

Facets

Prashanth.S (19MID0020)

Code

The image displays four sequential screenshots of the RStudio code editor, illustrating the step-by-step implementation of faceted plots. Each screenshot shows a 'File list' on the left, a code editor in the center, and a 'Description' panel on the right.

- Screenshot 1:** The code defines a scatter plot of 'tip' vs 'total_bill' from the 'tips' dataset. It then creates two faceted versions: 'horizontal' (faceted by sex) and 'vertical' (faceted by sex). A third figure, 'fig3', is created by combining these two faceted plots using 'facet_grid(sex ~ day)' and applying a custom theme with red background and bold text.
- Screenshot 2:** The code introduces 'dplyr' and 'mutate' to create a new 'sex' variable for the 'tips' dataset, converting 'Male' to 'Men' and 'Female' to 'Women'. This updated dataset is used to create a new faceted plot 'fig3'.
- Screenshot 3:** The code uses 'mutate' to create a 'smoker' variable based on the 'sex' variable. It then creates four faceted plots: 'fig1' (faceted by sex), 'fig2' (faceted by sex and smoker), 'fig3' (faceted by sex and smoker with 'free.y' scales), and 'fig4' (faceted by sex and smoker with 'free' scales).
- Screenshot 4:** The code loads the 'ChickWeight' dataset and creates two growth curves: 'p1' (a line plot of weight over time for individual chicks) and 'p2' (a smoothed line plot of weight over time for individual chicks).

Mam, I implemented the code in moodle as-well as in RStudio also. In-order to avoid confusion with continuous output, I am including my implementation in RStudio also.

Facets

Importing the libraries

```
library(reshape2)
library(ggplot2)
library(dplyr)
```

