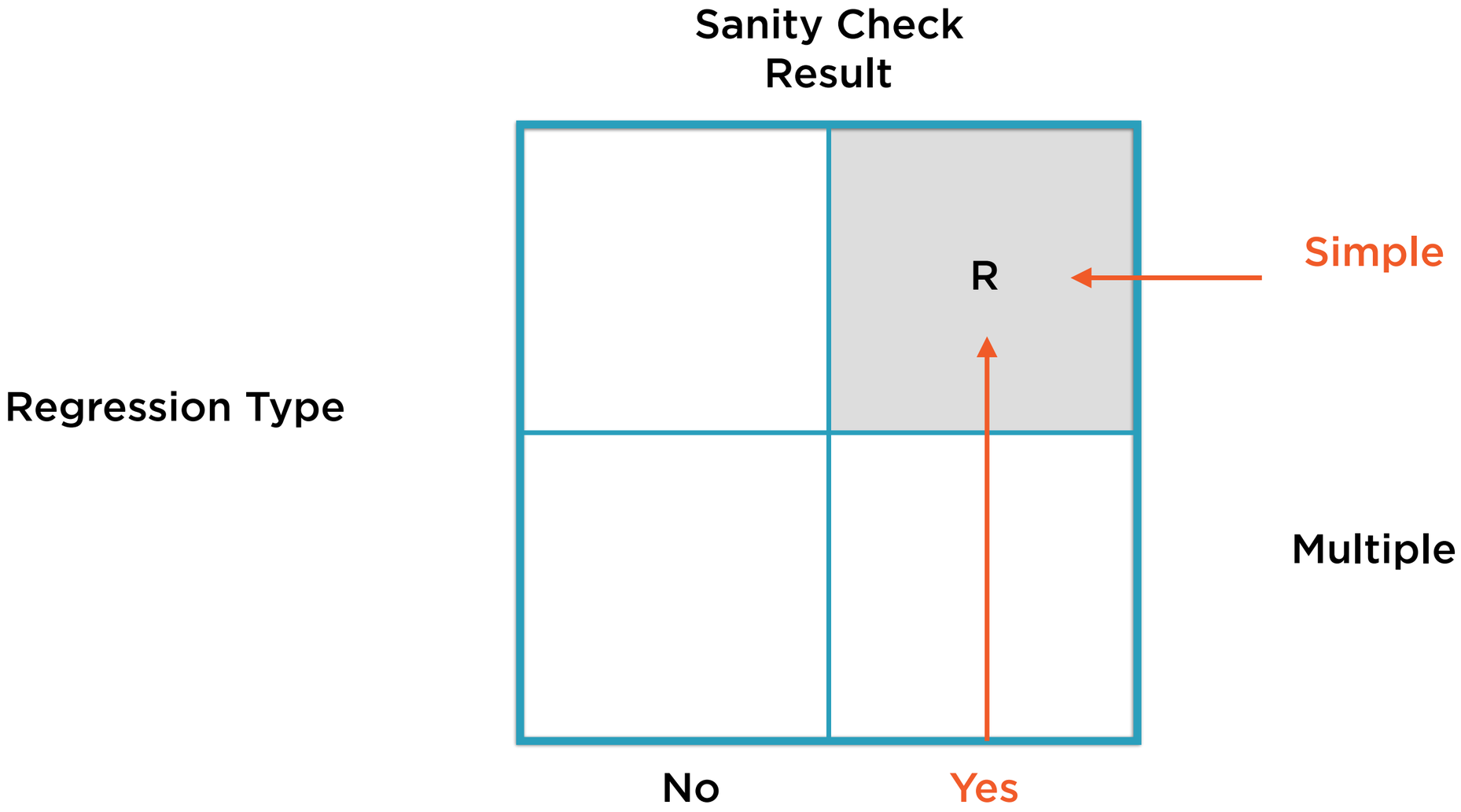
Excel or R for Linear Regression?



