

M32 - Flexdashboard

LearningSpoonsR

2019-01-12

Part 1. Getting Started

Part 2. Sample Layouts

Part 3. 실습!

Part 1. Getting Started

Motivation

Previously...

- ▶ `dplyr` - 전처리 & 표
- ▶ `ggplot2` - 차트
- ▶ `rmarkdown` - 문서화 (pdf, docx, html)

Now...

- ▶ `flexdashboard`
 - ▶ Advanced `rmarkdown`
 - ▶ (정적인) html dashboard
 - ▶ Monitoring & Reporting & Presentation

Easy & interactive dashboards for R

- ▶ R Markdown을 사용하여 관련 데이터 시각화 그룹을 대시 보드로 게시!
 1. 다양한 구성 요소 지원
 2. 격자 그래픽
 3. 표 형식의 데이터
 4. 게이지 및 밸류 박스
 5. 텍스트 주석
- ▶ 장점
 1. 유연하고 쉽게 레이아웃을 지정할 수 있습니다.
 2. 브라우저를 채우고 모바일 장치에 표시되도록 조정됩니다.
 3. Shiny를 사용하여 시각화를 동적으로 구현할 수 있습니다.
 4. Semi-interactive: `htmlwidget` (`plotly` & `dygraph`)
- ▶ <https://rmarkdown.rstudio.com/flexdashboard/index.html>
- ▶ 파일 - 새파일 - Rmarkdown - Template - Flexdashboard! - Knit

Part 2. Sample Layouts

1) Chart Stack (Fill) - 상하로 fill

- ▶ 두 차트의 간단한 스택입니다.
- ▶ 데이터 높이 특성을 지정하여 세로로 더 크게 만들 수 있습니다.

Chart Stack (Fill)

This layout is a simple stack of two charts. Note that one chart or the other could be made vertically taller by specifying the `data-height` attribute.

```
1 |---  
2 | title: "Chart Stack"  
3 | output: flexdashboard::flex_dashboard  
4 | ---  
5 |  
6 | ### Chart 1  
7 |  
8 | ```{r}  
9 |  
10 | ```  
11 |  
12 | ### Chart 2  
13 |  
14 | ```{r}  
15 |  
16 | ```  
17 |  
18 |  
19 |  
20 |  
21 |  
22 |  
23 |  
24 |  
25 |  
26 |  
27 |  
28 |  
29 |
```

Chart 1

Chart 2

2) Chart Stack (Scrolling) - 상하로 scroll

- ▶ 세 개의 차트로 구성된 간단한 스택입니다.
- ▶ 모든 차트를 표시 할 수 있는 충분한 공간을 확보하기 제공하기 위해 스크롤 레이아웃이 사용됩니다. (`vertical_layout: scroll`)
- ▶ 스크롤 할 수 있기 때문에 많은 수의 차트를 여러 페이지로 구성 할 수 있습니다.
- ▶ 더 많은 수의 차트의 경우에는 **Multiple Pages**를 참조하세요.

Chart Stack (Scrolling)

This layout is a simple stack of three charts. To provide enough room to display all the charts a scrolling layout is used (`vertical_layout: scroll`). Note that because of its ability to scroll this layout could easily accommodate many more charts (although for large numbers of charts you might consider organizing them into [Multiple Pages](#)).

```
1 |---
2 title: "Chart Stack (Scrolling)"
3 output:
4   flexdashboard::flex_dashboard:
5     vertical_layout: scroll
6 ---
7
8 ### Chart 1
9
10 ```{r}
11
12
13 ### Chart 2
14
15 ```{r}
16
17
18 ### Chart 3
19
20 ```{r}
21
22
23
24
25
```

Chart 1

Chart 2

Chart 3

3) Focal Chart (Top) - main-sub 상하

- ▶ 상단의 단일 차트에 중요성을 부여하며 보조차트 2개를 표현합니다.
- ▶ **orientation: rows**를 사용합니다.
- ▶ 각 행의 **data-height** 속성을 지정하여 상대 크기를 설정합니다.