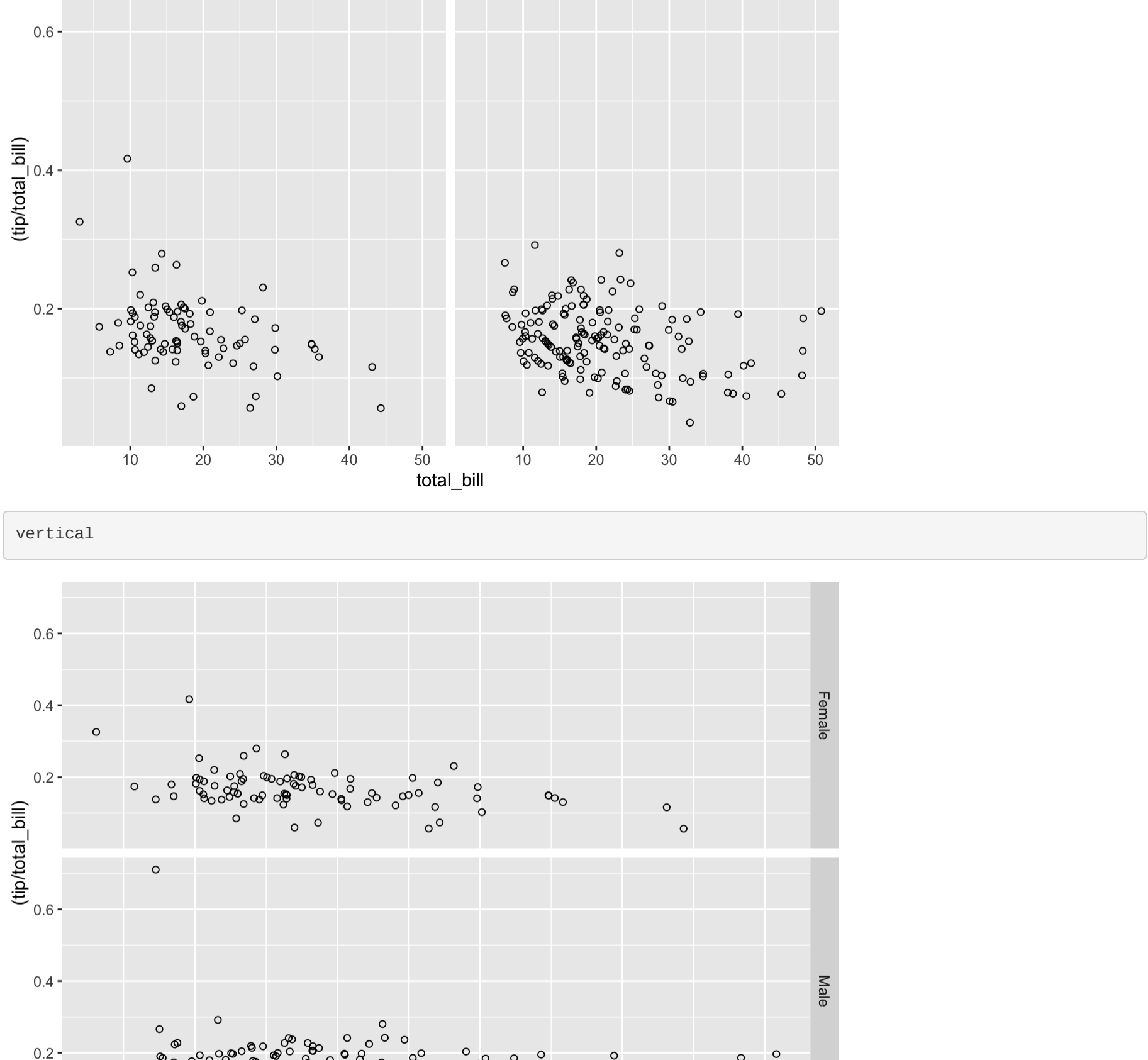
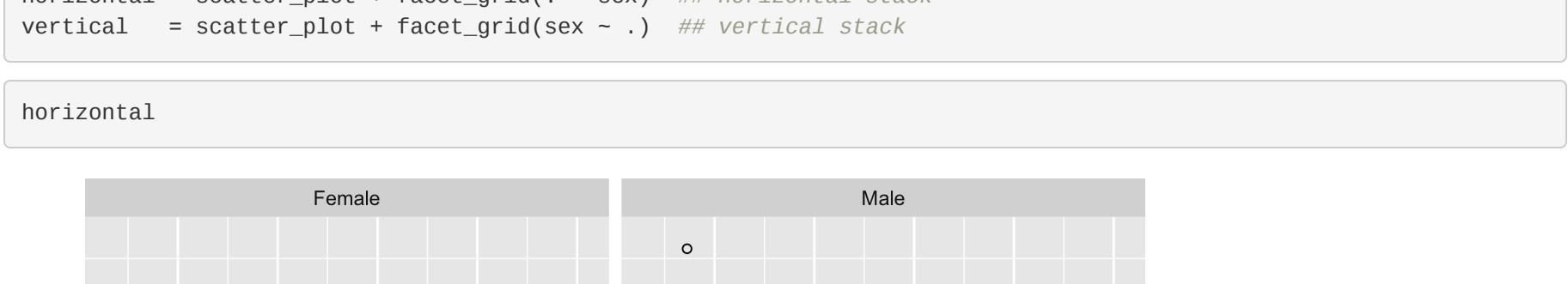
```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##   intersect, setdiff, setequal, union
```