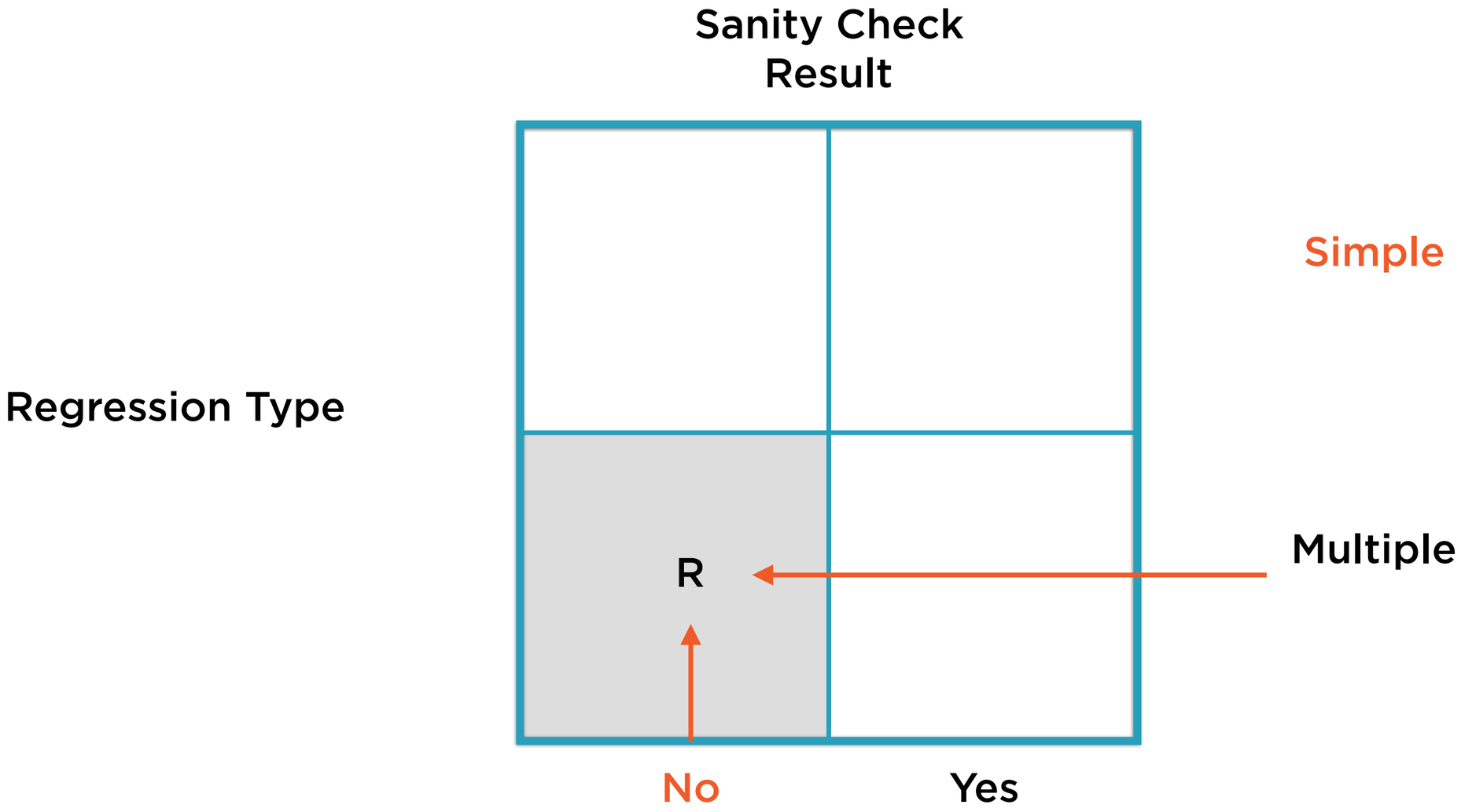
Excel or R for Linear Regression?



