Rotman

INTRO TO DATA VISUALIZATION

Part IV Build Dashboards with Quarto and Plotly Express

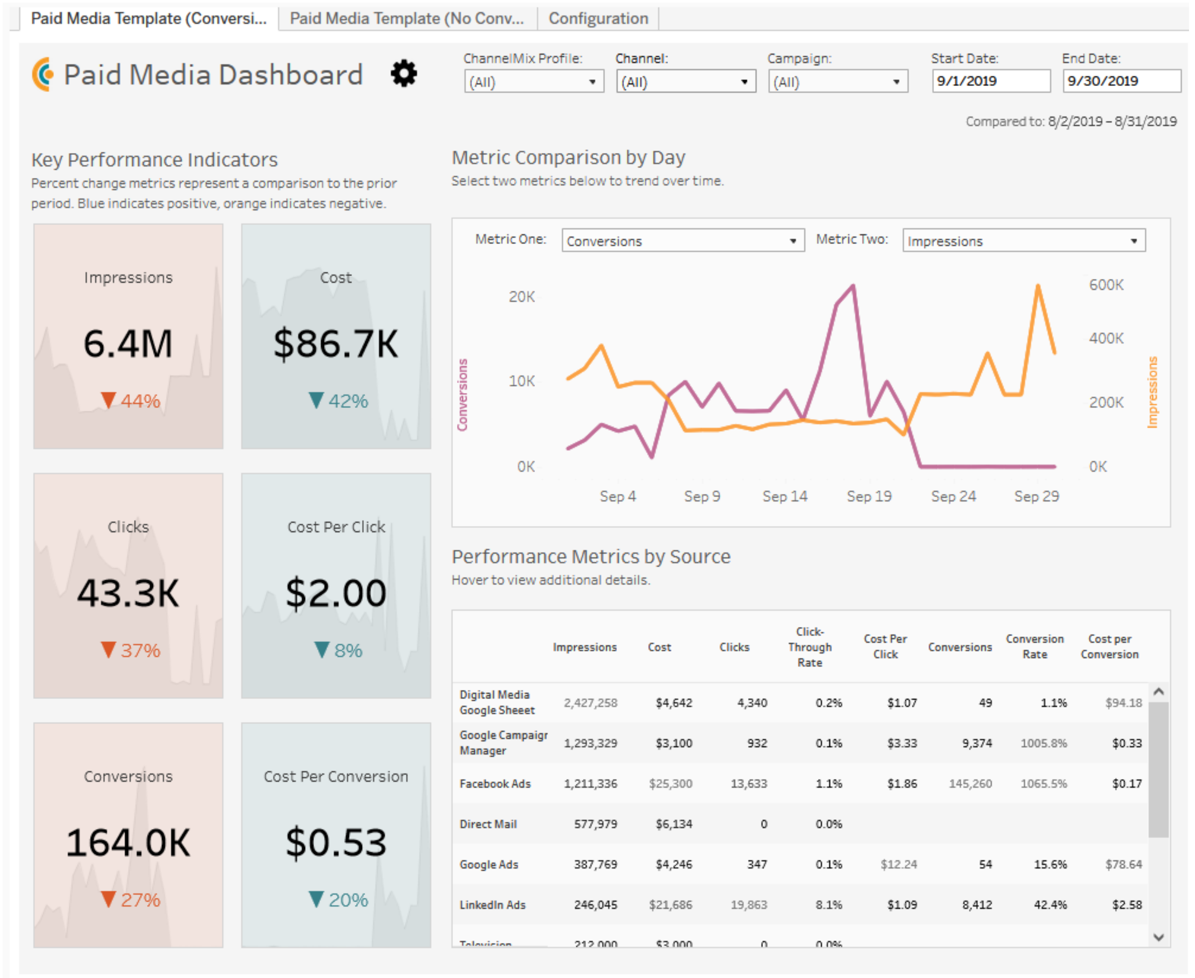
September 4, 2025 Prepared by Jay Cao / MDAL

Website: https://rmdal.github.io/mma-dv-2025/

Rotman School of Management
UNIVERSITY OF TORONTO

What is a Dashboard

- A way to display related data visualization and summary in one place
- Usually contains interactive and dynamic features
- Usually accessible via a web browser
- Market campaign performance example
  - Key Performance Indicators (KPIs)
  - Metric comparison line plot
  - Aggregated data table
  - Interactivity (data filters, etc.)



Key Performance Indicators

Percent change metrics represent a comparison to the prior period. Blue indicates positive, orange indicates negative.

Impressions 6.4M ▼44%	Cost $86.7K ▼42%
Clicks 43.3K ▼37%	Cost Per Click $2.00 ▼8%
Conversions 164.0K ▼27%	Cost Per Conversion $0.53 ▼20%

Metric Comparison by Day

Select two metrics below to trend over time.