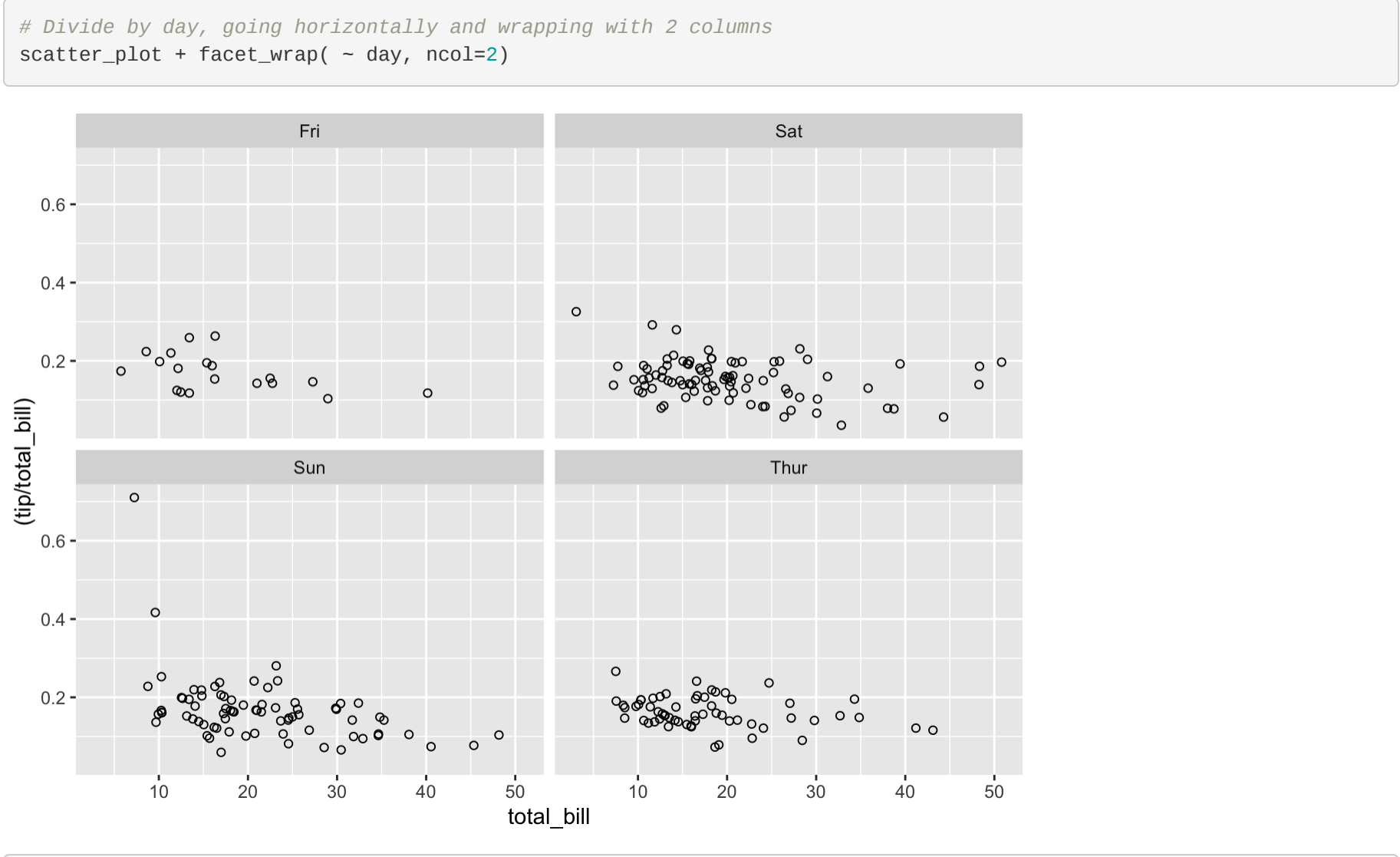
```
scatter_plot = ggplot(tips, aes(x=total_bill, y = (tip/total_bill))) + geom_point(shape=1)
```

```
## FACET GRID
horizontal = scatter_plot + facet_grid(. ~ sex) ## horizontal stack
vertical = scatter_plot + facet_grid(sex ~ .) ## vertical stack
```



