

# Gergely Daróczy

## Unified plots

### Contents

<b>Options</b>	<b>3</b>
<b>Default options</b>	<b>4</b>
Histogram . . . . .	4
Base R plot . . . . .	4
lattice . . . . .	6
ggplot2 . . . . .	6
Barplot . . . . .	7
Base R plot . . . . .	7
lattice . . . . .	10
ggplot2 . . . . .	10
Points . . . . .	10
Base R plot . . . . .	10
lattice . . . . .	11
ggplot2 . . . . .	11
Grouped plot . . . . .	11
Base R plot . . . . .	11
lattice . . . . .	14
ggplot2 . . . . .	14
Facets . . . . .	14
Base R plot . . . . .	14
lattice . . . . .	16
ggplot2 . . . . .	16

Boxplot . . . . .	17
Base R plot . . . . .	17
lattice . . . . .	20
ggplot2 . . . . .	20
Lines . . . . .	20
Base R plot . . . . .	20
lattice . . . . .	21
ggplot2 . . . . .	21
<b>Custom options</b>	<b>21</b>
Histogram . . . . .	23
Base R plot . . . . .	23
lattice . . . . .	24
ggplot2 . . . . .	24
Barplot . . . . .	24
Base R plot . . . . .	24
lattice . . . . .	29
ggplot2 . . . . .	34
Points . . . . .	34
Base R plot . . . . .	34
lattice . . . . .	34
ggplot2 . . . . .	34
Grouped plot . . . . .	34
Base R plot . . . . .	34
lattice . . . . .	34
ggplot2 . . . . .	34
Facets . . . . .	34
Base R plot . . . . .	34
lattice . . . . .	44
ggplot2 . . . . .	44
Boxplot . . . . .	44

Base R plot . . . . .	44
lattice . . . . .	44
ggplot2 . . . . .	44
Lines . . . . .	44
Base R plot . . . . .	44
lattice . . . . .	44
ggplot2 . . . . .	44

This document demonstrates the features of unifying plots in **pander**. First, if you do not want to deal with styling each of your images, just activate it: `evalsOptions('graph.unify', TRUE)`.

*Note:* please install `lattice` and `ggplot2` on your computer prior to trying to run all examples of this document.

## Options

There are a bunch of options you might want to check out, these are:

- `graph.nomargin`
- `graph.fontfamily`
- `graph.fontcolor`
- `graph.fontsize`
- `graph.grid`
- `graph.grid.minor`
- `graph.grid.color`
- `graph.grid.lty`
- `graph.bboxes`
- `graph.legend.position`
- `graph.background`
- `graph.panel.background`
- `graph.colors`
- `graph.color.rnd`
- `graph.axis.angle`
- `graph.symbol`

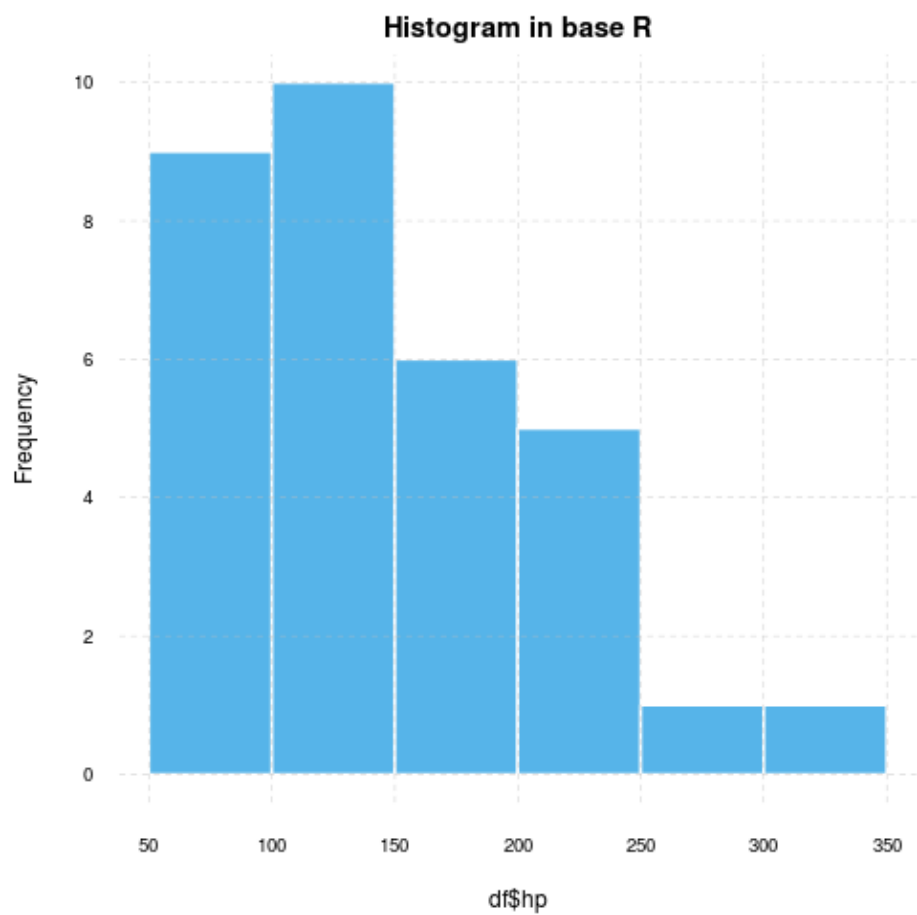
Find more details on [pander's homepage](#).

## Default options

Not touching the above ones, let us check out how different plots look like by calling `base R` graphing function, `lattice` or `ggplot2`!

### Histogram

Base R plot



WARNING<sup>1</sup>

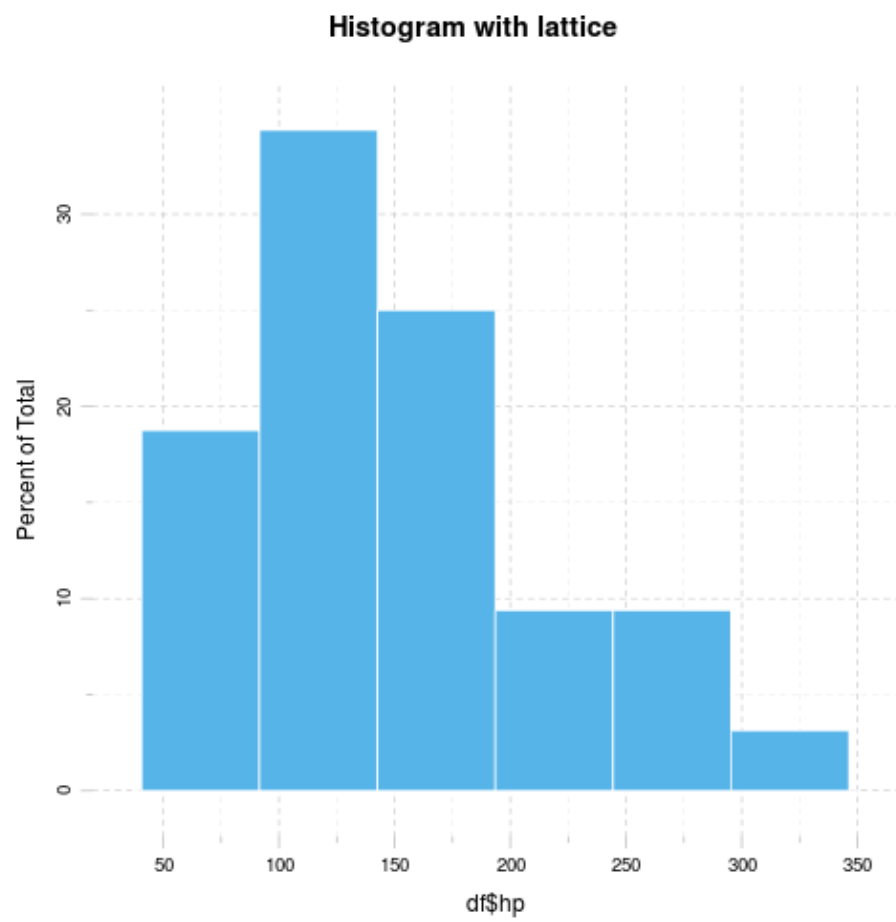


Figure 1:

`lattice`

`ggplot2`

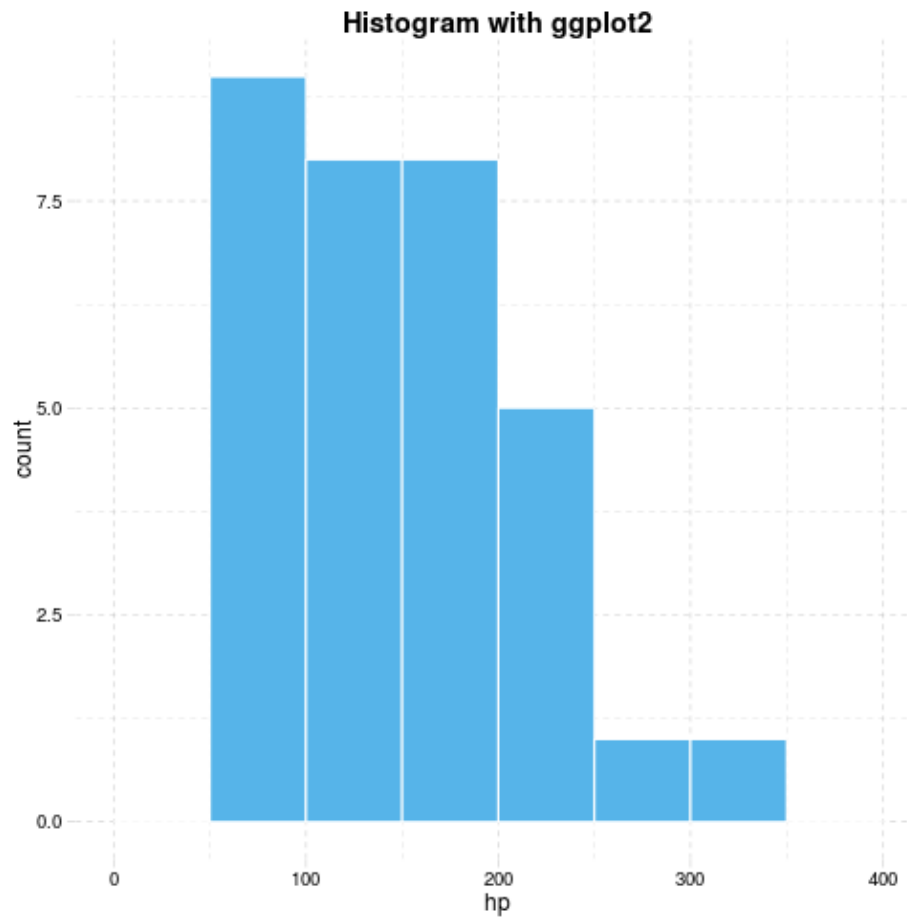


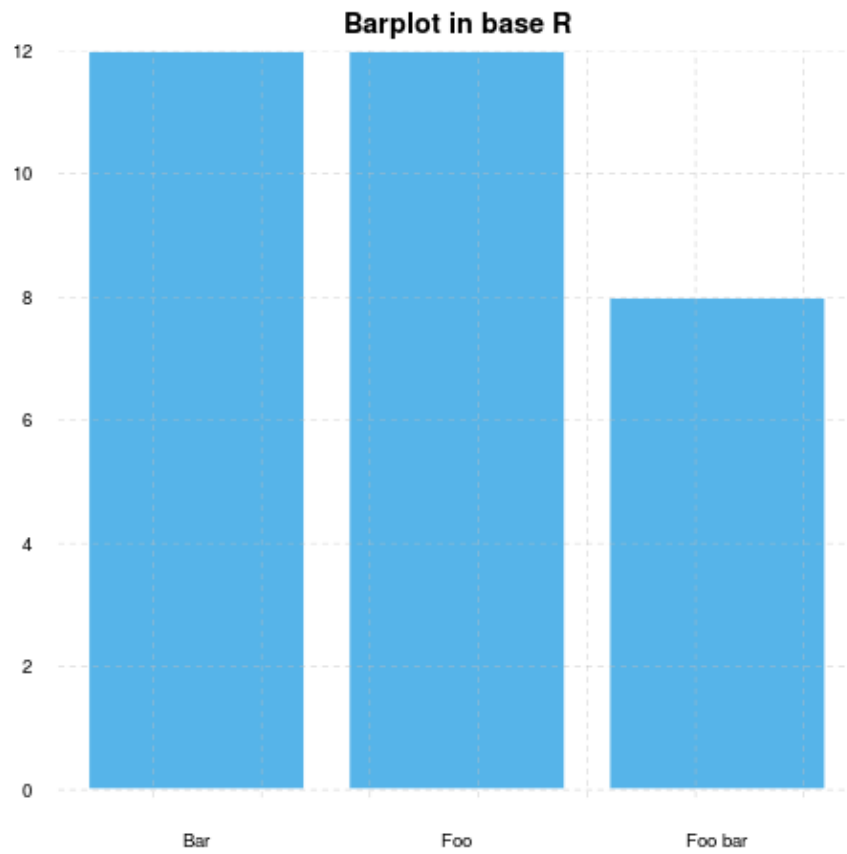
Figure 2:

---

<sup>1</sup>Applying default formatting to image is somehow compromised (the result could differ from what you specified in `panderOptions`). Hints: printing `lattice/ggplot2` is not needed and tweaking `base` plots with `par` might have some side-effects!

## Barplot

Base R plot



WARNING<sup>2</sup>

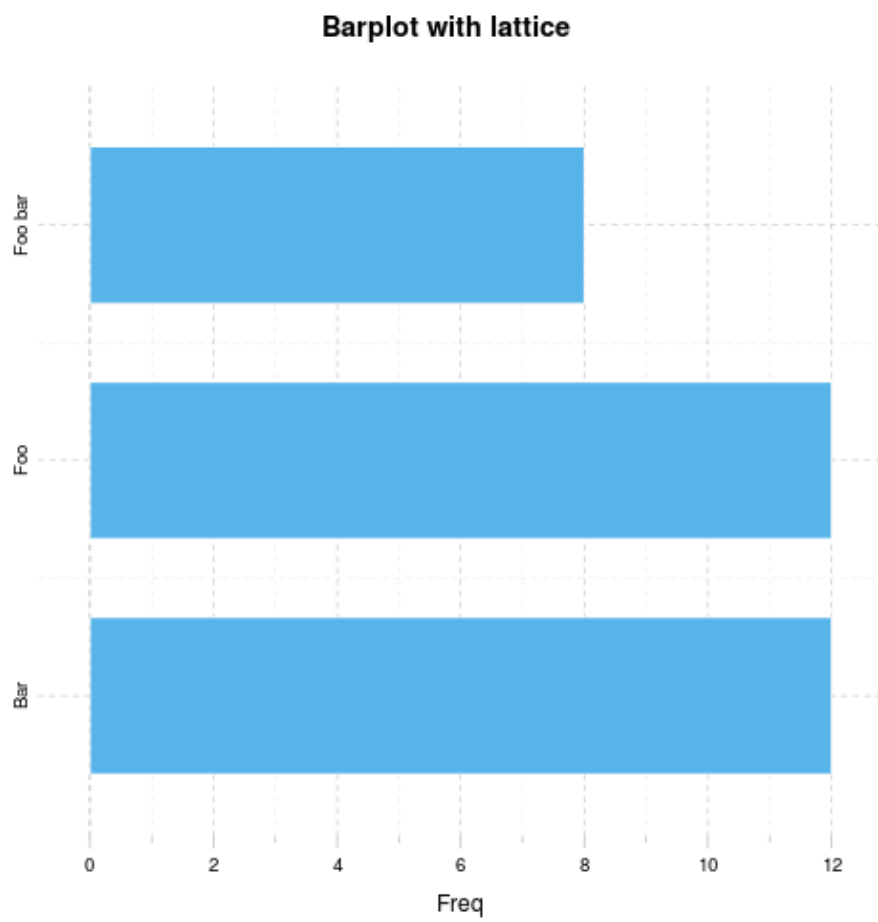


Figure 3:



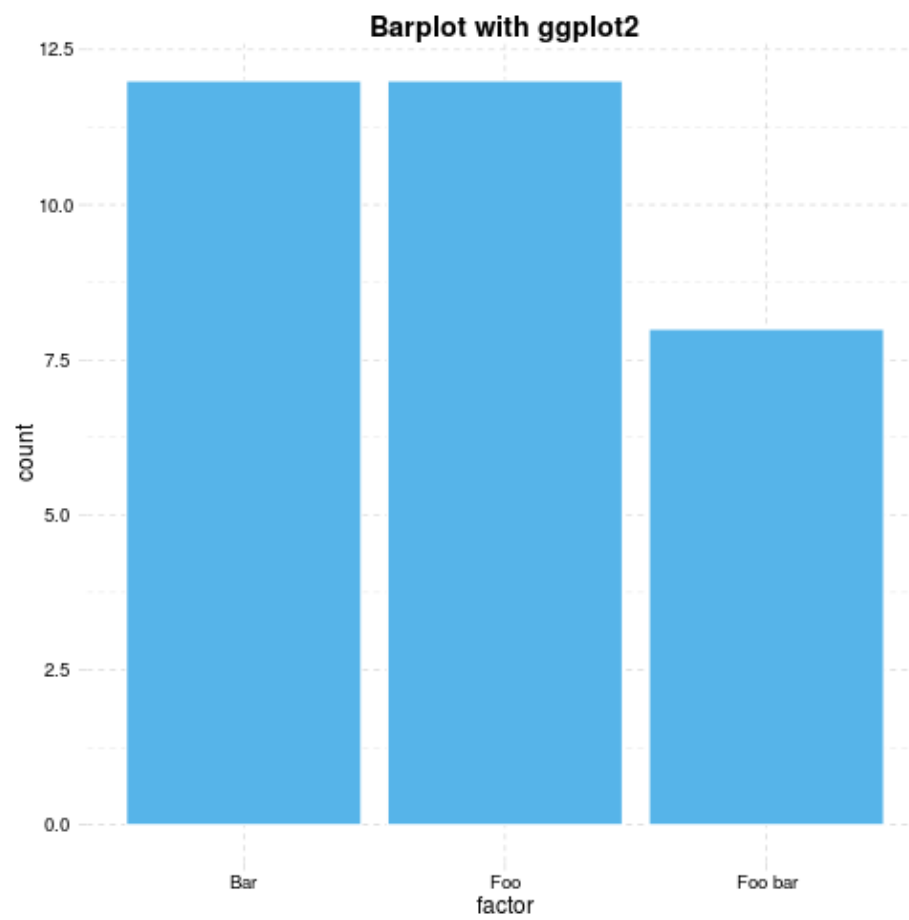


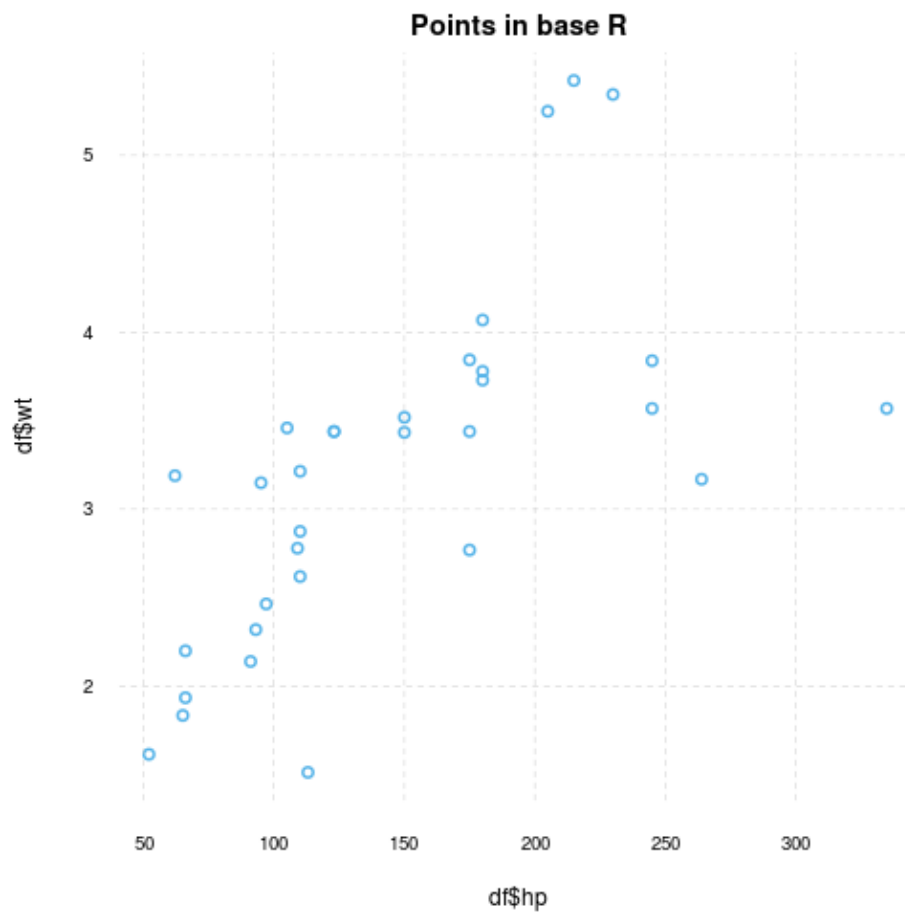
Figure 4:

`lattice`

`ggplot2`

Points

Base R plot



**WARNING**<sup>3</sup>

<sup>2</sup>Applying default formatting to image is somehow compromised (the result could differ from what you specified in `panderOptions`). Hints: printing `lattice/ggplot2` is not needed and tweaking `base` plots with `par` might have some side-effects!