Performance Metrics by Source

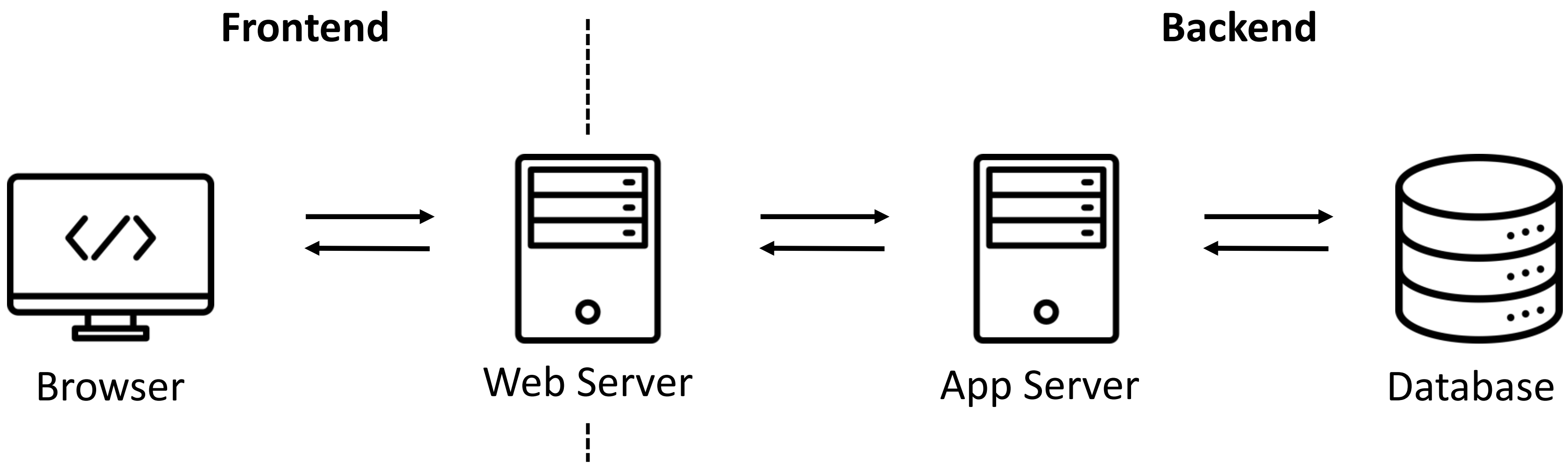
Hover to view additional details.

	Impressions	Cost	Clicks	Click-Through Rate	Cost Per Click	Conversions	Conversion Rate	Cost per Conversion
Digital Media Google Sheeet	2,427,258	$4,642	4,340	0.2%	$1.07	49	1.1%	$94.18
Google Campaigr Manager	1,293,329	$3,100	932	0.1%	$3.33	9,374	1005.8%	$0.33
Facebook Ads	1,211,336	$25,300	13,633	1.1%	$1.86	145,260	1065.5%	$0.17
Direct Mail	577,979	$6,134	0	0.0%				
Google Ads	387,769	$4,246	347	0.1%	$12.24	54	15.6%	$78.64
LinkedIn Ads	246,045	$21,686	19,863	8.1%	$1.09	8,412	42.4%	$2.58
Television	212,000	$3,000	0	0.0%				

Source: Paid Media Dashboard

Ref: https://www.tableau.com/learn/articles/dashboards/what-is

Most Dashboards are Web Apps



Render a web page
- HTML
- CSS
- JAVASCRIPT

Deliver static content
- HTML, CSS, JAVASCRIPT
- Images, Videos, etc.

Handle App Logic
Interact with DB
Generate dynamic content

e.g. a new HTML page based on user input, DB query and some computation logic.

Run-time language: Python, Javascript (Node.js), R (Shiny), etc.

Store and manage data

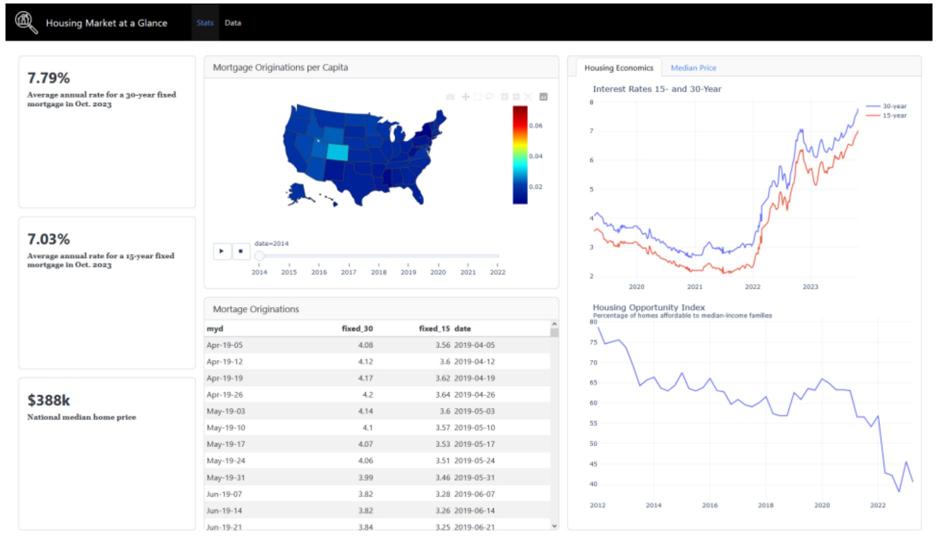
Python Dashboard Tooling

- Purpose of those tools
  - Make dashboard building easy
  - Require minimum knowledge on web technology (html, css, javascript, etc.)
- Landscape
  - Dash (by Plotly)
  - Panel (by Anaconda)
  - Voila (a subproject of Project Jupyter by Quantstack)
  - Streamlit
  - Gradio (specialize in ML apps)
  - Shiny for Python (by Posit)
  - Quarto Dashboards (by Posit)

