Dashboards in R

Enrique J. De La Hoz D.

Data Science - UTB

What is a Dashboard
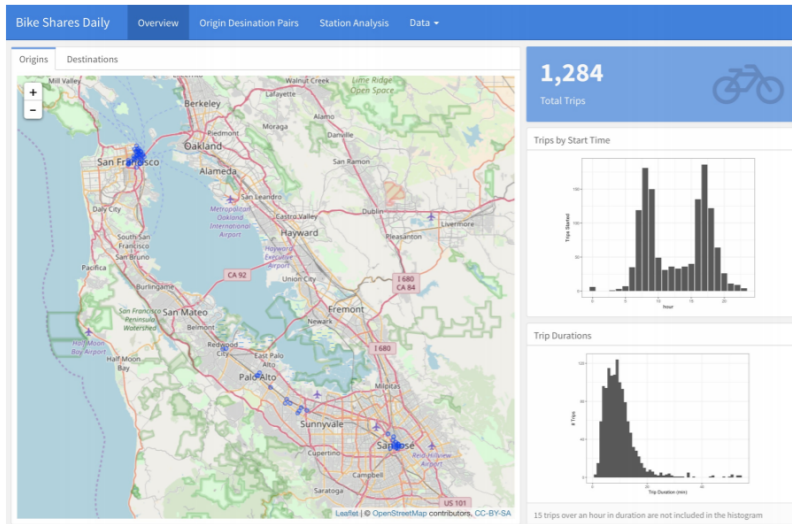

Figure 1: Example 1

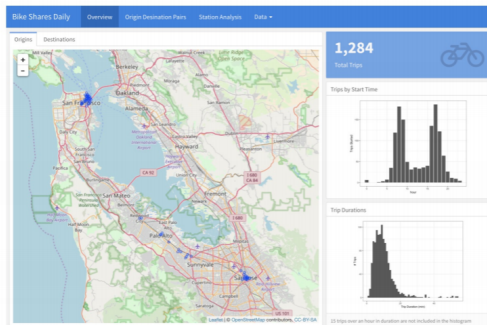# What is a FlexDashboard

What is flexdashboard?

Figure 2: Example 1

Rmarkdown the header

```
---
title: "Bikeshare"
output: html_document
---
```

Turning R Markdown into a flexdashboard

```
---
title: "Bikeshare"
output:
  flexdashboard::flex_dashboard:
---
```
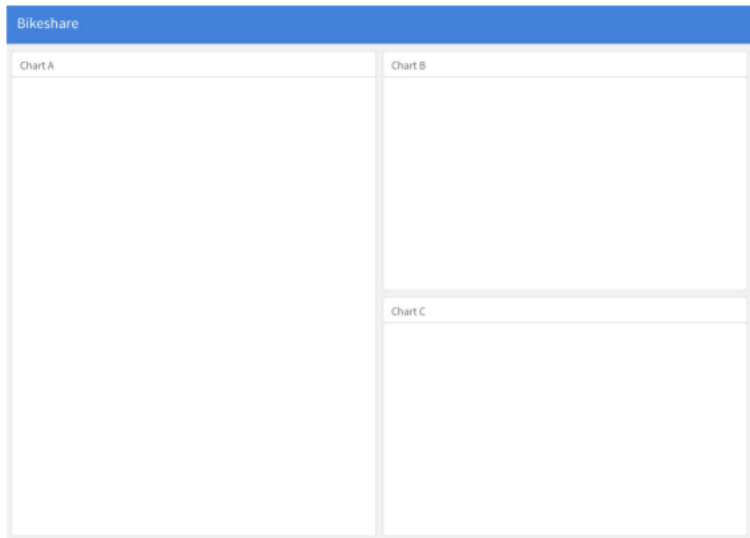
A flexdashboard is made of charts . . .

Bikeshare

Chart A

Chart B

Chart C

Figure 3:

A flexdashboard is made of charts . . .

```
---
title: "Bikeshare"
output:
  flexdashboard::flex_dashboard
---

```{r setup, include=FALSE}
library(flexdashboard)
```

Column

Chart A

```{r}
```

Chart B

```{r}
```

Chart C

```{r}
```
```

. . . and charts are arranged in columns

Figure 5:

Knit from template

Data

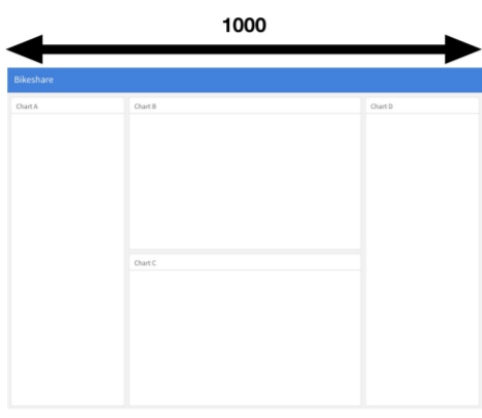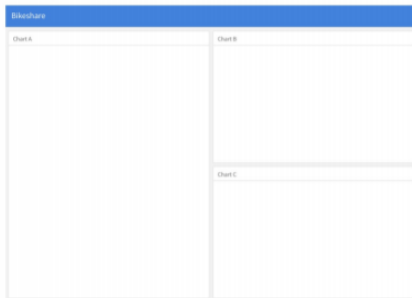- We'll use data of a Bikesharing System

Columns Width

Figure 8:

Using rows instead of columns

```
---
title: "Bikeshare"
output:
  flexdashboard::flex_dashboard:
    orientation: rows
---
```

Comparing Layouts

Figure 9:

Scrolling
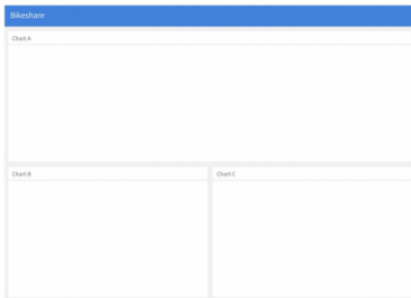
```
---
title: "Bikeshare"
output:
  flexdashboard::flex_dashboard:
    orientation: rows
    vertical_layout: scroll
---
```

Tabsets

Figure 10:

Pages

Figure 11:

Creating Menus

Graphs

- Adding Graphs: The Basics

```r
plot(trips_df$time, trips_df$rides)
```
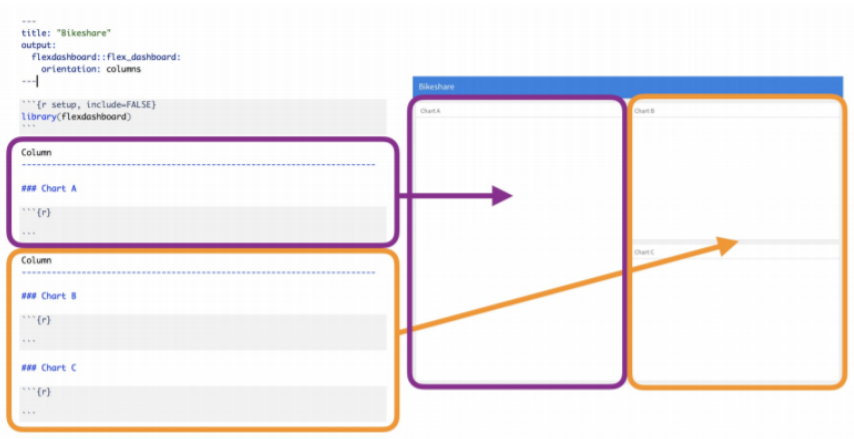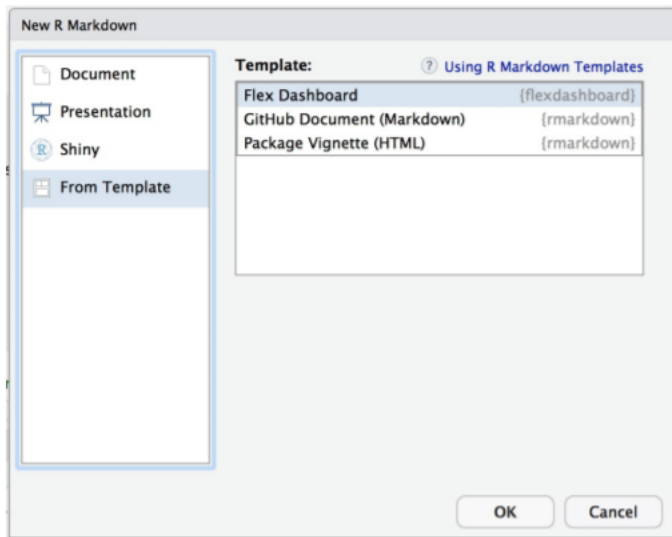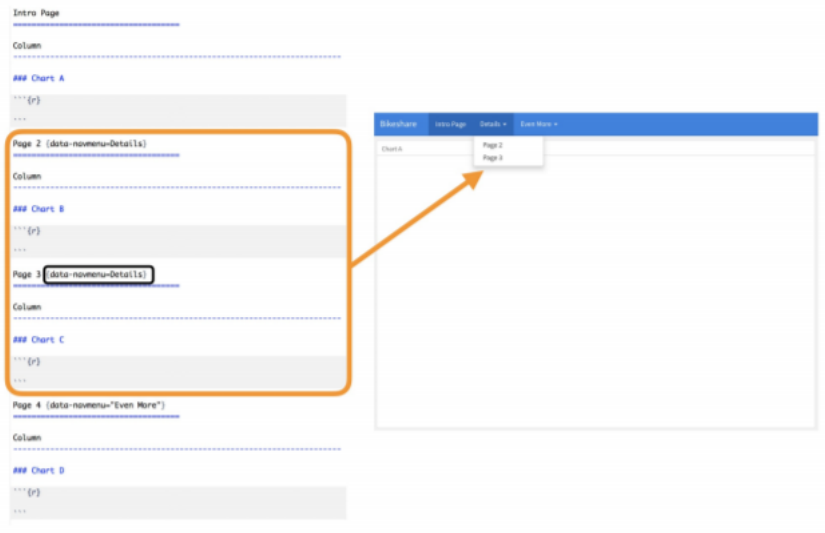
Resizing Graphs