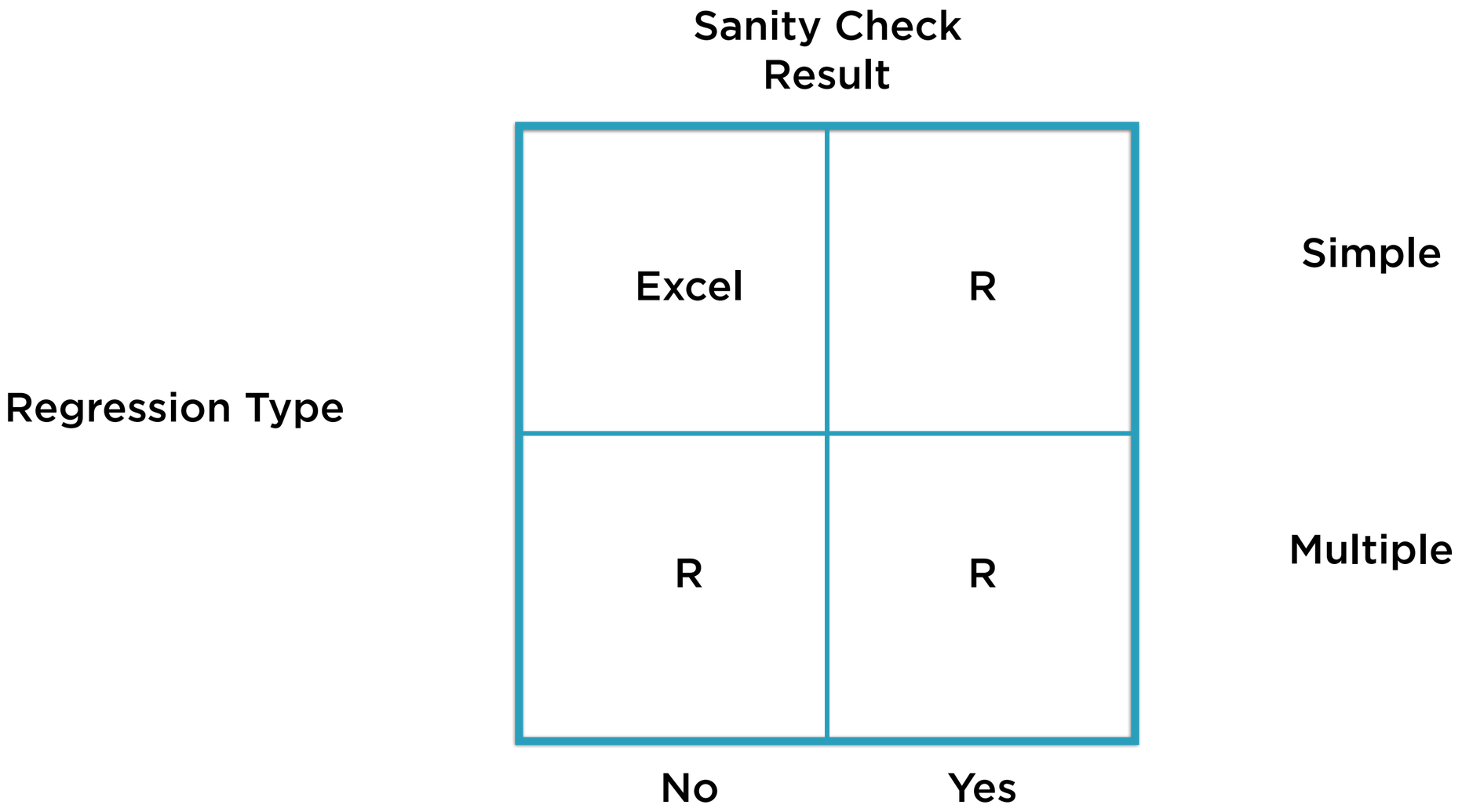
Excel or R for Linear Regression?



Excel or R for Linear Regression?



Excel or R for Linear Regression?