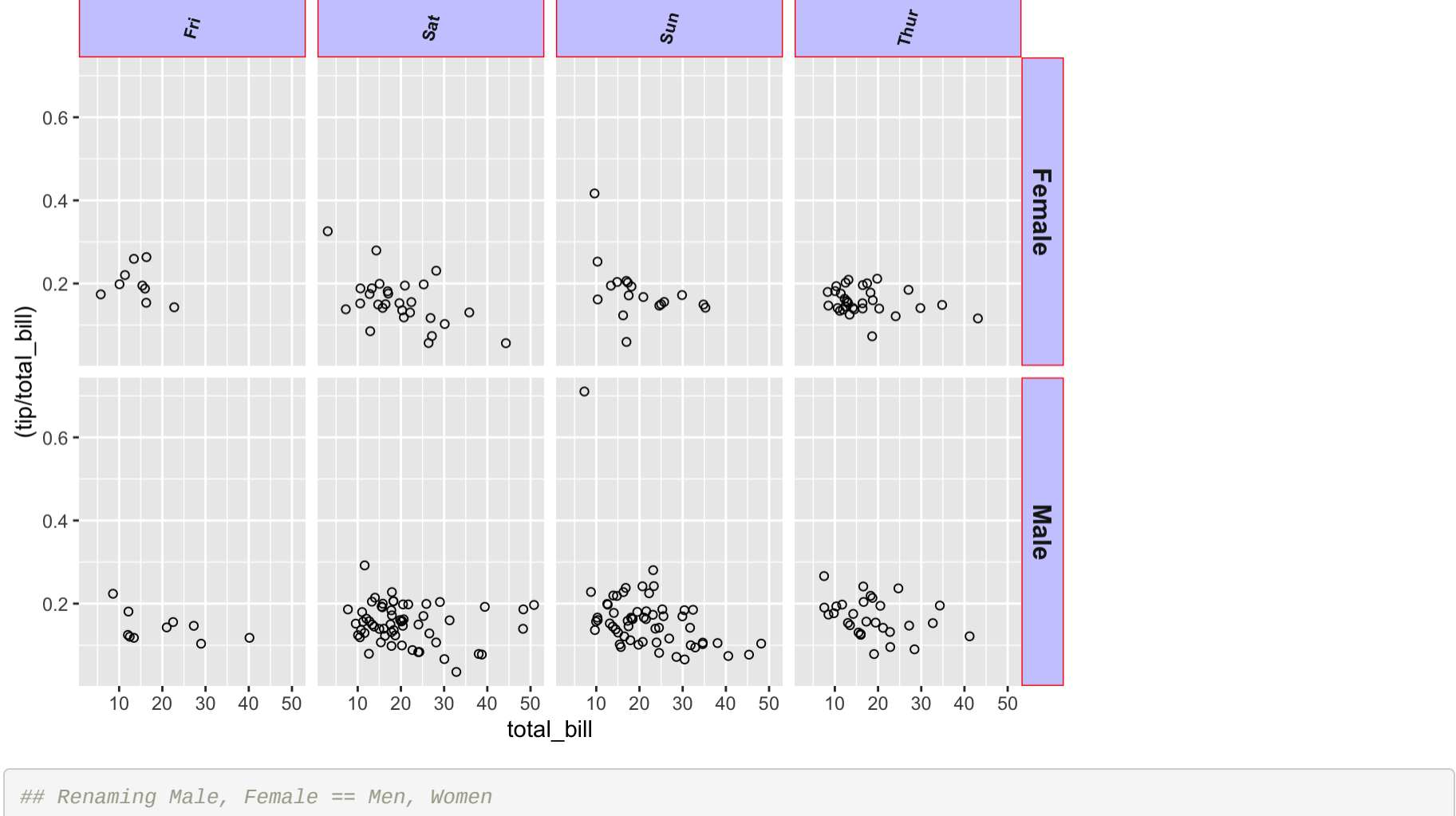
Facet Wrap

```
# Divide by day, going horizontally and wrapping with 2 columns
scatter_plot + facet_wrap(~ day, ncol=2)
```



```
## sex -> vertical and day -> horizontal
fig3 = scatter_plot + facet_grid(sex ~ day) +
  theme(
    strip.text.x = element_text(size=8, angle=75, face='bold'),
    strip.text.y = element_text(size=12, face='bold'),
    strip.background = element_rect(color='red', fill='#CCCCFF')
  )
```