<sup>3</sup>Applying default formatting to image is somehow compromised (the result could differ from what you specified in `panderOptions`). Hints: printing `lattice/ggplot2` is not needed and tweaking `base` plots with `par` might have some side-effects!

lattice

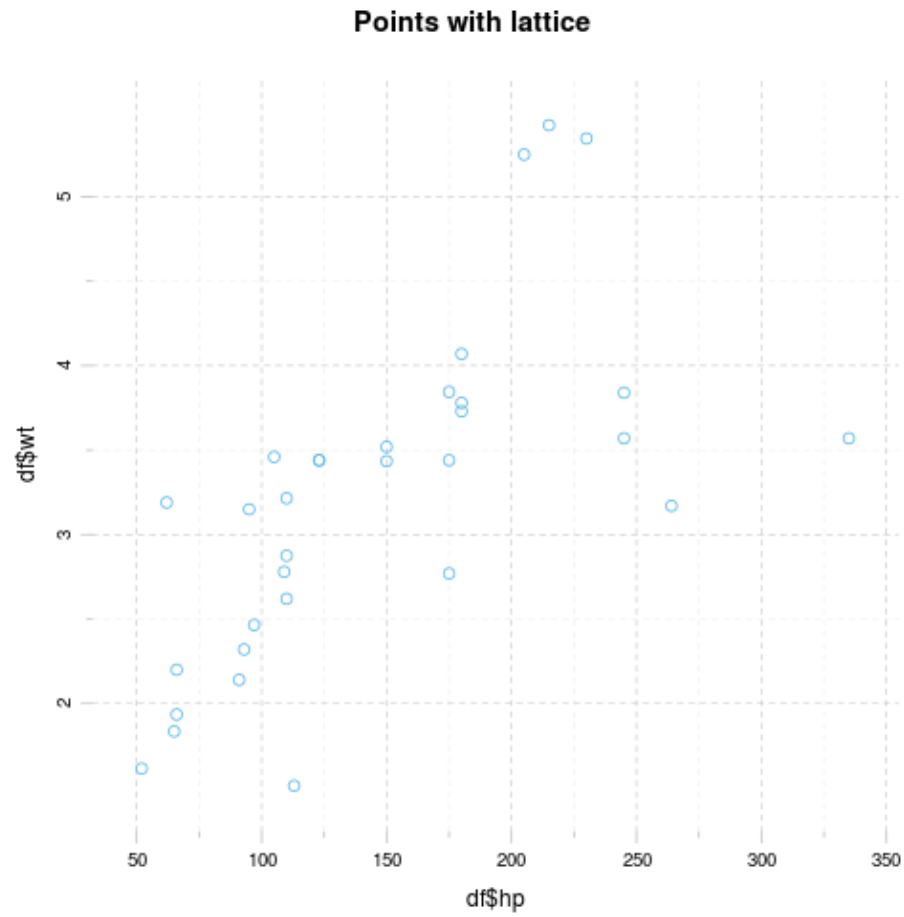


Figure 5:

ggplot2

**Grouped plot**

**Base R plot**

I have no idea how to do that besides manually adding `points`.

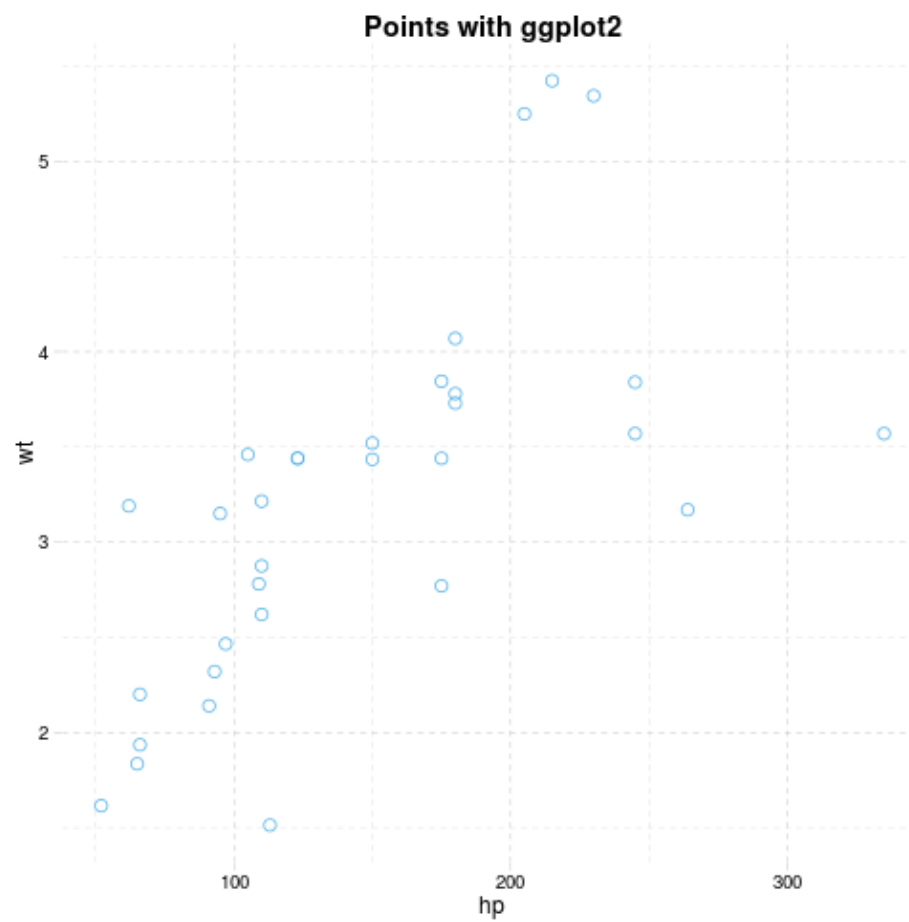


Figure 6:

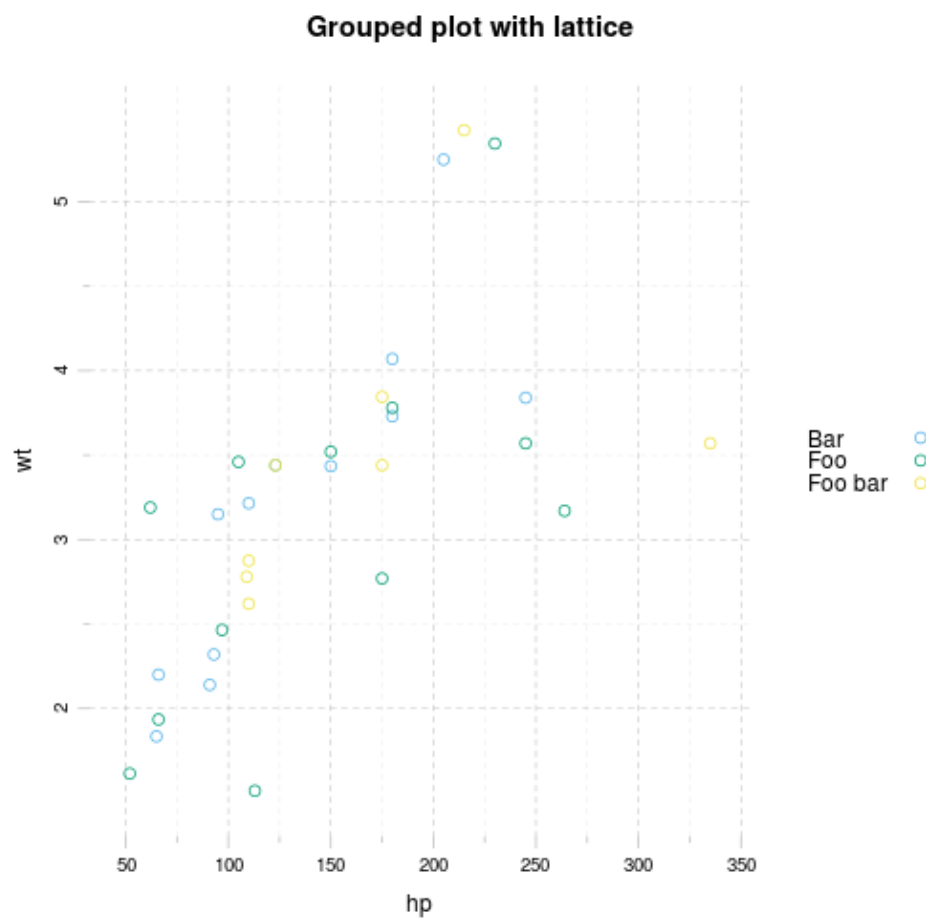


Figure 7:

`lattice`

`ggplot2`

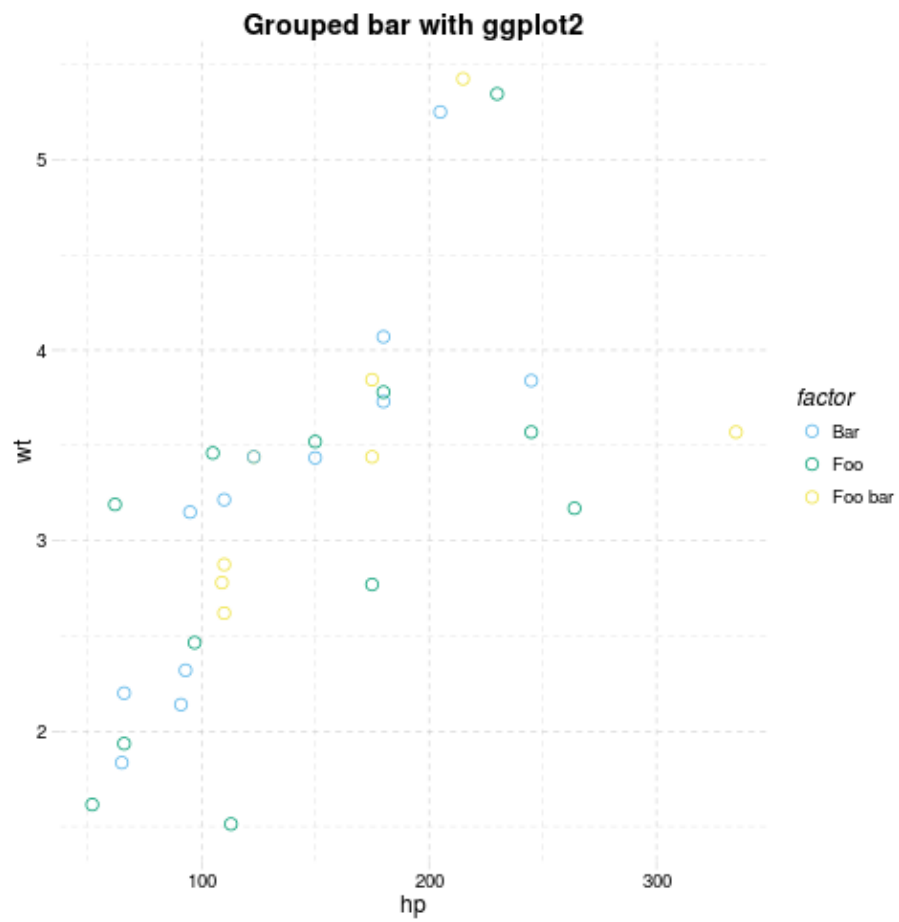


Figure 8:

## Facets

### Base R plot

I have no idea how to do that besides `par(mfrow=c(foo, bar))`.