Ref: 1) https://pyviz.org/dashboarding/index.html; 2) https://posit.co/blog/why-shiny-for-python/

Quarto Dashboards

- Easy to use (Markdown + Python or R)
  - IMO, truly need little knowledge of web dev
- Support Python and R
- Support simple interactivity (via Javascript)
  - can be deployed as static web pages
- Support enhanced interactivity (via Shiny for Python)
  - need to be deployed with Shiny Server
- Downside
  - Fairly new and in active dev; hence feature-incomplete and may have bugs



Housing Market at a Glance
Stats
Data
Mortgage Originations per Capita
date=2014
Housing Economics
Median Price
Interest Rates 15- and 30-Year
30-year
15-year
Housing Opportunity Index
Percentage of homes affordable to median-income families

7.79%
Average annual rate for a 30-year fixed mortgage in Oct. 2023

7.03%
Average annual rate for a 15-year fixed mortgage in Oct. 2023

$388k
National median home price

Mortage Originations

myd	fixed_30	fixed_15	date
Apr-19-05	4.08	3.56	2019-04-05
Apr-19-12	4.12	3.6	2019-04-12
Apr-19-19	4.17	3.62	2019-04-19
Apr-19-26	4.2	3.64	2019-04-26
May-19-03	4.14	3.6	2019-05-03
May-19-10	4.1	3.57	2019-05-10
May-19-17	4.07	3.53	2019-05-17
May-19-24	4.06	3.51	2019-05-24
May-19-31	3.99	3.46	2019-05-31
Jun-19-07	3.82	3.28	2019-06-07
Jun-19-14	3.82	3.26	2019-06-14
Jun-19-21	3.84	3.25	2019-06-21

An example of Quarto dashboard with simple interactivity
https://ivelasq.github.io/mortgage-dashboard/

Ref: https://quarto.org/docs/dashboards/

Simple Interactivity Dashboard

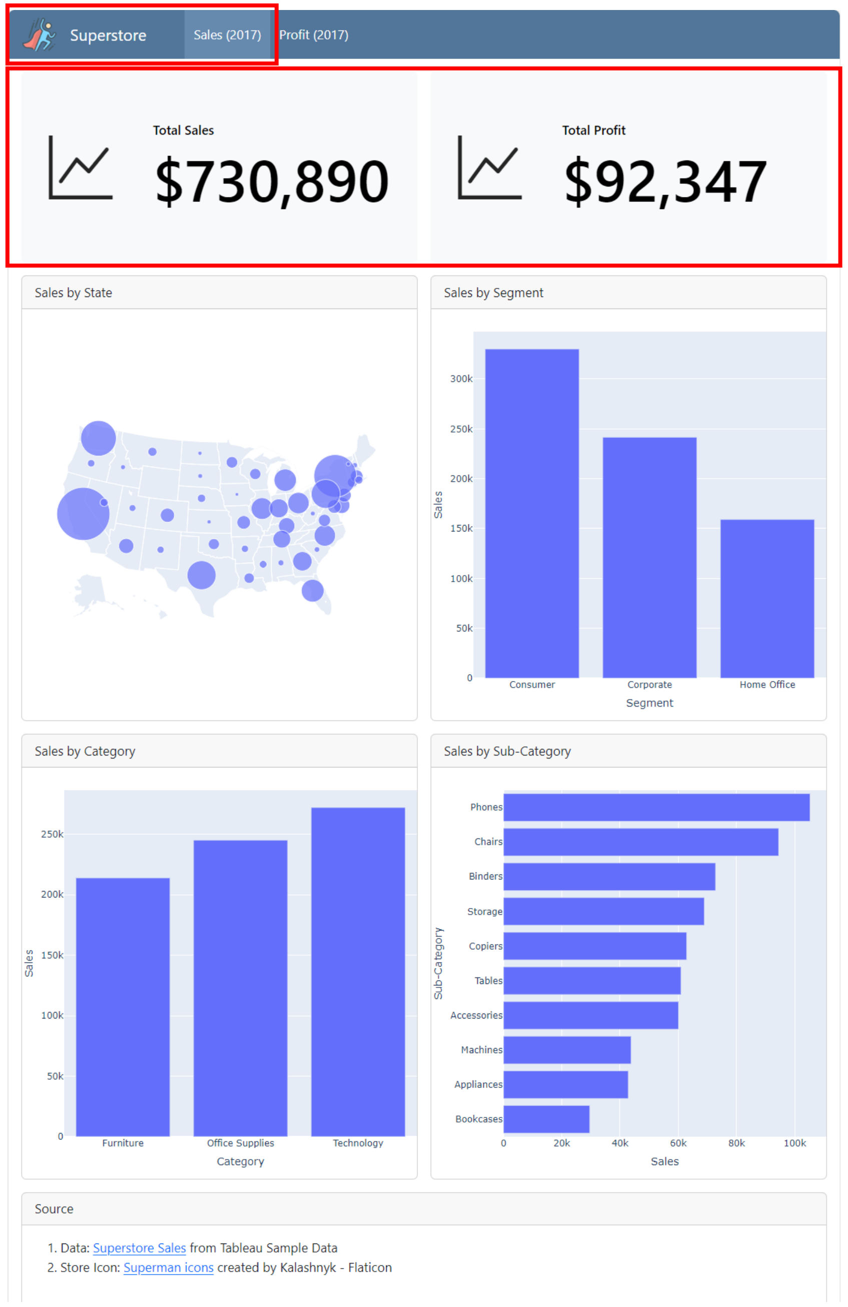
- Dashboard that need no backend to run (i.e. no app server, DB, etc.)
  - Sometimes called “static” dashboard
- Need only a web server to make it available to the internet
  - For example, you can host it on Github for free
- Use Javascript for basic interactivity (e.g., simple data filter)
  - You don’t need to how to code in Javascript
- Quarto Dashboard examples
  - https://jjallaire.github.io/stock-explorer-dashboard/ (note the web host)