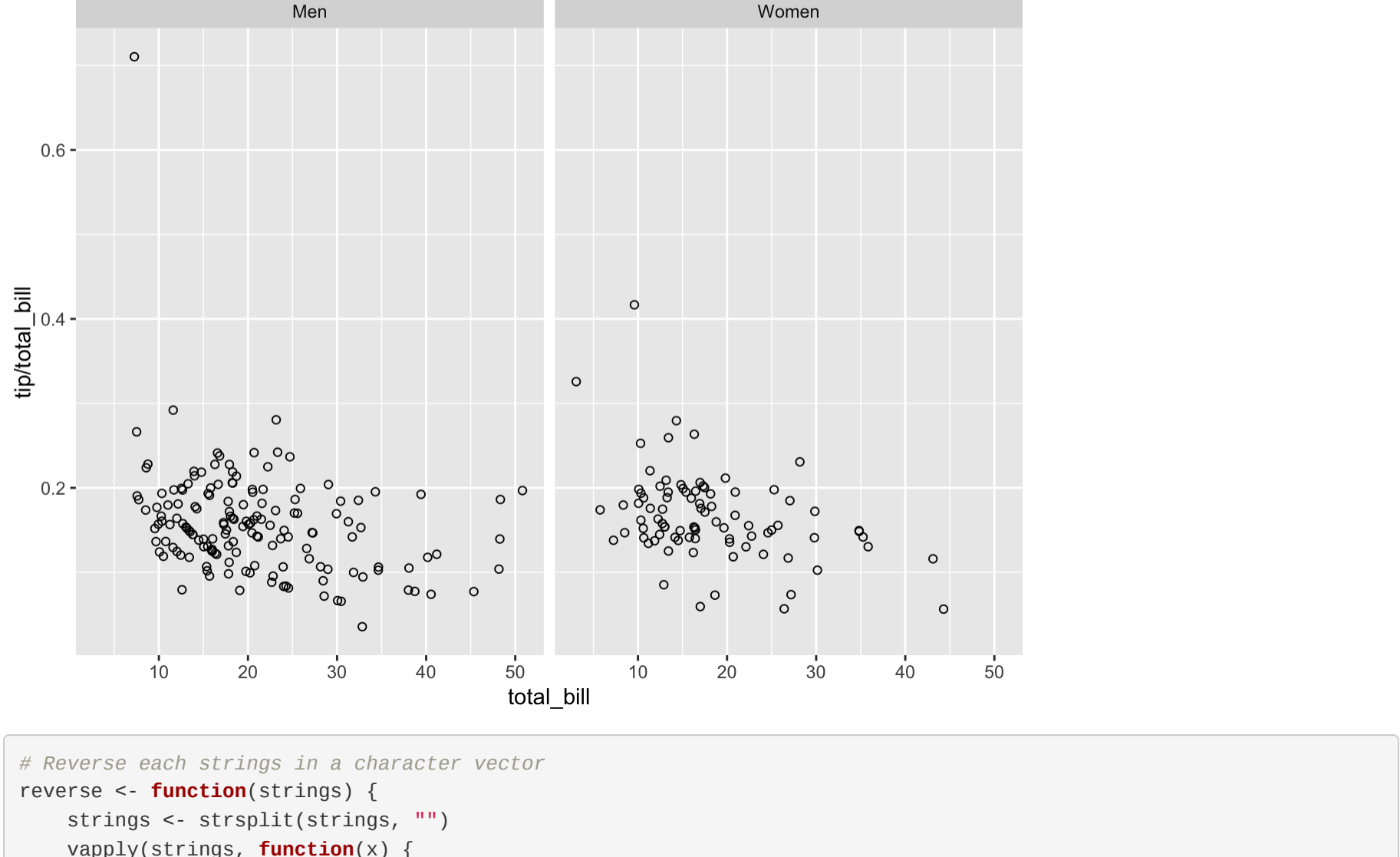
fig3



```
## Renaming Male, Female == Men, Women
df = tips
df = mutate(df, sex = ifelse(sex == 'Male', "Men", "Women"))
```

Reversing the string

```
sp2 <- ggplot(df, aes(x=total_bill, y=tip/total_bill)) + geom_point(shape=1)
sp2 + facet_grid(. ~ sex)
```



```
# Reverse each strings in a character vector
reverse <- function(strings) {
  strings <- strsplit(strings, "")
  vapply(strings, function(x) {
    paste(rev(x), collapse = "")
  }, FUN.VALUE = character(1))
}
```

```
sp2 + facet_grid(. ~ sex, labeller=labeler(sex = reverse))
```