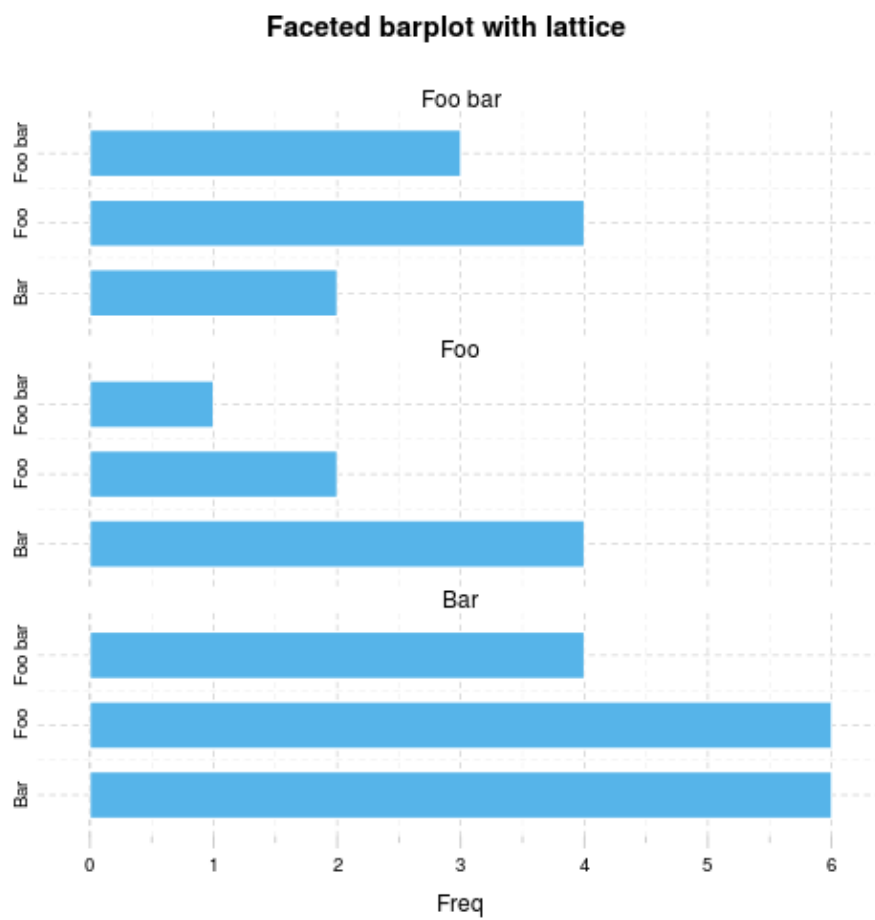


Figure 9:

lattice

ggplot2

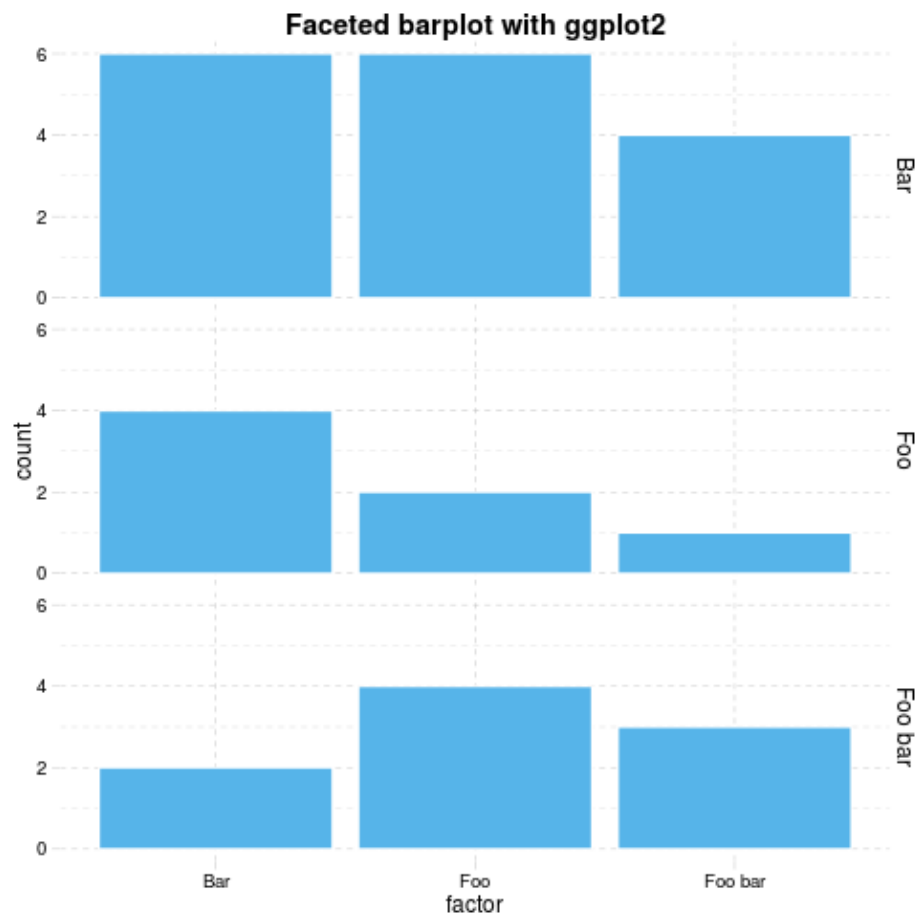
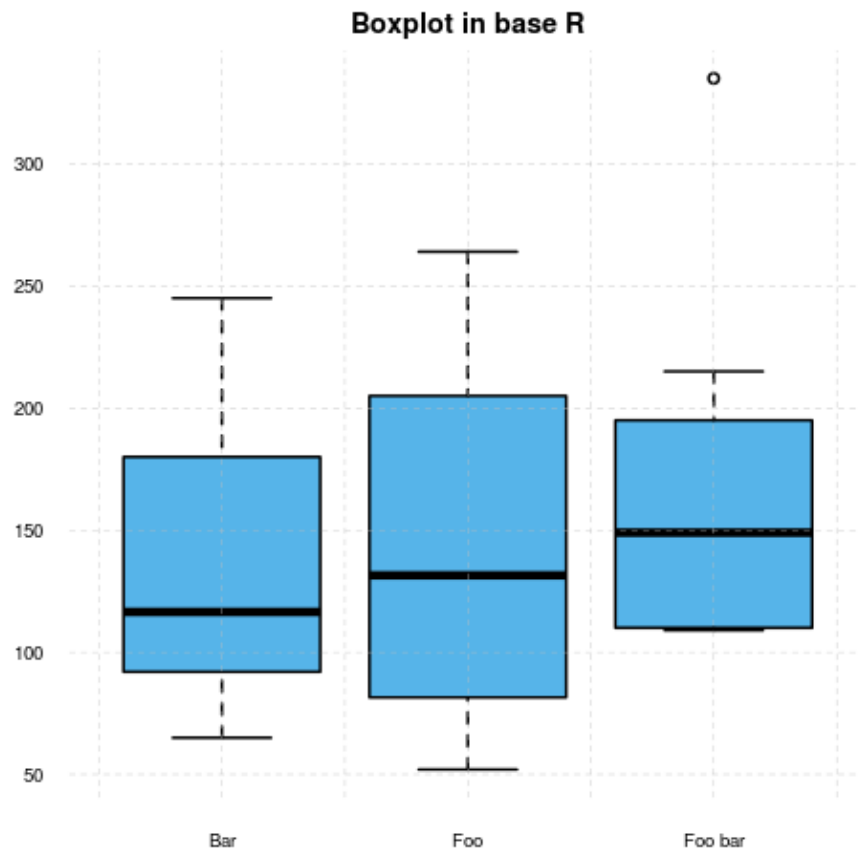


Figure 10:



## Boxplot

Base R plot



WARNING<sup>4</sup>

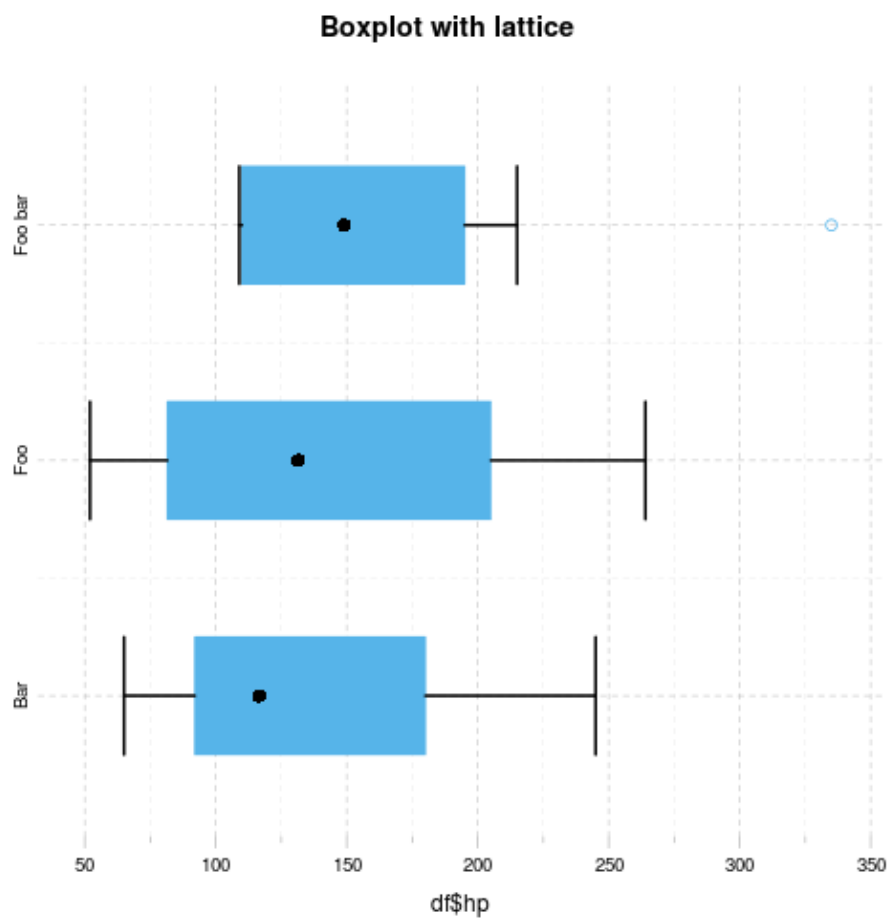


Figure 11:

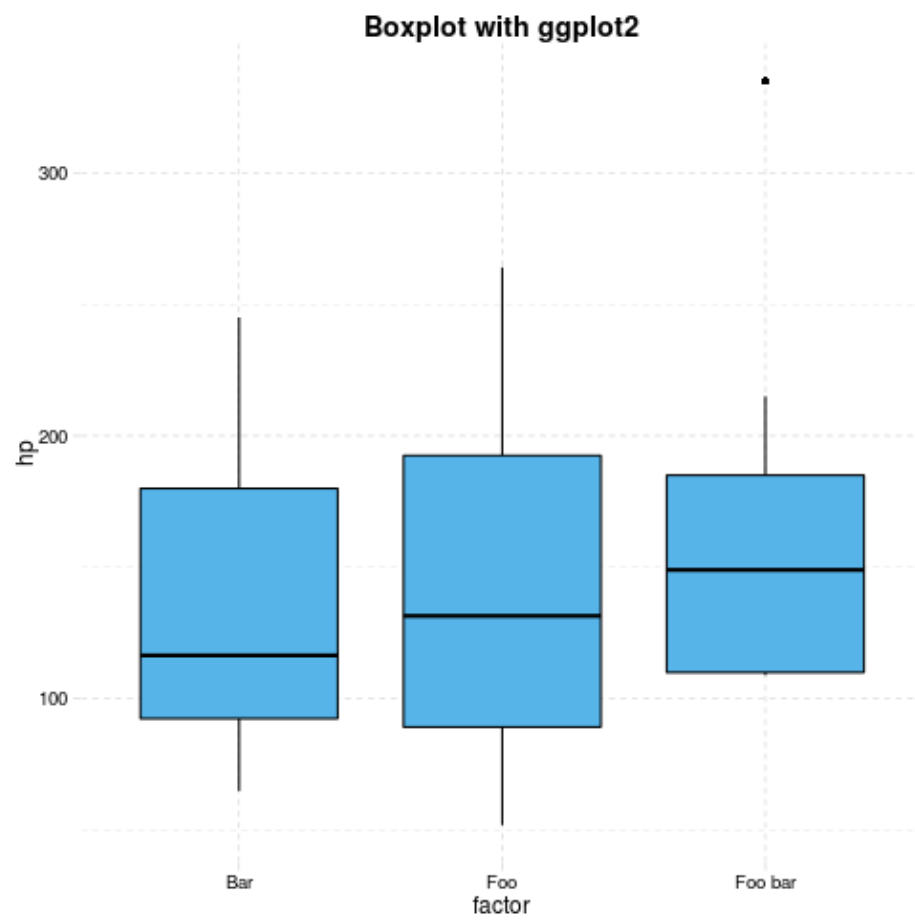


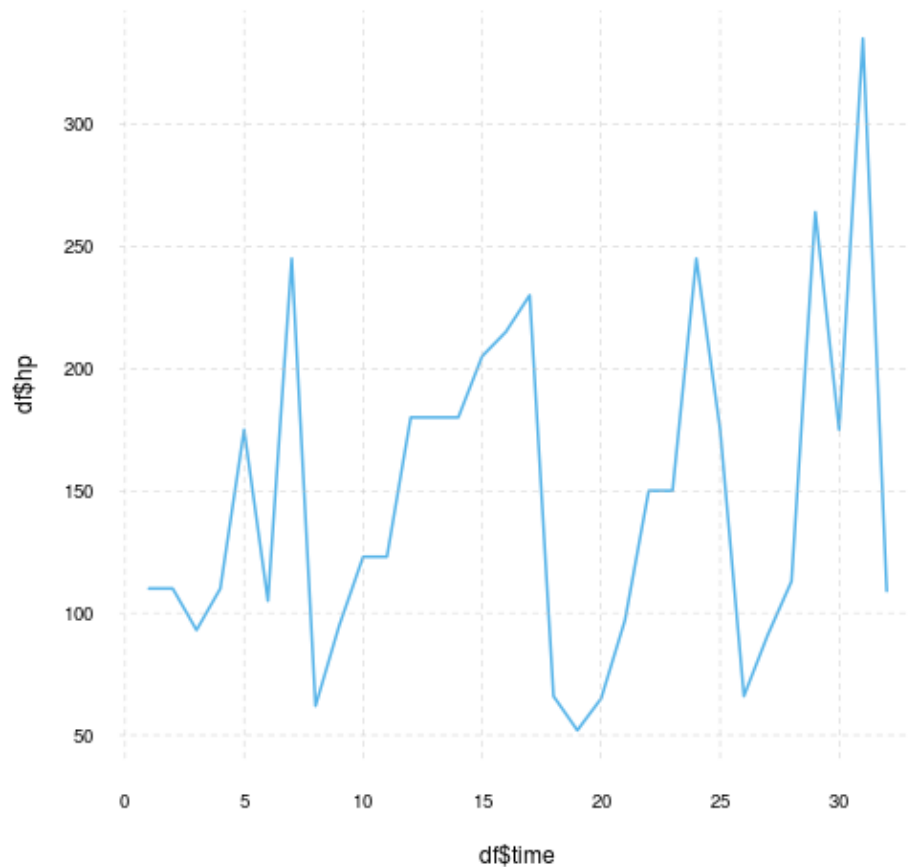
Figure 12:

**lattice**

**ggplot2**

**Lines**

**Base R plot**



**WARNING**<sup>5</sup>

<sup>4</sup>Applying default formatting to image is somehow compromised (the result could differ from what you specified in `panderOptions`). Hints: printing `lattice/ggplot2` is not needed and tweaking `base` plots with `par` might have some side-effects!

<sup>5</sup>Applying default formatting to image is somehow compromised (the result could differ from what you specified in `panderOptions`). Hints: printing `lattice/ggplot2` is not needed and tweaking `base` plots with `par` might have some side-effects!

lattice

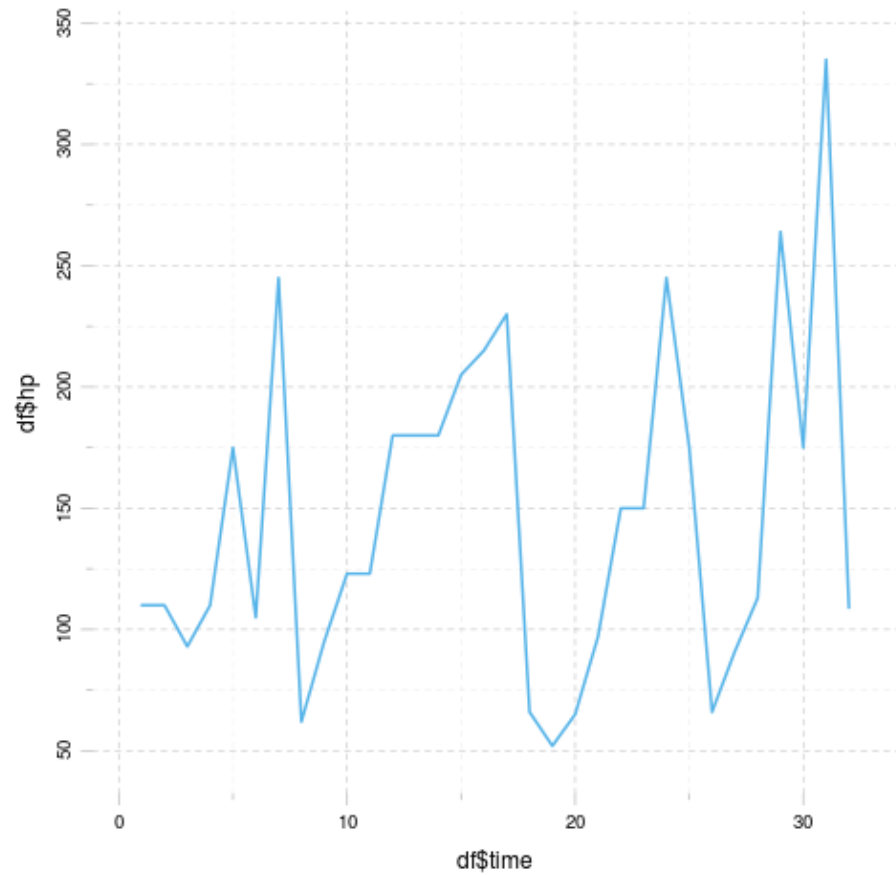


Figure 13:

ggplot2

## Custom options

Below you can find some images generated by the exact same commands but with some modified `panderOptions`:

```
panderOptions('graph.fontfamily', "Comic Sans MS")
panderOptions('graph.fontsize', 18)
```

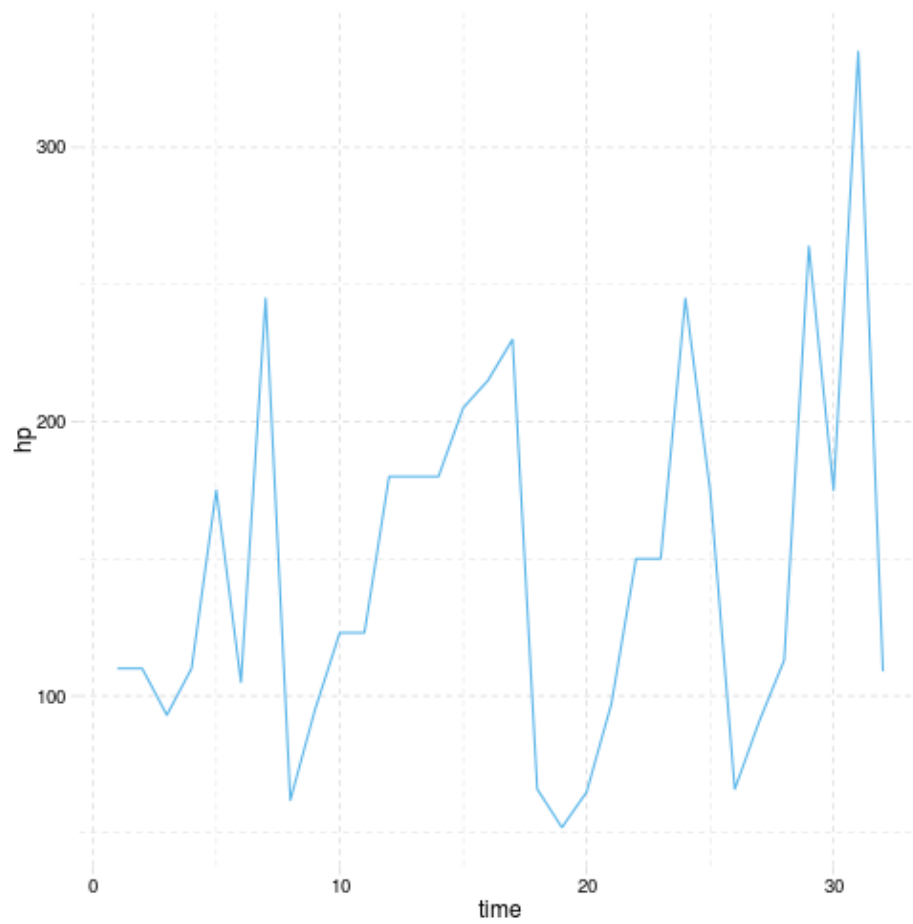


Figure 14:

```

panderOptions('graph.fontcolor', 'blue')
panderOptions('graph.grid.color', 'blue')
panderOptions('graph.axis.angle', 3)
panderOptions('graph.bboxes', T)
panderOptions('graph.legend.position', 'top')
panderOptions('graph.colors', rainbow(5))
panderOptions('graph.grid', FALSE)
panderOptions('graph.symbol', 22)