Enhanced Interactivity Dashboard

- Dynamically retrieve data and display/visualize information
  - E.g., real-time update, complex underlying data transformation & analytics, etc.
- Support complex interactivity
  - E.g., user-based interactivity, user inputs that trigger backend business logics, etc.
- Require App server, DB, etc. in addition to a Web server
- Quarto Dashboard Examples
  - https://jjallaire.shinyapps.io/penguins-dashboard/ (note the web host)

Simple Db – Code Skeleton 1

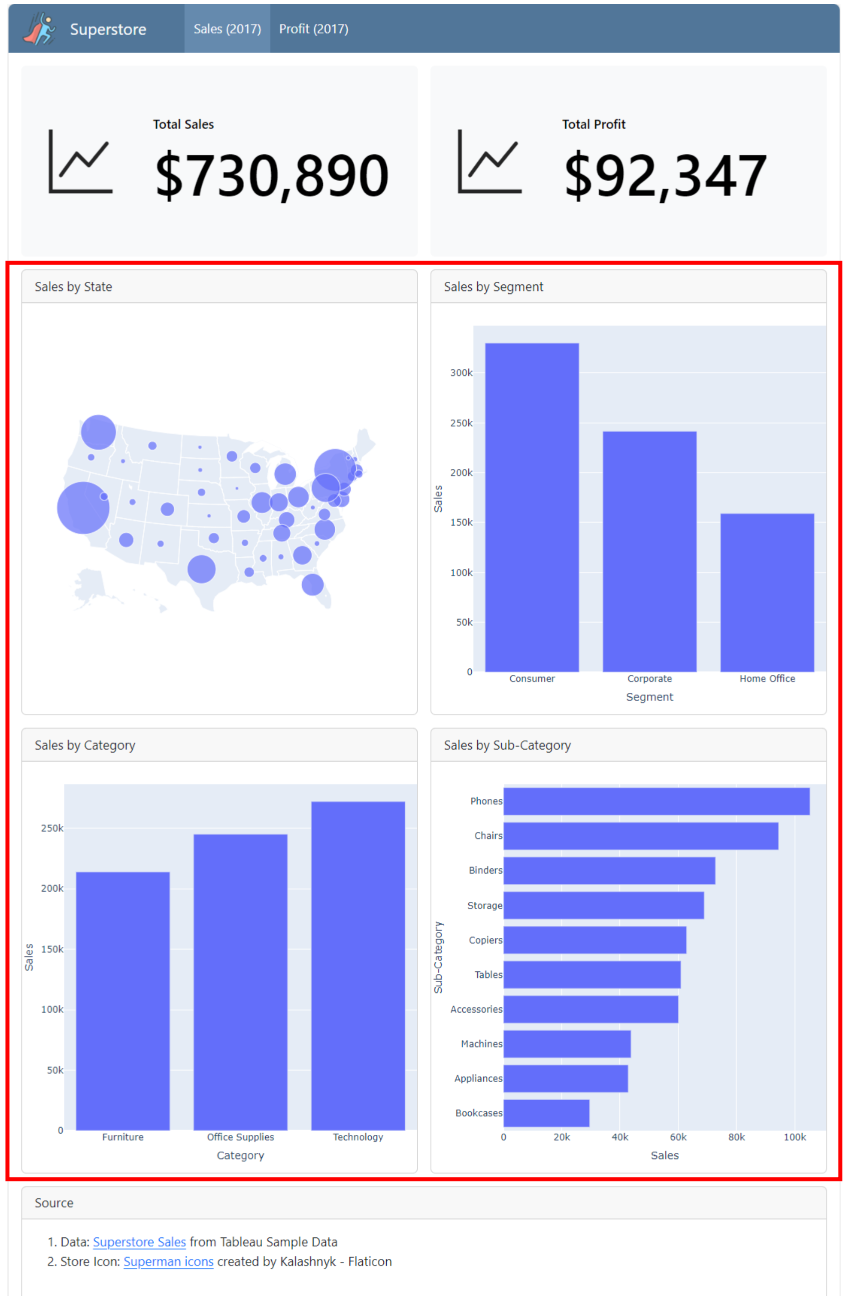
~~~
---
title: "Superstore"
format:
  dashboard:
    logo: super.png
---

