Categories

1. [Programming](https://datascienceplus.com/category/programming/)

Tags

1. [Data Visualisation](https://datascienceplus.com/tag/data-visualisation/)
2. [R Markdown](https://datascienceplus.com/tag/r-markdown/)
3. [R Programming](https://datascienceplus.com/tag/rstats/)

In this article, you learn how to make Automated Dashboard Visualizations with Ranking in R. First you need to install the `rmarkdown` package into your R library. Assuming that you installed the `rmarkdown`, next you create a new `rmarkdown` script in R.

After this you type the following code in order to create a dashboard with rmarkdown and flexdashboard:

---

title: "Dashboard visualizations in R: Ranking"

author: "Kristian Larsen"

output:

flexdashboard::flex\_dashboard:

orientation: rows

vertical\_layout: scroll

---

```{r setup, include=FALSE}

library(flexdashboard)

library(ggplot2)

library(plotly)

# Prepare data: group mean city mileage by manufacturer.

cty\_mpg <- aggregate(mpg$cty, by=list(mpg$manufacturer), FUN=mean) # aggregate

colnames(cty\_mpg) <- c("make", "mileage") # change column names

cty\_mpg <- cty\_mpg[order(cty\_mpg$mileage), ] # sort

cty\_mpg$make <- factor(cty\_mpg$make, levels = cty\_mpg$make) # to retain the order in plot.

```

Row

-----------------------------------------------------------------------

### Chart A: Ordered Bar Chart

```{r}

library(ggplot2)

theme\_set(theme\_bw())

# Draw plot

ggplot(cty\_mpg, aes(x=make, y=mileage)) +

geom\_bar(stat="identity", width=.5, fill="tomato3") +

labs(title="Ordered Bar Chart",

subtitle="Make Vs Avg. Mileage",

caption="source: mpg") +

theme(axis.text.x = element\_text(angle=65, vjust=0.6))

ggplotly(p = ggplot2::last\_plot())

```

### Chart B: Lollipop Chart

```{r}

library(ggplot2)

theme\_set(theme\_bw())

# Plot

ggplot(cty\_mpg, aes(x=make, y=mileage)) +

geom\_point(size=3) +

geom\_segment(aes(x=make,

xend=make,

y=0,

yend=mileage)) +

labs(title="Lollipop Chart",

subtitle="Make Vs Avg. Mileage",

caption="source: mpg") +

theme(axis.text.x = element\_text(angle=65, vjust=0.6))

ggplotly(p = ggplot2::last\_plot())

```

Row

-----------------------------------------------------------------------

### Cart C: Dot Plot

```{r}

library(ggplot2)

library(scales)

theme\_set(theme\_classic())

# Plot

ggplot(cty\_mpg, aes(x=make, y=mileage)) +

geom\_point(col="tomato2", size=3) + # Draw points

geom\_segment(aes(x=make,

xend=make,

y=min(mileage),

yend=max(mileage)),

linetype="dashed",

size=0.1) + # Draw dashed lines

labs(title="Dot Plot",

subtitle="Make Vs Avg. Mileage",

caption="source: mpg") +

coord\_flip()

ggplotly(p = ggplot2::last\_plot())

```

### Cart D: Slope Chart

```{r}

library(ggplot2)

library(scales)

theme\_set(theme\_classic())

# prep data

df <- read.csv("<https://raw.githubusercontent.com/selva86/datasets/master/gdppercap.csv>")

colnames(df) <- c("continent", "1952", "1957")

left\_label <- paste(df$continent, round(df$`1952`),sep=", ")

right\_label <- paste(df$continent, round(df$`1957`),sep=", ")

df$class <- ifelse((df$`1957` - df$`1952`) < 0, "red", "green")

# Plot

p <- ggplot(df) + geom\_segment(aes(x=1, xend=2, y=`1952`, yend=`1957`, col=class), size=.75, show.legend=F) +

geom\_vline(xintercept=1, linetype="dashed", size=.1) +

geom\_vline(xintercept=2, linetype="dashed", size=.1) +

scale\_color\_manual(labels = c("Up", "Down"),

values = c("green"="#00ba38", "red"="#f8766d")) + # color of lines

labs(x="", y="Mean GdpPerCap") + # Axis labels

xlim(.5, 2.5) + ylim(0,(1.1\*(max(df$`1952`, df$`1957`)))) # X and Y axis limits

# Add texts

p <- p + geom\_text(label=left\_label, y=df$`1952`, x=rep(1, NROW(df)), hjust=1.1, size=3.5)

p <- p + geom\_text(label=right\_label, y=df$`1957`, x=rep(2, NROW(df)), hjust=-0.1, size=3.5)

p <- p + geom\_text(label="Time 1", x=1, y=1.1\*(max(df$`1952`, df$`1957`)), hjust=1.2, size=5) # title

p <- p + geom\_text(label="Time 2", x=2, y=1.1\*(max(df$`1952`, df$`1957`)), hjust=-0.1, size=5) # title

# Minify theme

p + theme(panel.background = element\_blank(),

panel.grid = element\_blank(),

axis.ticks = element\_blank(),

axis.text.x = element\_blank(),

panel.border = element\_blank(),

plot.margin = unit(c(1,2,1,2), "cm"))

ggplotly(p = ggplot2::last\_plot())

```

Row

-----------------------------------------------------------------------

### Cart E: Dumbbell Plot

```{r}

library(ggplot2)

library(ggalt)

theme\_set(theme\_classic())

health <- read.csv("<https://raw.githubusercontent.com/selva86/datasets/master/health.csv>")

health$Area <- factor(health$Area, levels=as.character(health$Area)) # for right ordering of the dumbells

# health$Area <- factor(health$Area)

gg <- ggplot(health, aes(x=pct\_2013, xend=pct\_2014, y=Area, group=Area)) +

geom\_dumbbell(color="#a3c4dc",

size=0.75,

point.colour.l="#0e668b") +

scale\_x\_continuous(label=percent) +

labs(x=NULL,

y=NULL,

title="Dumbbell Chart",

subtitle="Pct Change: 2013 vs 2014",

caption="Source: <https://github.com/hrbrmstr/ggalt>") +

theme(plot.title = element\_text(hjust=0.5, face="bold"),

plot.background=element\_rect(fill="#f7f7f7"),

panel.background=element\_rect(fill="#f7f7f7"),

panel.grid.minor=element\_blank(),

panel.grid.major.y=element\_blank(),

panel.grid.major.x=element\_line(),

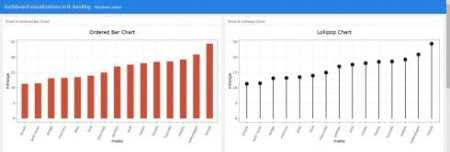
axis.ticks=element\_blank(),

legend.position="top",

panel.border=element\_blank())

plot(gg)

```

Screenshot:  
[](https://i1.wp.com/datascienceplus.com/wp-content/uploads/2018/12/V3-Ranking.jpg?ssl=1)