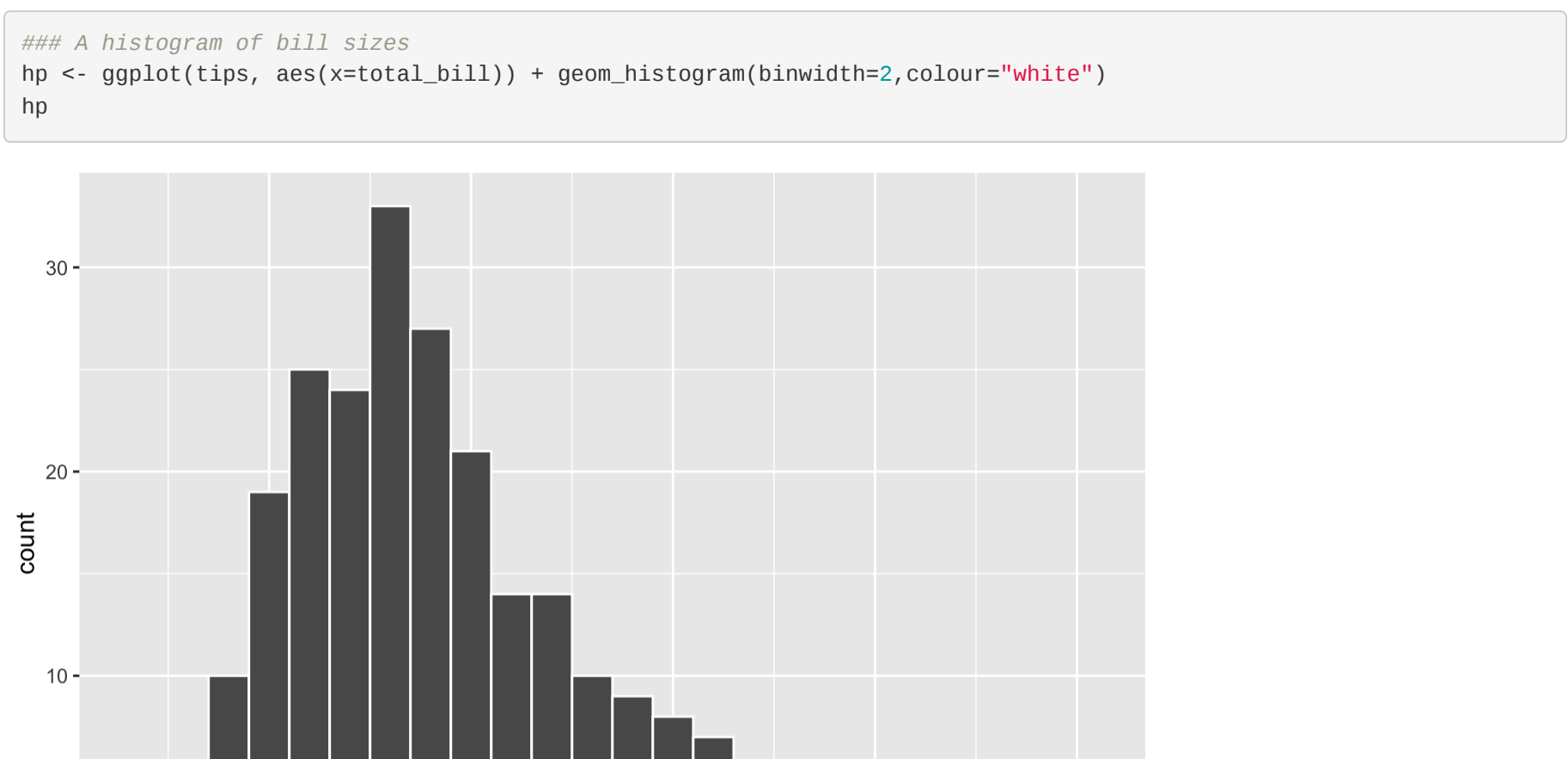
```
## sex -> vertical and day -> horizontal
fig3 = scatter_plot + facet_grid(sex ~ day) +
  theme(
    strip.text.x = element_text(size=8, angle=75, face='bold'),
    strip.text.y = element_text(size=12, face='bold'),
    strip.background = element_rect(color='red', fill='#CCCCFF')
  )
```