Demo

Simple regression models in R

Data Frame: Data in Rows and Columns

Each row represents 1 observation

DATE	OPEN	...	ADJUSTED CLOSE
2016-12-01	772	...	779
2016-11-01	758	...	747
2006-01-01	302	...	309

Each column represents 1 variable (a list or vector)

From File to Data Frame

DATE	OPEN	…	ADJUSTED CLOSE
2016-12-01	772	…	779
2016-11-01	758	…	747
2006-01-01	302	…	309

File

DATE	OPEN	…	ADJUSTED CLOSE
2016-12-01	772	…	779
2016-11-01	758	…	747
2006-01-01	302	…	309

Data Frame

Stripping Irrelevant Columns

DATE	OPEN	...	ADJUSTED CLOSE
2016-12-01	772	...	779
2016-11-01	758	...	747
2006-01-01	302	...	309

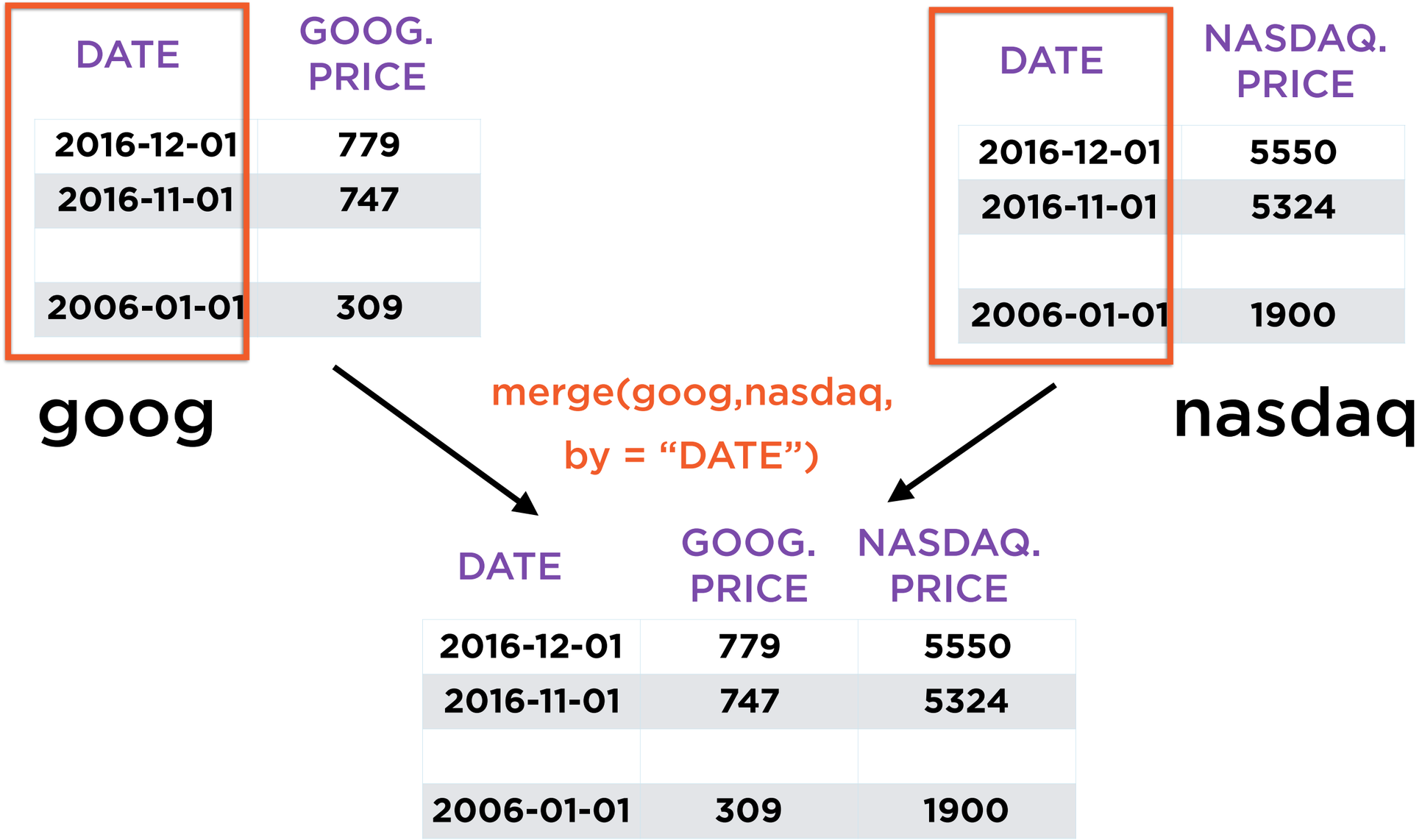
Data Frame 1

[,c("DATE","ADJUSTED CLOSE"]