Focal Chart (Top)

This layout fills the page completely and gives prominence to a single chart at the top (with two secondary charts included below). To achieve this layout it uses `orientation: rows` and specifies `data-height` attributes on each row to establish their relative sizes.

```
1 |---
2 |title: "Focal Chart (Top)"
3 |output:
4 |  flexdashboard::flex_dashboard:
5 |    orientation: rows
6 |  ---
7 |
8 |  Row {data-height=650}
9 |  -----
10 |
11 |  ### Chart 1
12 |
13 |  ```{r}
14 |  ```
15 |
16 |  Row {data-height=350}
17 |  -----
18 |
19 |  ### Chart 2
20 |
21 |  ```{r}
22 |  ```
23 |
24 |  ### Chart 3
25 |
26 |  ```{r}
27 |  ```
28 |
```

Chart 1

Chart 2

Chart 3

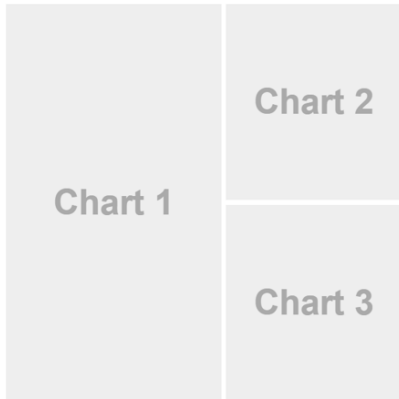
4) Focal Chart (Left) - main-sub 좌우

- ▶ 왼쪽의 단일 차트에 중요성을 부여하며 우측에 보조 차트를 제공합니다.
- ▶ `orientation: columns`는 default 입니다.
- ▶ 각 열의 `data-width` 속성을 지정하여 상대 크기를 설정합니다.

Focal Chart (Left)

This layout fills the page completely and gives prominence to a single chart on the left (with two secondary charts included to the right). Note that `data-width` attributes are specified on each column to establish their relative sizes.

```
1 |---
2 |title: "Focal Chart (Left)"
3 |output: flexdashboard::flex_dashboard
4 |---
5 |
6 |Column {data-width=600}
7 |-----
8 |
9 |### Chart 1
10 |
11 |```{r}
12 |```
13 |
14 |Column {data-width=400}
15 |-----
16 |
17 |### Chart 2
18 |
19 |```{r}
20 |```
21 |
22 |### Chart 3
23 |
24 |```{r}
25 |```
26 |
```



5) Chart Grid (2 by 2)

- ▶ 2x2 그리드 차트입니다.
- ▶ 기본 `vertical_scroll: fill`를 사용하지만, 때로는 페이지를 스크롤하는 것(`vertical_layout: scroll`)이 바람직 할 수 있습니다.
- ▶ `orientation: rows`는 차트 기준선이 수평으로 정렬되도록하는 데 사용됩니다.

Chart Grid (2x2)

This layout is a 2x2 grid of charts. This layout uses the default `vertical_scroll: fill` behavior however depending on the ideal display size for the charts it might be preferable to allow the page to scroll (`vertical_layout: scroll`). Note also that `orientation: rows` is used to ensure that the chart baselines line up horizontally.

```
1 |---
2 |title: "Row Orientation"
3 |output:
4 |  flexdashboard::flex_dashboard:
5 |    orientation: rows
6 |---
7 |
8 |Row
9 |-----
10 |
11 |### Chart 1
12 |
13 |```{r}
14 |
15 |
16 |### Chart 2
17 |
18 |```{r}
19 |
20 |
21 |Row
22 |-----
23 |
24 |### Chart 3
25 |
26 |```{r}
27 |
28 |
29 |### Chart 4
30 |
31 |```{r}
32 |
33 |
```

Chart 1

Chart 2

Chart 3

Chart 4

6) abset Column - main-sub 좌우 (tab)

- ▶ 오른쪽 열을 두 개의 탭으로 표시합니다.
- ▶ 탭은 표시 할 구성 요소 수가 많고 사용자가 스크롤하지 않아도 되게 하는 경우에 특히 유용합니다.