```{python}
# load dataset, prepare it for display and plot

```

# Sales (`{python} year`)

## Row {height=15%}

```{python}
#| content: valuebox
#| title: "Total Sales"

```

```{python}
#| content: valuebox
#| title: "Total Profit"

```
~~~



1. Data: Superstore Sales from Tableau Sample Data
2. Store Icon: Superman icons created by Kalashnyk - Flaticon

Simple Db – Code Skeleton 2

~~~
## Row {height=35%}

```{python}
#| title: Sales by State
#| padding: 0

# sales by state plot
```

```{python}
#| title: Sales by Segment
#| padding: 0

# sales by segment plot
```

## Row {height=35%}

```{python}
#| title: Sales by Category
#| padding: 0

# sales by category plot
```

```{python}
#| title: Sales by Sub-Category
#| padding: 0

# sales by sub-category plot
```
~~~



Total Sales
$730,890

Total Profit
$92,347

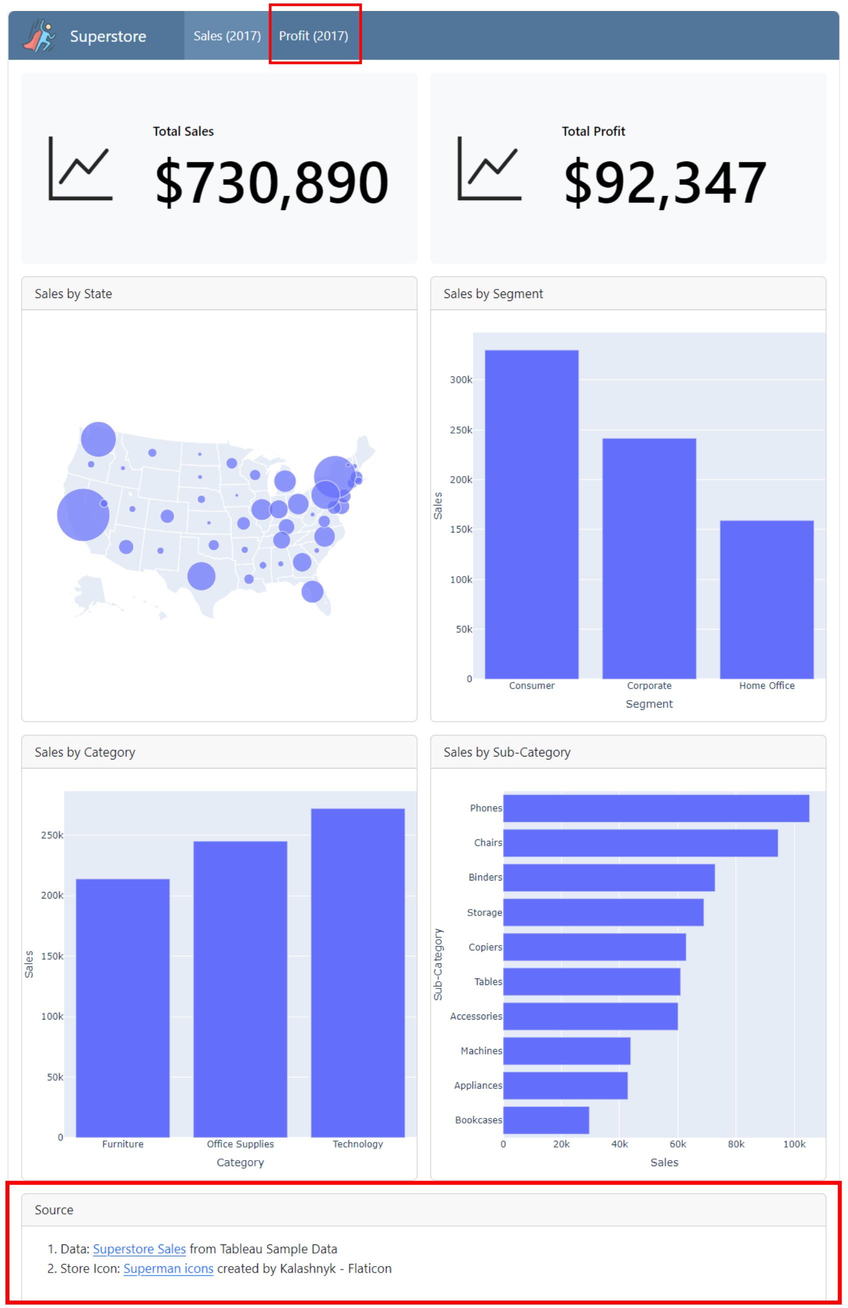
Source

1. Data: Superstore Sales from Tableau Sample Data
2. Store Icon: Superman icons created by Kalashnyk - Flaticon

Simple Db – Code Skeleton 3

```
## Row {height=15%}

::: {.card title="Source"}

:::

# Profit (`{python} year`)

**It's Your Turn to Build.**
```

```
> quarto render superstore.qmd
```