DATE	ADJUSTED CLOSE
2016-12-01	779
2016-11-01	747
2006-01-01	309

Data Frame 2

Merging Data Frames



Negative Indices => Exclude Data

goog

DATE	GOOG. PRICE	NASDAQ. PRICE
2016-12-01	779	5550
2016-11-01	747	5324
2006-01-01	309	1900

Row 1

Row nrow(goog)

Column 1

```
goog[-nrow(goog),-1]
```

Negative Indices => Exclude Data

goog

DATE	GOOG. PRICE	NASDAQ. PRICE
2016-12-01	779	5550
2016-11-01	747	5324
2006-01-01	309	1900

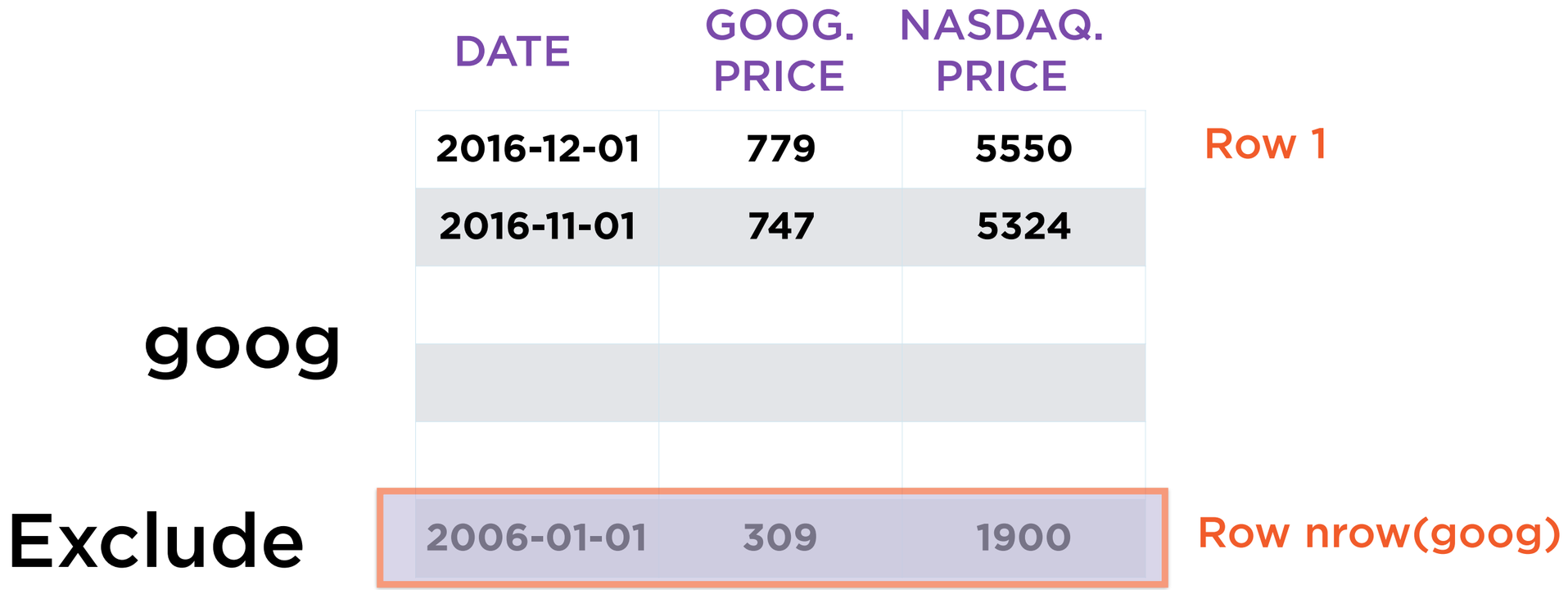
Row 1

Row nrow(goog)