Tabset Column

This layout displays the right column as a set of two tabs. Tabs are especially useful when you have a large number of components to display and prefer not to require the user to scroll to access everything.

```
1  |---
2  title: "Tabset Column"
3  output: flexdashboard::flex_dashboard
4  |---
5
6  Column
7  -----
8
9  ### Chart 1
10
11  ```{r}
12  ```
13
14  Column {.tabset}
15  -----
16
17  ### Chart 2
18
19  ```{r}
20  ```
21
22  ### Chart 3
23
24  ```{r}
25  ```
26
```



7) Tabset Row - main-sub 상하 (tab)

- ▶ 하단 행을 두 개의 탭으로 표시합니다.
- ▶ `{.tabset-fade}` 속성은 탭을 전환 할 때 페이드 인 / 아웃 효과를 사용하는 데에도 사용됩니다.

Tabset Row

This layout displays the bottom row as a set of two tabs. Note that the `{.tabset-fade}` attribute is also used to enable a fade in/out effect when switching tabs.

```
1 |---
2 |title: "Tabset Row"
3 |output:
4 |  flexdashboard::flex_dashboard:
5 |    orientation: rows
6 |  ---
7 |
8 |  Row
9 |  -----
10 |
11 |  ### Chart 1
12 |
13 |  ```{r}
14 |
15 |
16 |  Row {.tabset .tabset-fade}
17 |  -----
18 |
19 |  ### Chart 2
20 |
21 |  ```{r}
22 |
23 |
24 |  ### Chart 3
25 |
26 |  ```{r}
27 |
28 |
```



8) Multiple Pages

- ▶ 이 레이아웃은 레벨 1 마크 다운 헤더 (=====)를 사용하여 여러 페이지를 정의합니다.
- ▶ 두 번째 페이지에서는 orientation을 바꾼게 보이시나요?
- ▶ 사용자 정의 **data-width** 및 **data-height** 특성이있는 여러 열과 행을 사용하는 방법도 설명합니다.

Multiple Pages

This layout defines multiple pages using a level 1 markdown header (=====). Each page has its own top-level navigation tab. Further, the second page uses a distinct orientation via the `data-orientation` attribute. The use of multiple columns and rows with custom `data-width` and `data-height` attributes is also demonstrated.

```

1 |---
2 |title: "Multiple Pages"
3 |output: flexdashboard::flex_dashboard
4 |---
5 |
6 |Page 1
7 |=====
8 |
9 |Column {data-width=600}
10 |-----
11 |
12 |### Chart 1
13 |
14 |```{r}
15 |
16 |Column {data-width=400}
17 |-----
18 |
19 |### Chart 2
20 |
21 |```{r}
22 |
23 |
24 |### Chart 3
25 |
26 |```{r}
27 |
28 |
29 |Page 2 {data-orientation=rows}
30 |=====
31 |
32 |Row {data-height=600}
33 |-----
34 |
35 |### Chart 1
36 |
37 |```{r}
38 |
39 |
40 |Row {data-height=400}
41 |-----
42 |
43 |### Chart 2
44 |
45 |```{r}
46 |
47 |
48 |### Chart 3
49 |
50 |```{r}
51 |
52 |
53 |

```

9) Input Sidebar

- ▶ flexdashboard 페이지에 사이드 바를 추가하는 방법을 보여줍니다
- ▶ (샤이니 기반 대시 보드는 종종 사용자 입력 컨트롤을 사이드 바에 표시합니다).
- ▶ 사이드 바를 포함 시키려면 `.sidebar` 클래스를 레벨 2 헤더에 추가하십시오
(-----).

Input Sidebar

This layout demonstrates how to add a sidebar to a flexdashboard page (Shiny-based dashboards will often present user input controls in a sidebar). To include a sidebar you add the `.sidebar` class to a level 2 header (-----):

```
1 |--
2 title: "Sidebar"
3 output: flexdashboard::flex_dashboard
4 runtime: shiny
5 ---
6
7 Inputs {.sidebar}
8 -----
9
10 ```{r}
11 # shiny inputs defined here
12 ```
13
14 Column
15 -----
16
17 ### Chart 1
18
19 ```{r}
20
21
22 ### Chart 2
23
24 ```{r}
25
26
```