Superstore Sales (2017) Profit (2017)

Total Sales $730,890

Total Profit $92,347

Sales by State

Sales by Segment

Sales by Category

Sales by Sub-Category

Source

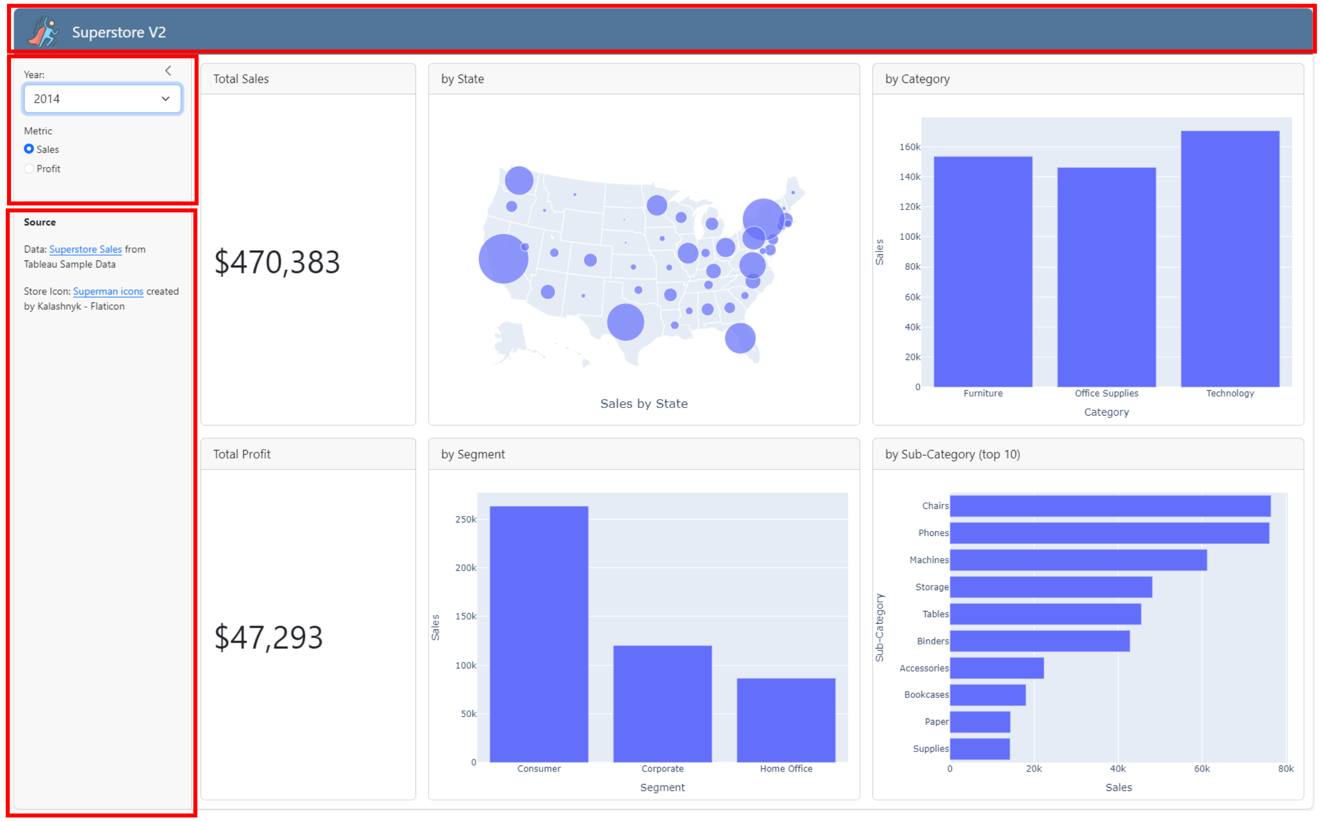
1. Data: Superstore Sales from Tableau Sample Data
2. Store Icon: Superman icons created by Kalashnyk - Flaticon

Enhanced Dashboard – Code Skeleton 1

~~~
---
title: "Superstore V2"
format:
  dashboard:
    logo: super.png
server: shiny
---

```{python}
# load python library (shiny, plotly.express, etc.)