Column 1

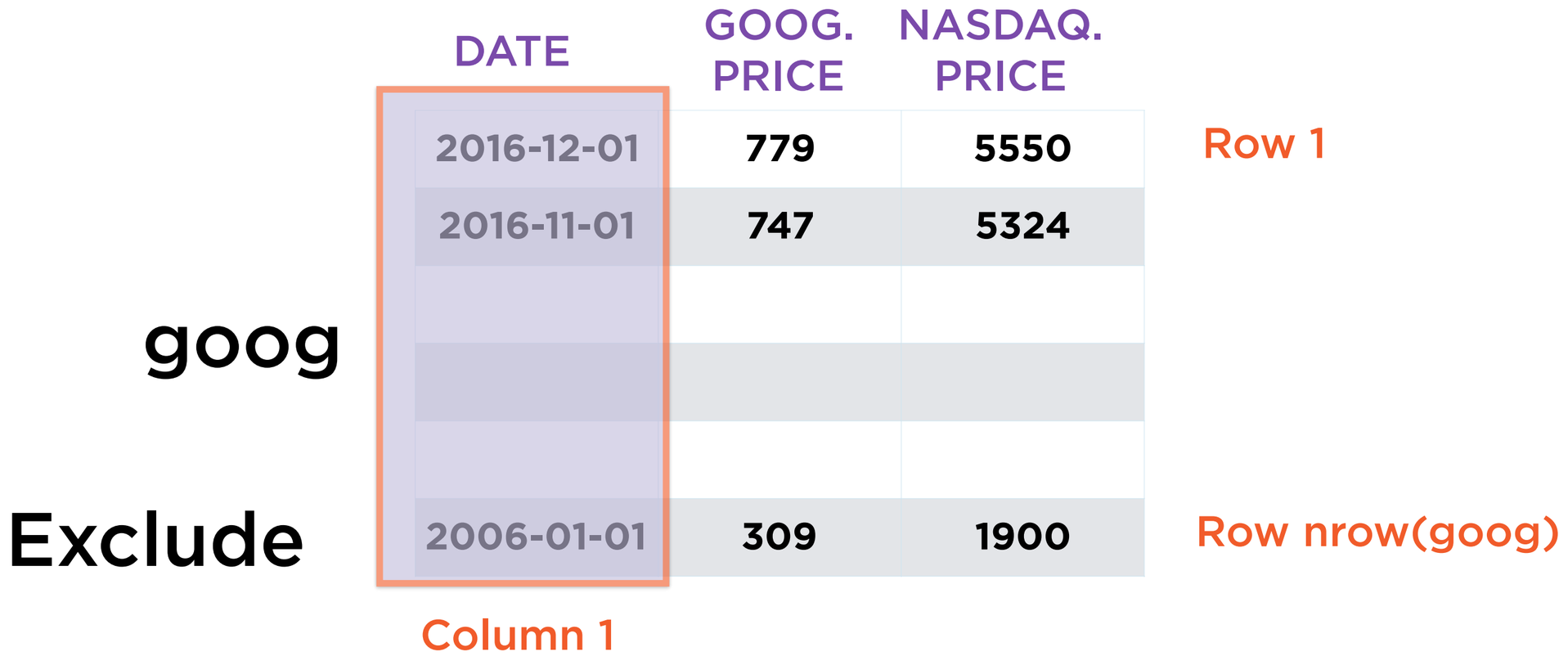
```
goog[-nrow(goog),-1]
```

Negative Indices => Exclude Data



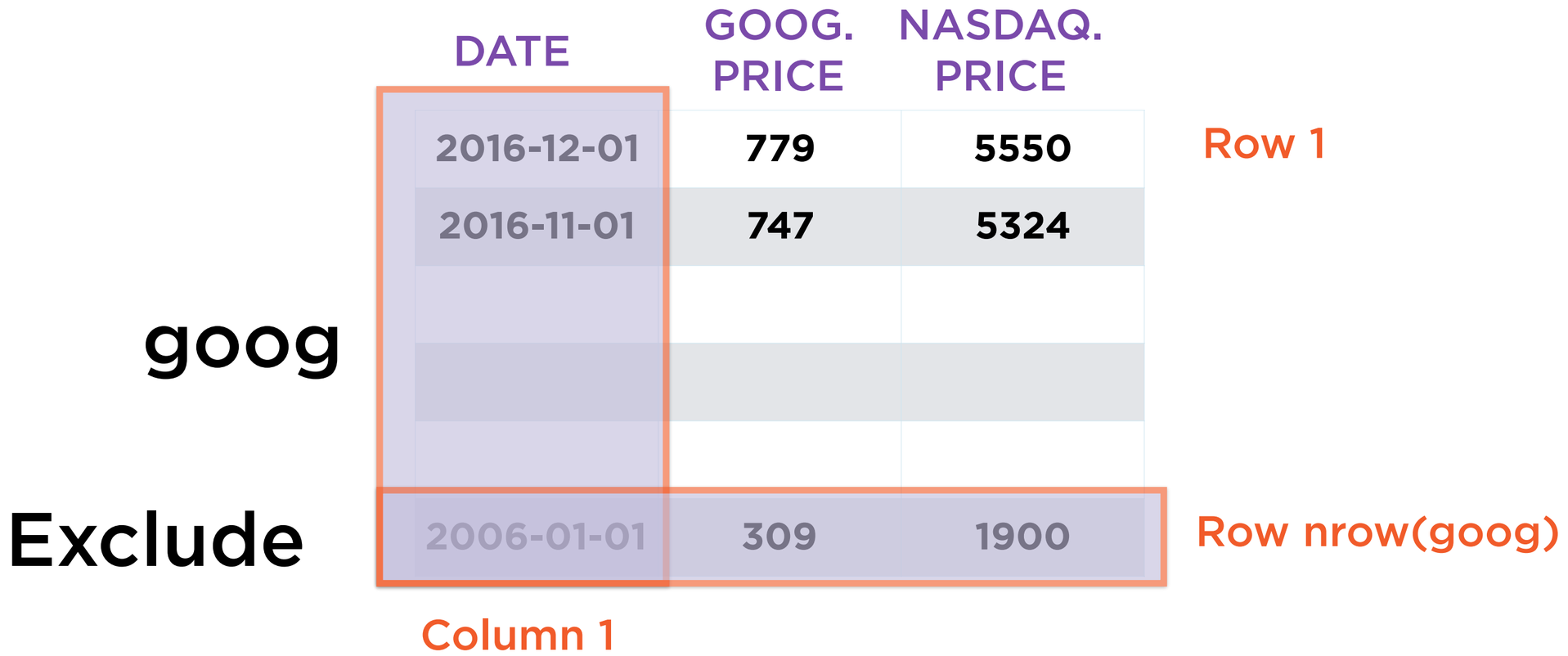

```
goog[-nrow(goog),-1]
```

Negative Indices => Exclude Data




```
goog[-nrow(goog),-1]
```

Negative Indices => Exclude Data




```
goog[-nrow(goog),-1]
```

Element-wise Operations

779	5550

/

747	5324

=

779/747	5550/5324
…	…
…	…

```
goog[-nrow(goog),-1]/
goog[-1,-1]
```

Element-wise Operations

779/747	5550/5324
…	…
…	…

-

1	1
1	1
1	1
1	1
1	1

=

779/747 - 1	5550/5324 -1
…	…
…	…

```
goog[-nrow(goog),-1]/
goog[-1,-1] - 1
```

This converts prices to returns

Summary

Built regression models in R

Avoided some common regression pitfalls

Use simple regression models in R

- to explain variance
- to make forecasts