fig3

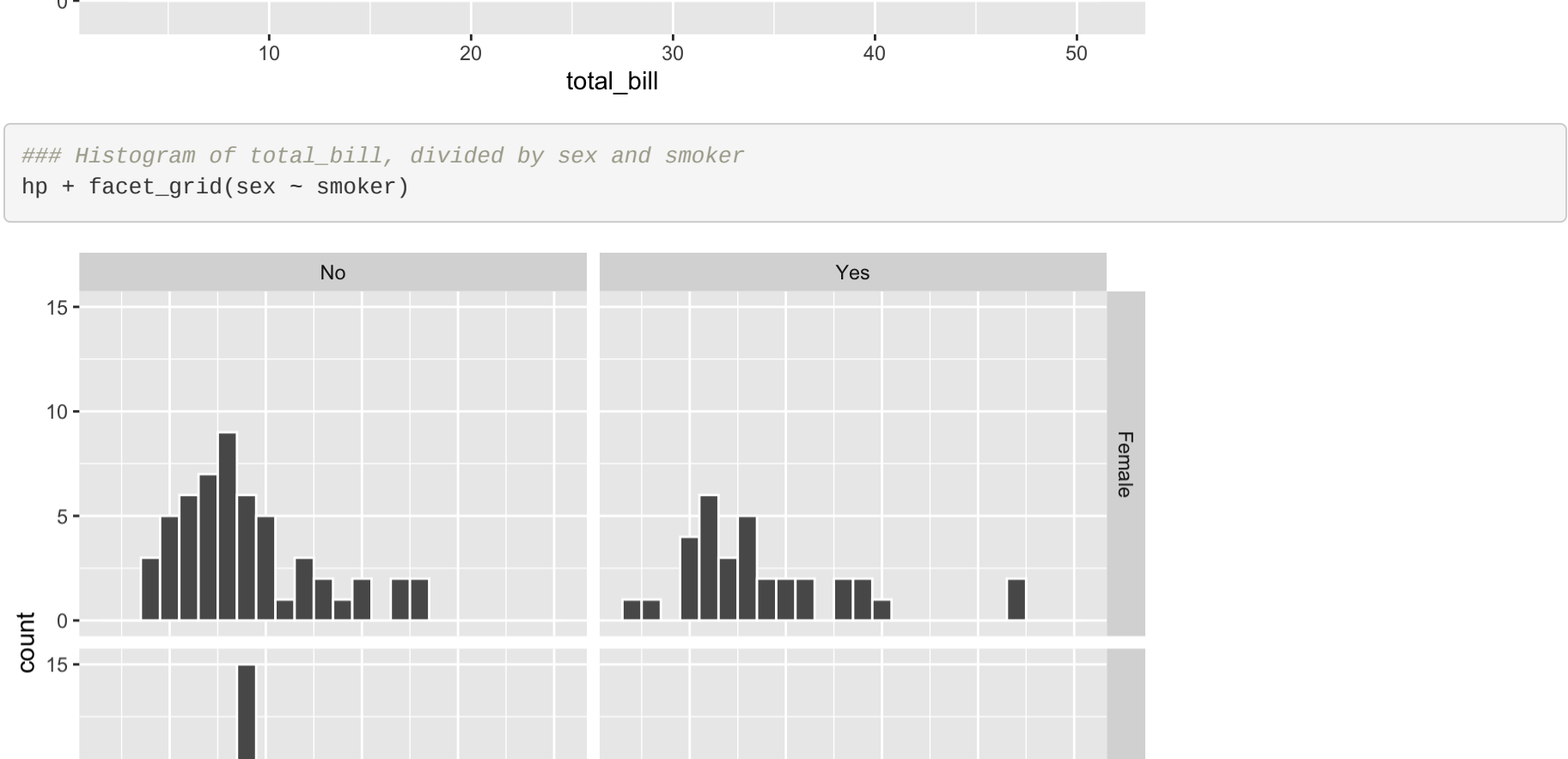


Free Scales

```
### A histogram of bill sizes
hp <- ggplot(tips, aes(x=total_bill)) + geom_histogram(binwidth=2, colour="white")
hp
```



```
### Histogram of total_bill, divided by sex and smoker
hp + facet_grid(sex ~ smoker)
```



```
### Same as above, with scales="free_y"
```

```
hp + facet_grid(sex ~ smoker, scales="free_y")
```



```
### With panels that have the same scaling, but different range (and therefore different physical sizes)
```

```
hp + facet_grid(sex ~ smoker, scales="free", space="free")
```