# load and prepare data

~~~

## {.sidebar}
```{python}
ui.input_select()

ui.input_radio_buttons()
```

Source

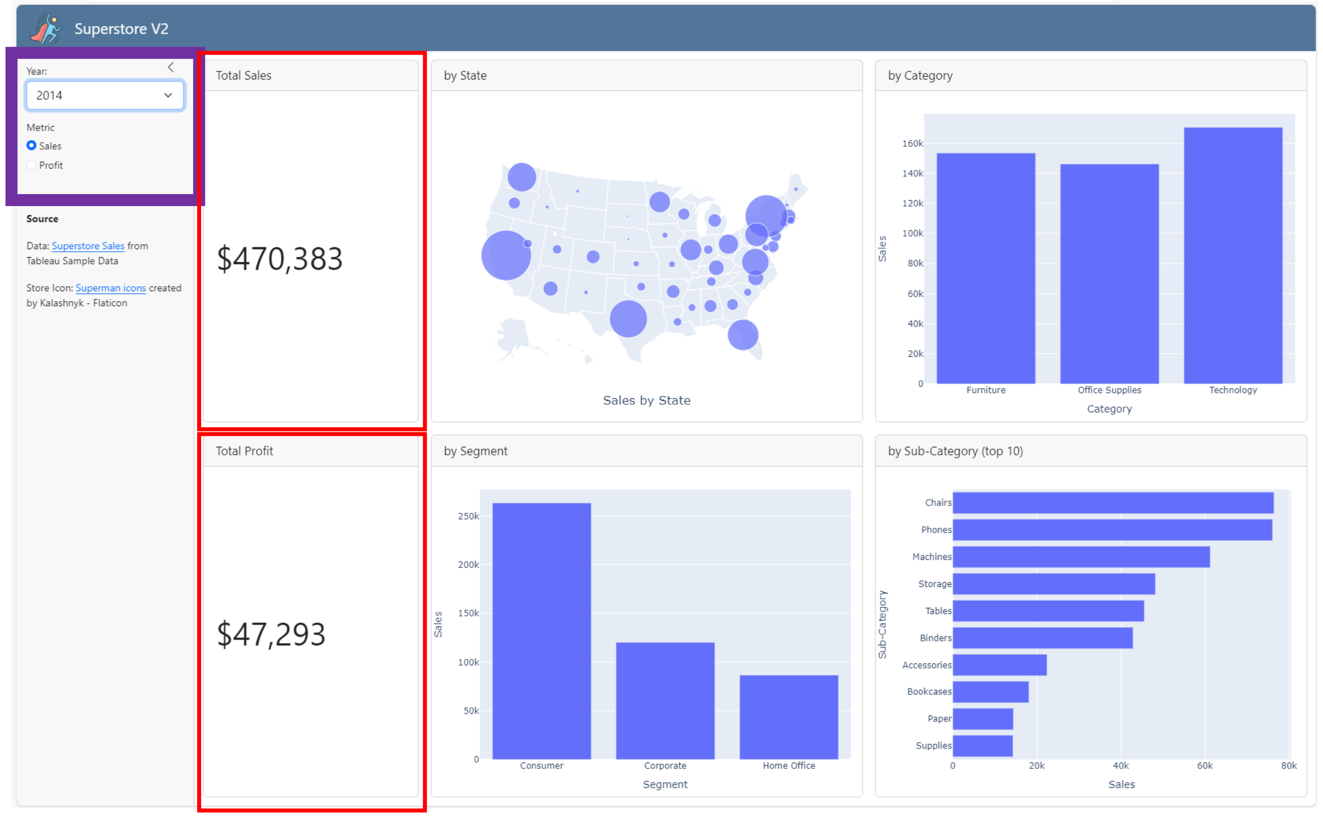
Data: [Superstore
Sales](https://public.tableau.com/app/learn/sample-
data) from Tableau Sample Data
```



Superstore V2

Year: 2014

Metric
[✓ Sales]
Profit

Source

Data: Superstore Sales from Tableau Sample Data

Store Icon: Superman icons created by Kalashnyk - Flaticon

Total Sales
$470,383

Total Profit
$47,293
```

# Enhanced Dashboard – Code Skeleton 2

~~~~
```{python}
@reactive.calc
def ss_year():
   return ss[ss["Ship
Date"].dt.year==int(input.year())]

@reactive.calc
def sales():
   return ss_year()["Sales"].sum()

# more reactive calculation
```