`{r, fig.width=10, fig.height=5}`

Trips by Start Time
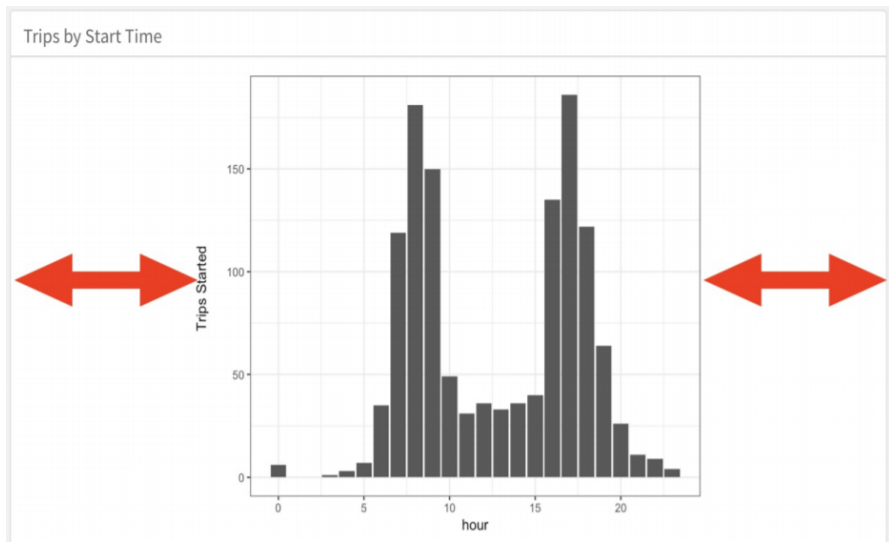
Results
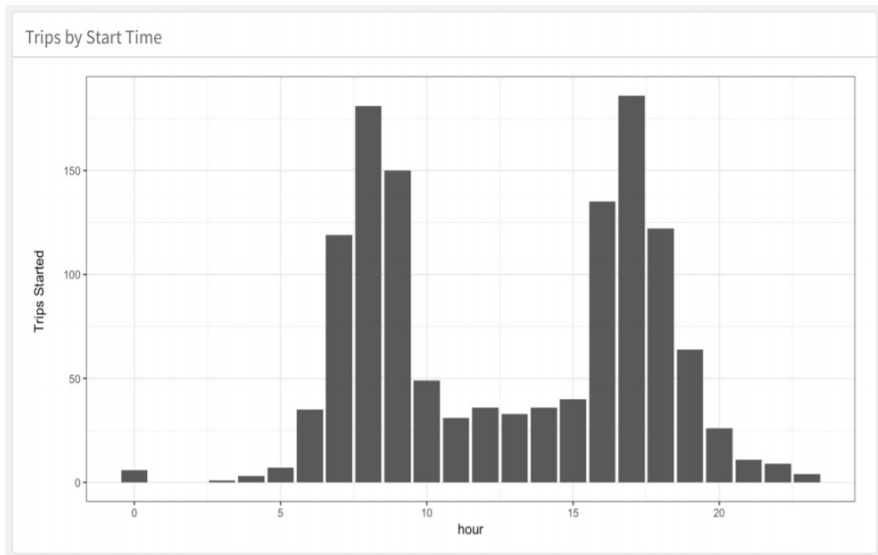
Figure 14:

Web friendly visualizations

- What is web-friendly?
  - ▶ Dynamic
  - ▶ Interactive
  - ▶ Based on html/javascript

Web-friendly options

- plotly
- highcharter
- dygraphs
- rbokeh
- ggvis

The magic of plotly

```r
library(plotly)
ggplotly(my_ggplot)
```
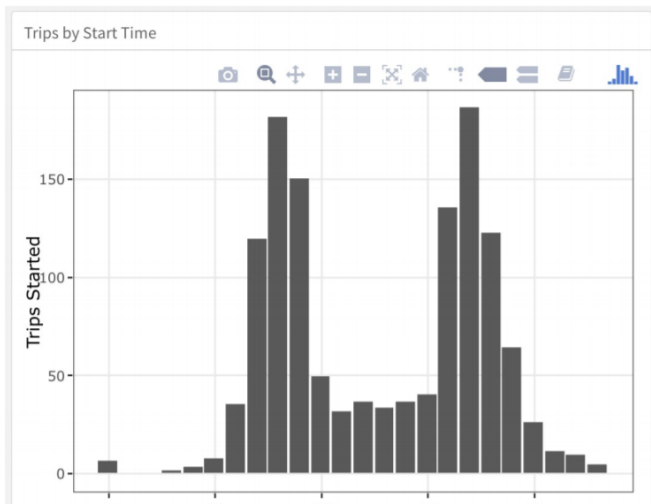
Adding Widgets

- What's an htmlwidget?
    - R packages that connect to Javascript libraries.
    - Web-friendly visualizations of all sorts.
    - Well suited to be part of a dashboard.

Learn more at: http://htmlwidgets.org

leaflet Basics

```r
library(leaflet)
leaflet() %>%
addTiles() %>%
addMarkers(lng = data_df$longitude, lat = data_df$latitude)
```

leaflet with Dataframes

```r
library(leaflet)
leaflet(data_df) %>%
addTiles() %>%
addMarkers()
```

Highilighting a value

Total Trips

```{r}
num_trips <- nrow(trips_df)

valueBox(num_trips)
```

Total Trips

```{r}
num_trips <- nrow(trips_df)

valueBox(num_trips,
         caption = 'Total Daily Trips',
         icon = 'fa-bicycle')
```

Total Trips

```{r}
num_trips <- nrow(trips_df)

valueBox(prettyNum(num_trips, big.mark = ','),
         caption = 'Total Daily Trips',
         icon = 'fa-bicycle')
```

Figure 16:

Gauges

```r
library(flexdashboard)
trips_df<- read.csv('bikeshare.csv')

num_trips <- nrow(trips_df)
short_trips <- sum(trips_df$duration_sec < 600) # under 10 min
pct_short_trips <- round(100 * short_trips / num_trips, 0)

gauge(value = pct_short_trips,
      min = 0,
      max = 100)
```

Links

```
valueBox(prettyNum(num_trips, big.mark = '
,'), caption = 'Total Daily Trips'
, icon = 'fa-bicycle' , href = '#trip-raw-data')
```

Basic Tables

```r
library(knitr)
kable(my_data_df)
```

Web-friendly Tables

```r
library(DT)
datatable(my_data_df)
```

Eliminating Row Numbering

```
library(DT)
datatable(my_data_df, rownames = FALSE)
```

Changing Rows per Page

```r
library(DT)
datatable(my_data_df, rownames = FALSE,
options = list(pageLength = 15))
```

Adding Buttons

```r
library(DT)
datatable(
my_data_df, rownames = FALSE,
extensions = 'Buttons', options = list(
dom = 'Bfrtip',buttons = c('copy',
                           'csv','excel' , 'pdf', 'print')))
```

More Information on DT

https://rstudio.github.io/DT/

Converting to a Storyboard

```
---
title: "Bike Shares Daily"
output:
  flexdashboard::flex_dashboard:
    orientation: columns
    vertical_layout: fill
    storyboard: true
---
```

Interactive Shiny Dashboard

- Why should I add Shiny?
  - Interactivity
  - Lightweight

- Why not
  - Complication
  - Hosting

Alternatives to Shiny

- Flexdashboard with Shiny
- Interactive Markdown Document

How to do it?

- Add to the YAML
    - runtime: shiny

Creating a sidebar

- Column {data-width=200 .sidebar}

Adding user inputs

Column {data-width=200 .sidebar}

```r
SliderInput("duration_slider",
            label = "Select maximum trip duration
            to display",
            min = 0,
            max = 120,
            value = 15,
            step = 5,
            dragRange = TRUE)
```