Chart 1

Chart 2

10) Input Sidebar (Global) - Multiple Pages + Sidebar

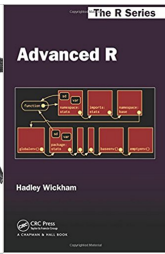
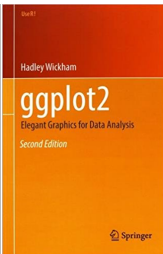
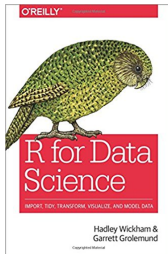
- ▶ 여러 페이지에서 사이드 바를 공통으로 (즉, 모든 페이지에 대해) 표시 할 수 있습니다.
- ▶ 전역(Global) 사이드 바를 포함 시키려면 레벨 1 헤더 (=====)에 **.sidebar**를 추가하세요.

Input Sidebar (Global)

If you have a layout that uses [Multiple Pages](#) you may want the sidebar to be global (i.e. present for all pages). To include a global sidebar you add the `.sidebar` class to a level 1 header (=====):

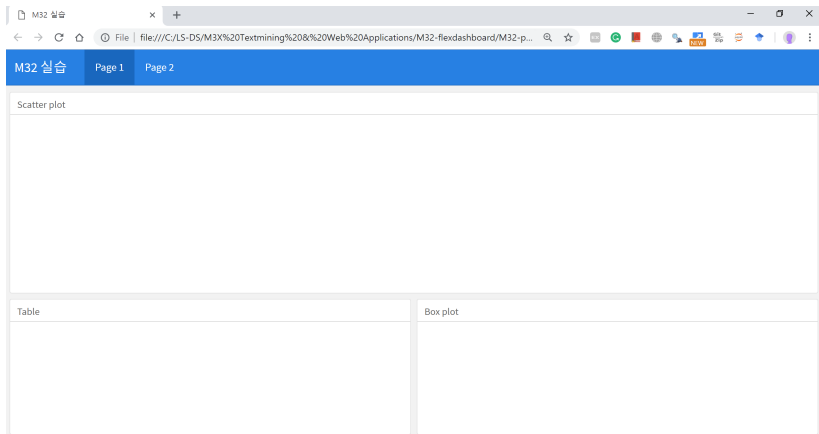
```
1 |---
2 |title: "Sidebar for Multiple Pages"
3 |output: flexdashboard::flex_dashboard
4 |runtime: shiny
5 |---
6 |
7 |Sidebar {.sidebar}
8 |=====
9 |
10 |````{r}
11 |# shiny inputs defined here
12 |````
13 |
14 |Page 1
15 |=====
16 |
17 |### Chart 1
18 |
19 |````{r}
20 |````
21 |
22 |Page 2
23 |=====
24 |
25 |### Chart 2
26 |
27 |````{r}
28 |````
29 |
```

많이 익숙해지셨나요? R관련 서적을 몇 개 소개해드립니다.



Part 3. 실습!

Step 1. Create blank dashboard

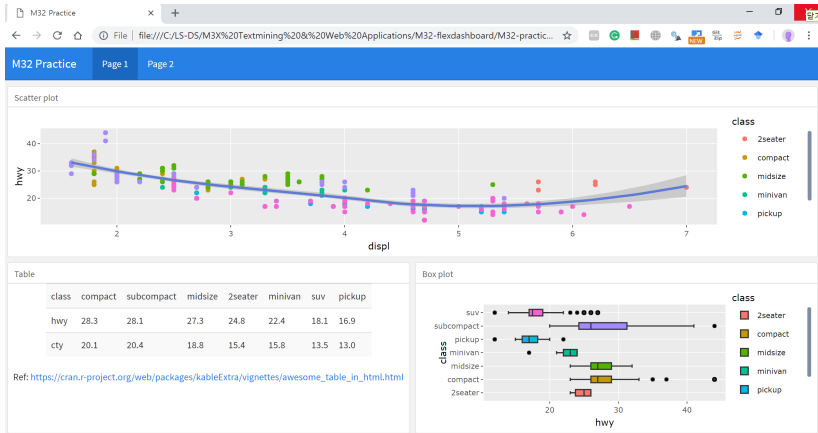


M32 실습

Page 1 Page 2

Scatter plot + facet

Step 2. Fill in the blanks






```
"Express Yourself"
```

```
## [1] "Express Yourself"
```