Column {width=20%}
```{python}
#| title: Total Sales
#| padding: 0
@render.ui
def sales_value_box():
   return shiny.ui.value_box()
```

```{python}
#| title: Total Profit
#| padding: 0
@render.ui
def profit_value_box():
   return shiny.ui.value_box()
```
~~~~


Superstore V2

Year: 2014

Metric
[✓ Sales]
Profit

Source

Data: Superstore Sales from Tableau Sample Data

Store Icon: Superman icons created by Kalashnyk - Flaticon

Total Sales: $470,383

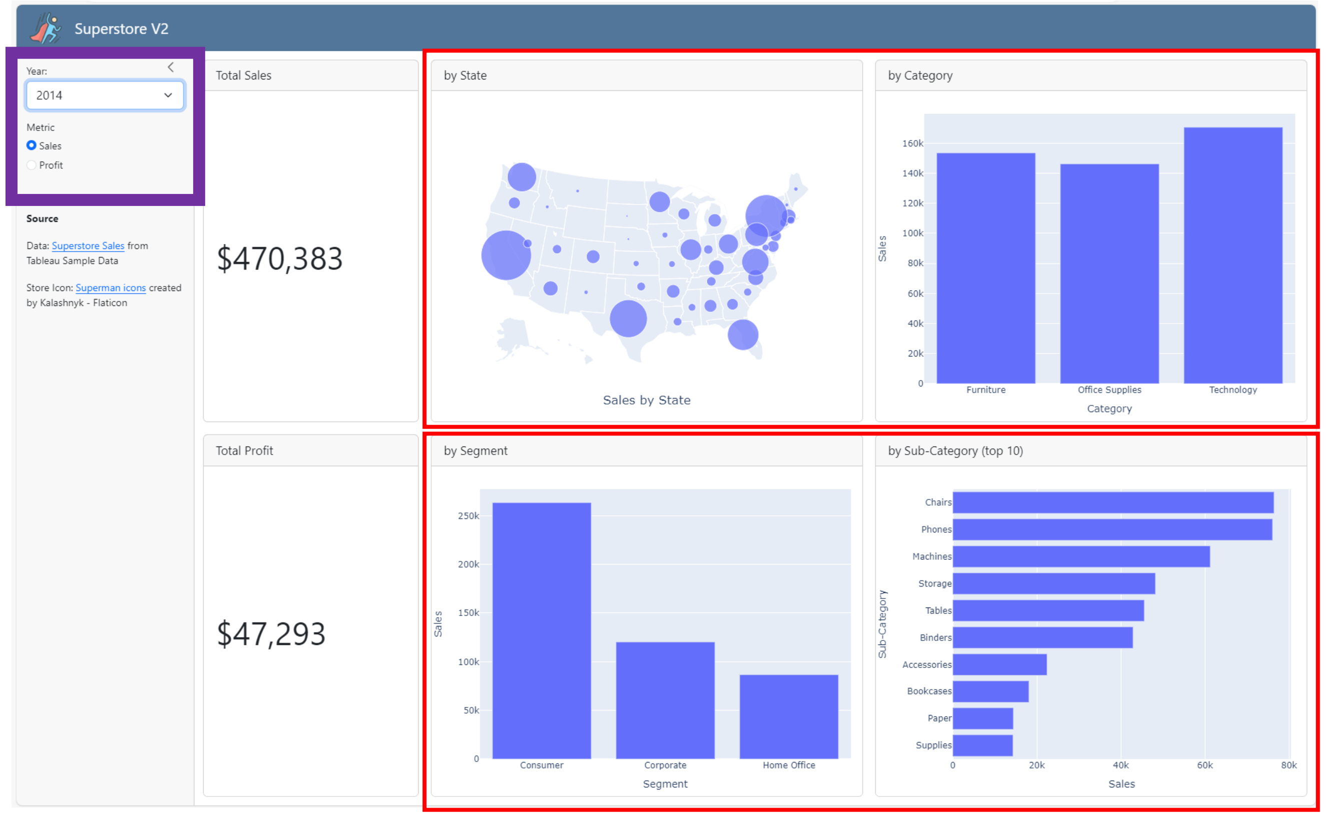
Total Profit: $47,293

Enhanced Dashboard – Code Skeleton 3

~~~
## Column {width=40%}
```{python}
#| title: by State
#| padding: 0

@render_widget
def plot_by_state():
    fig = px.scatter_geo()
    fig.layout.update()
    return fig
~~~

```{python}
#| title: by Segment
#| padding: 0

@render_widget
def plot_by_segment()
```

## Column {width=40%}
```{python}
#| title: by Category
#| padding: 0
```

```{python}
#| title: by Sub-Category (top 10)
#| padding: 0
```
```


Superstore V2
Year:
2014
Metric
Sales
Profit
Source
Data: Superstore Sales from Tableau Sample Data
Store Icon: Superman icons created by Kalashnyk - Flaticon
Total Sales
$470,383
Total Profit
$47,293
by State
Sales by State
by Category
Furniture
Office Supplies
Technology
Category
Sales
by Segment
Consumer
Corporate
Home Office
Segment
by Sub-Category (top 10)
Chairs
Phones
Machines
Storage
Tables
Binders
Accessories
Bookcases
Paper
Supplies
Sub-Category

```
> quarto render superstore-v2.qmd
> shiny run app.py
```
```

Deploy Your Dashboard

- Simple (static) dashboard
  - Any cloud services that can host websites
    - e.g., Quarto Pub, Github, Posit Connect Cloud, etc.
    - Document: Deploy to Quarto Pub, Github, Posit Connect Cloud (and more)
  - Of course you can setup your own web server too
  - You can use `quarto publish` command to make the deployment easy

- Enhanced (shiny) dashboard
  - Publishing services that supports shiny app
    - e.g., shinyapps.io Posit Connect Cloud and Shiny on Space from Hugging Face
    - Document: Deploy to shinyapp.io, Posit Connect Cloud (as of Aug 22, 2025, need to deploy as a shiny for Python app rather than a Quarto dynamic dashboard), Shiny on Space
  - You can also setup your own shiny server

Posit Connect Cloud

- Posit's next-generation online publishing platform
  - Support Quarto document, Shiny for Python app, Streamlit app, etc.

- Easy to use
  - Deploy a document or an app by pointing to its Github repo

- As of August 25, 2025
  - Deploying a simple (static) Quarto dashboard works well
  - For enhanced dynamic dashboard, need to deploy as a Shiny for Python app (instead of directly as a Quarto dynamic dashboard)