```

## Histogram

Base R plot

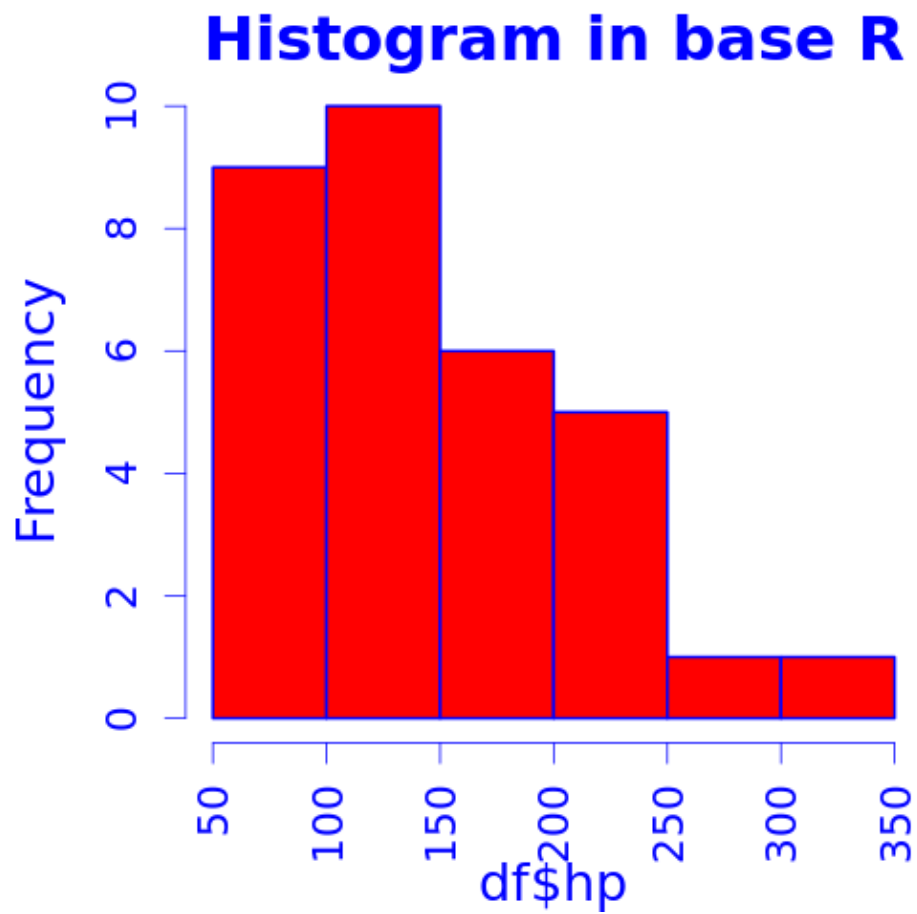


Figure 15:

`lattice`

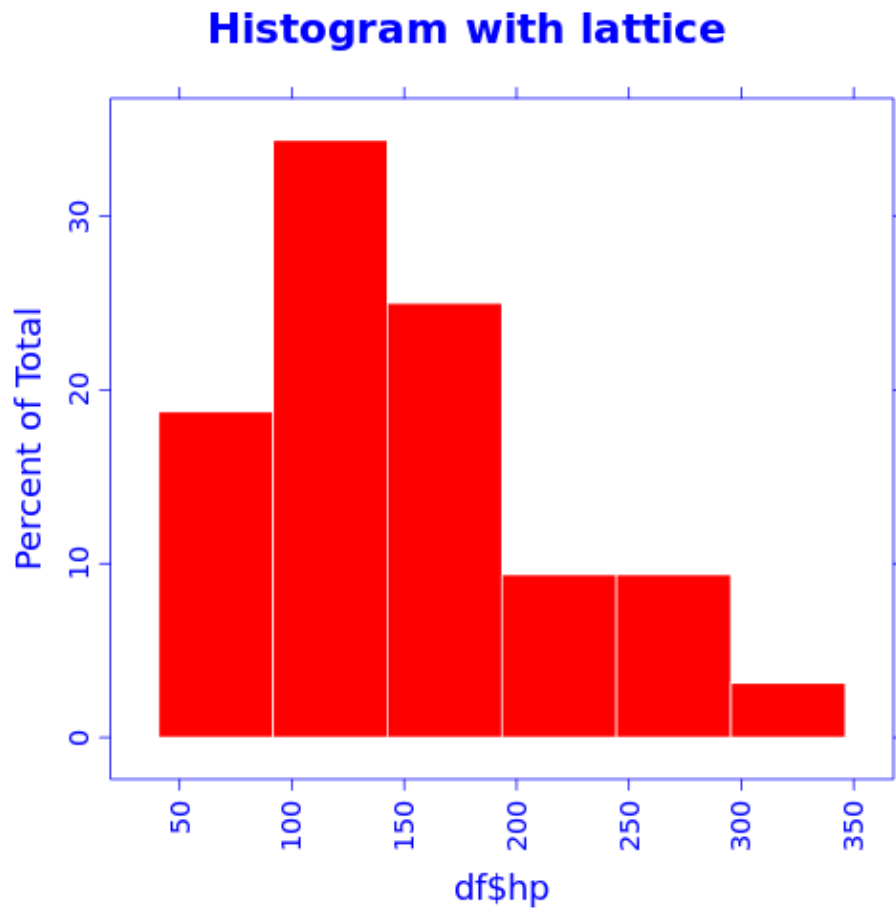


Figure 16:

`ggplot2`

**Barplot**

**Base R plot**

Yeah, the “Foo bar” label is cropped. We need a custom `mar` option here:

But wait, we lost the color! Right: unfortunately coloring base R plots is really hackish, `pander` is adding the `col` attribute to the calls. If you start to tweak `par` in a chunk, you should prepare to some unwanted side-effects. Solution:



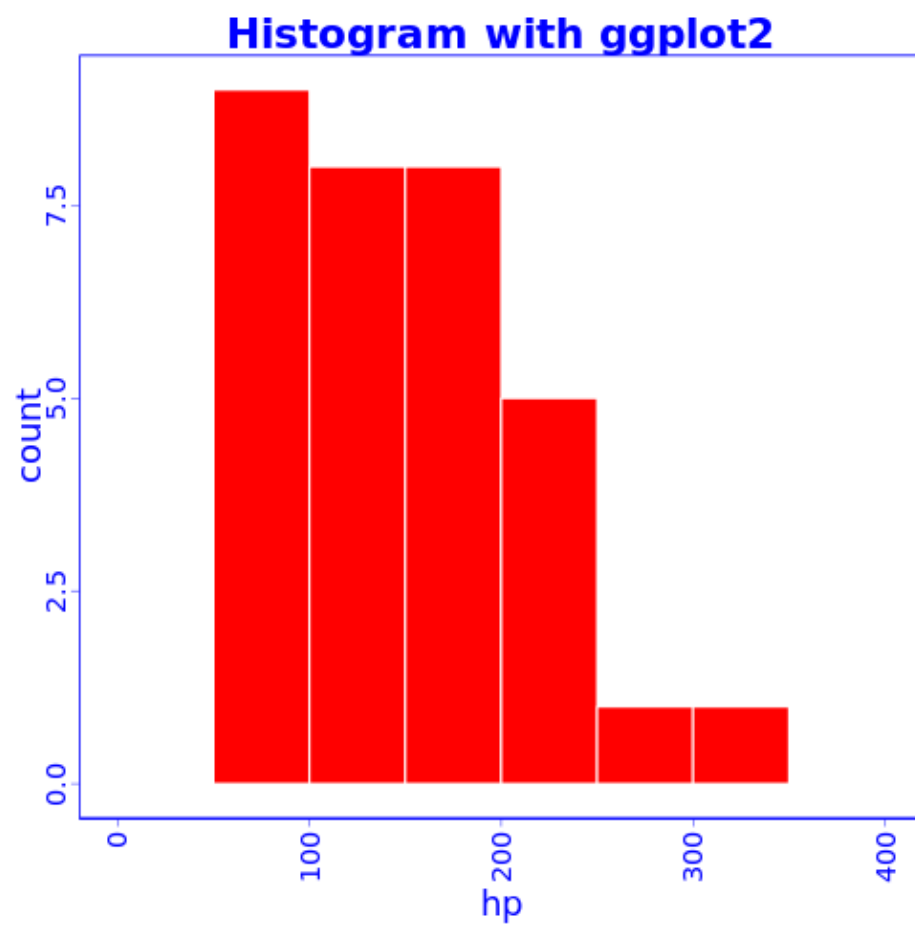


Figure 17:

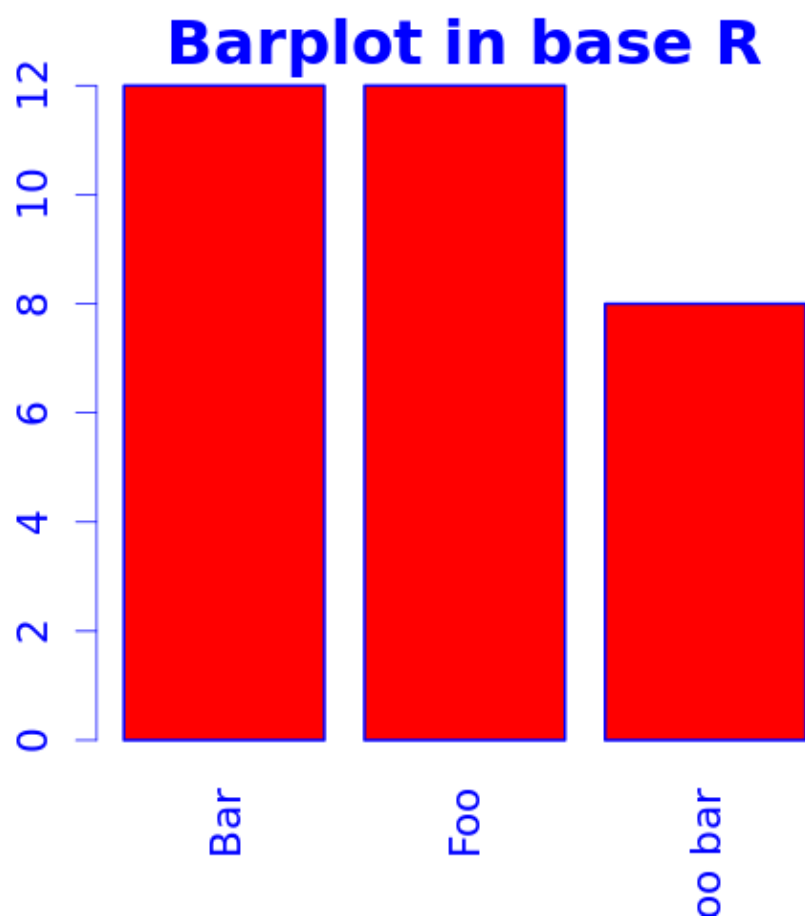


Figure 18:

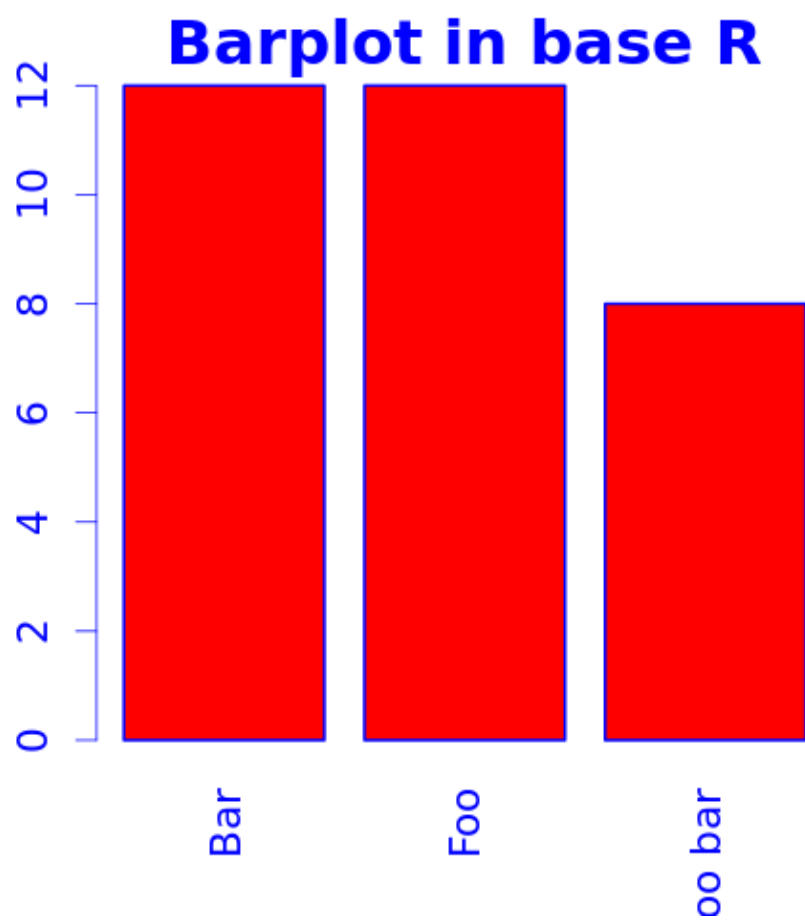


Figure 19:

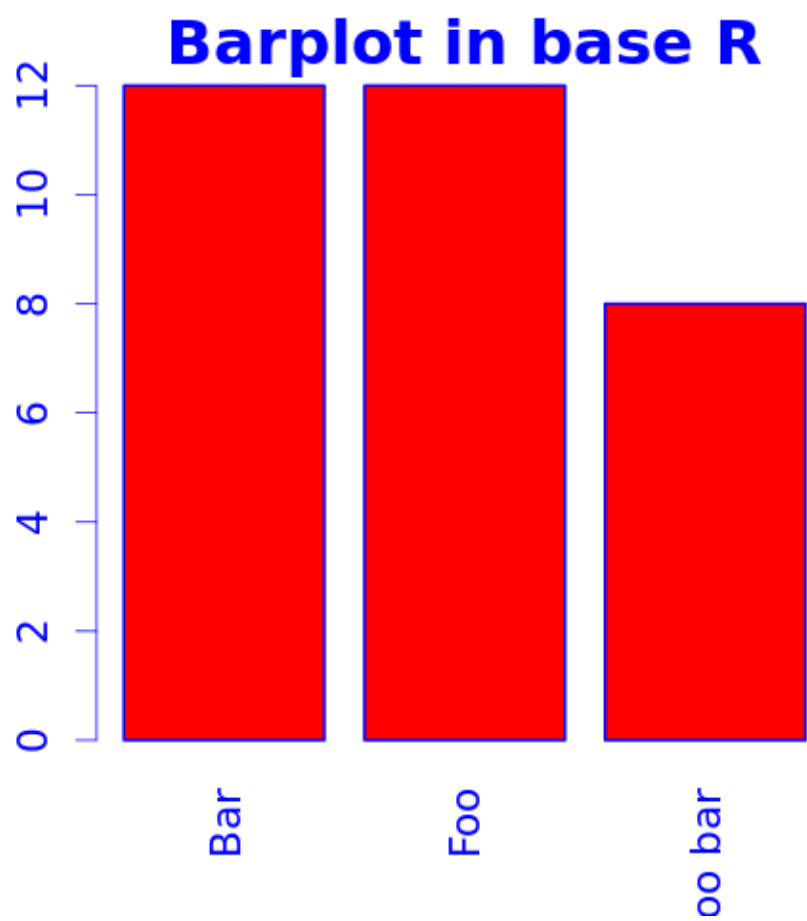


Figure 20:

lattice

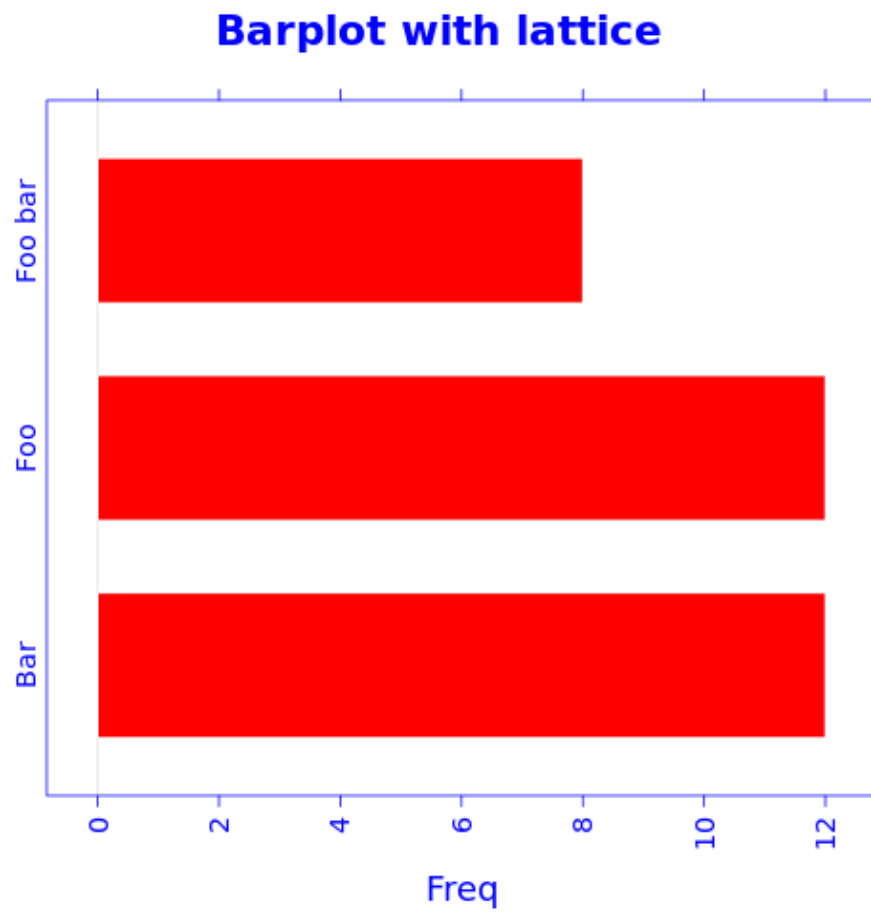


Figure 21:

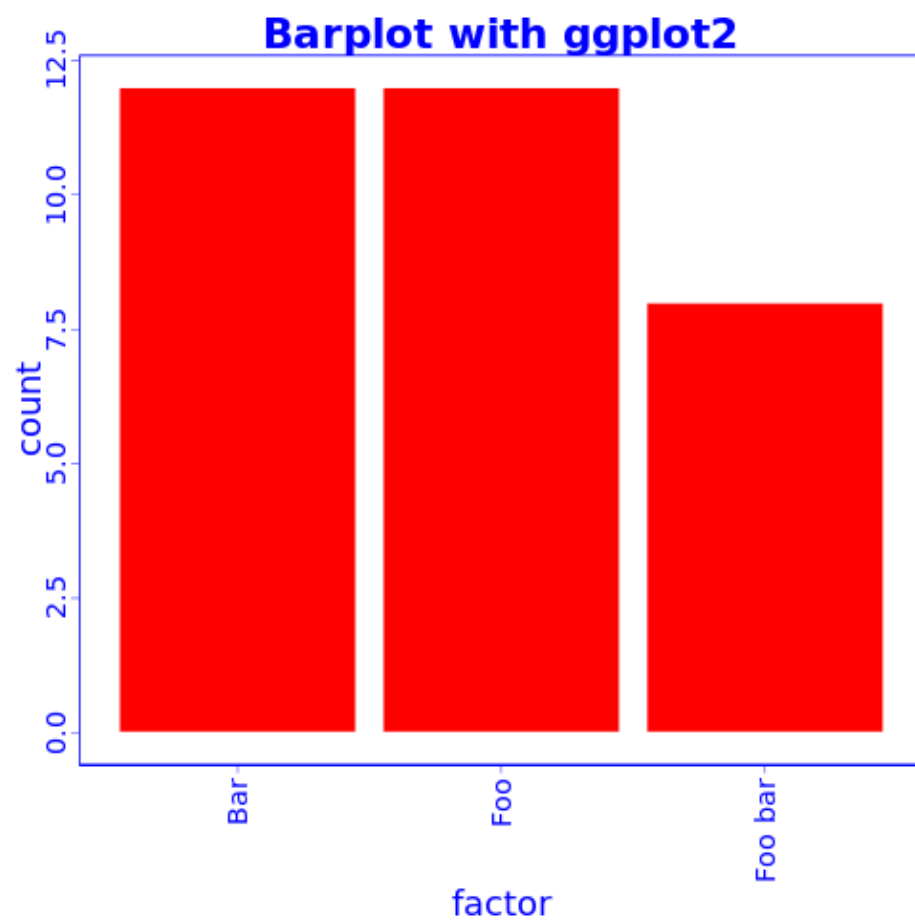


Figure 22:

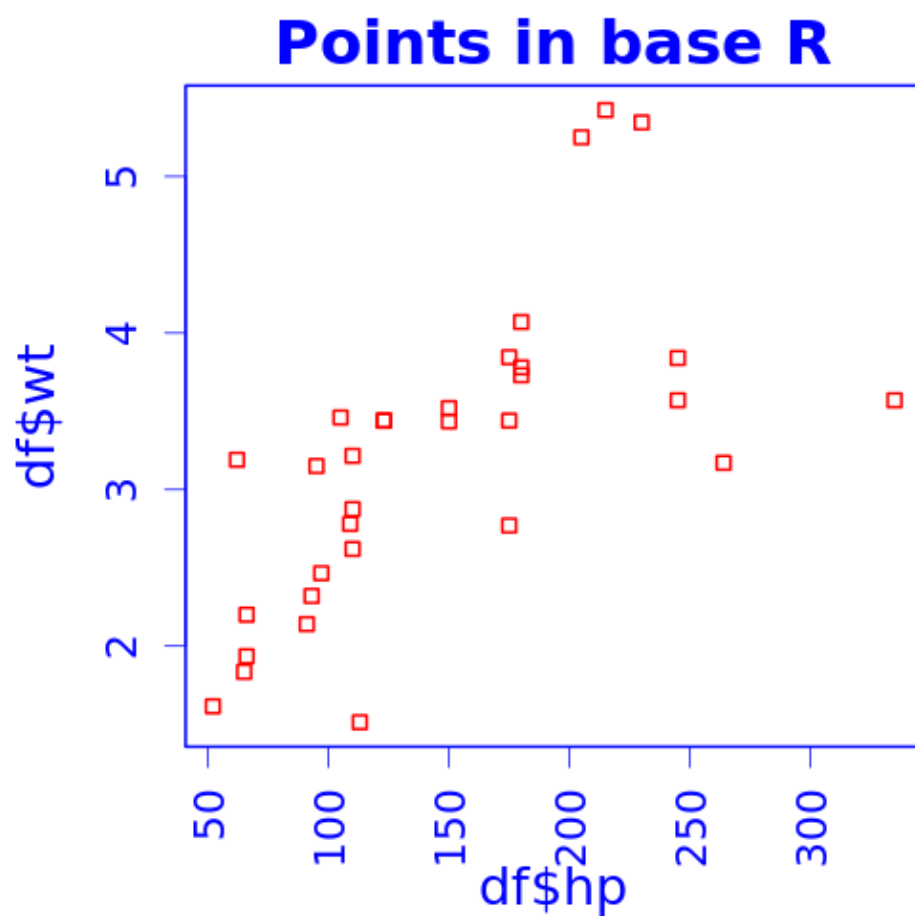


Figure 23:

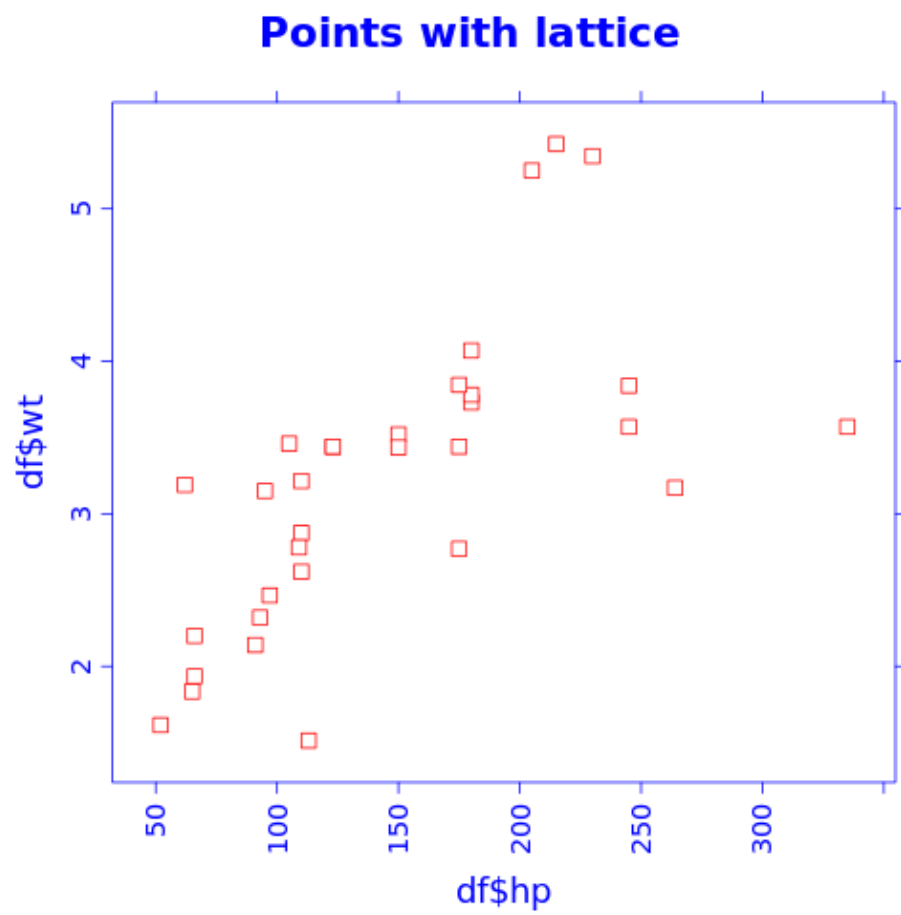


Figure 24:



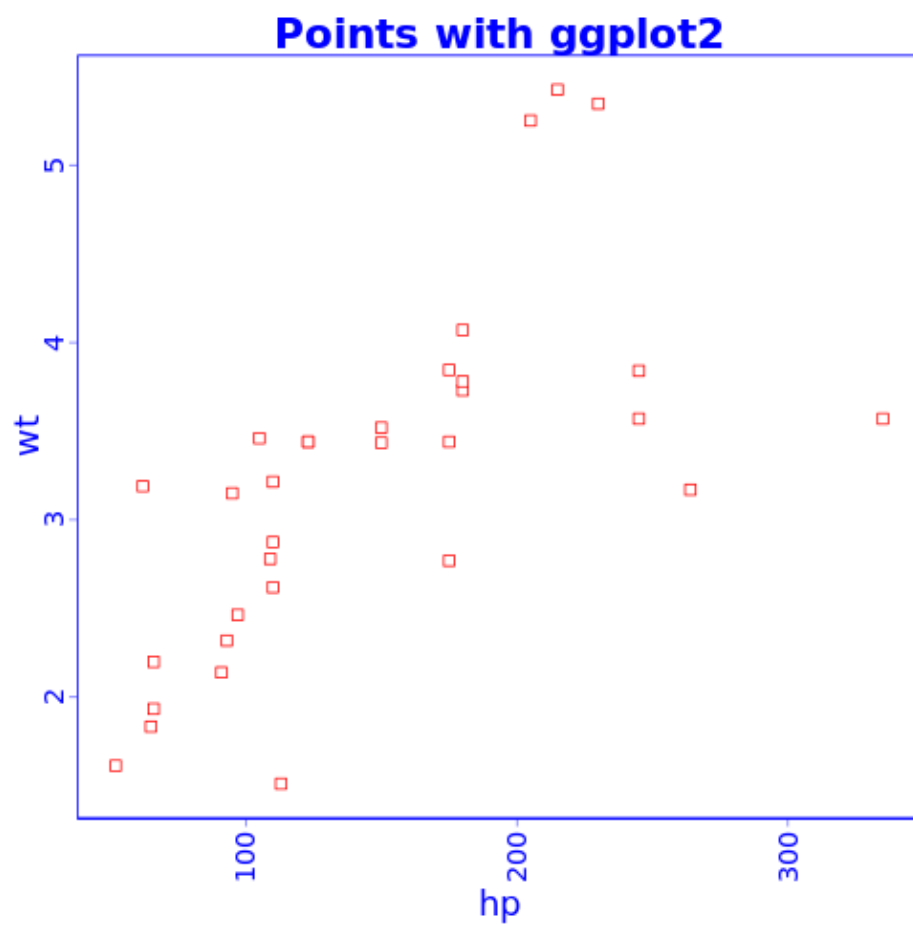


Figure 25:

**ggplot2**

**Points**

**Base R plot**

**lattice**

**ggplot2**

**Grouped plot**

**Base R plot**

I have no idea how to do that besides manually adding `points`.

**lattice**

**ggplot2**

**Facets**

**Base R plot**

I have no idea how to do that besides `par(mfrow=c(foo, bar))`.

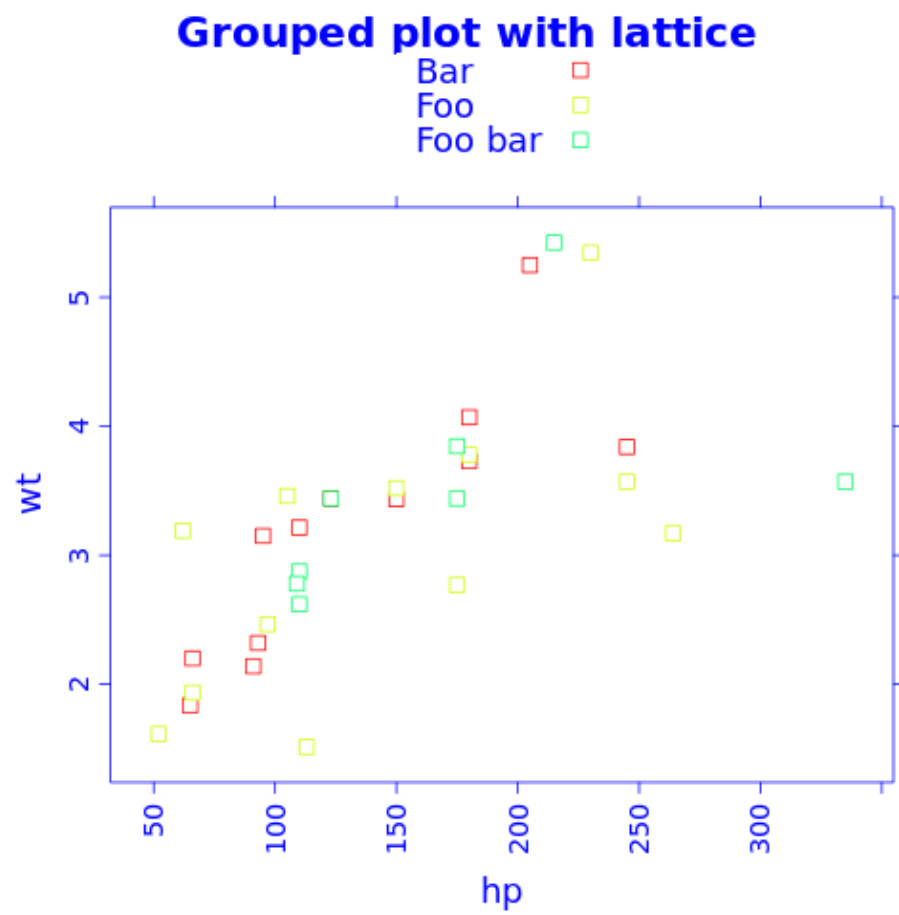


Figure 26:

## Grouped bar with ggplot2

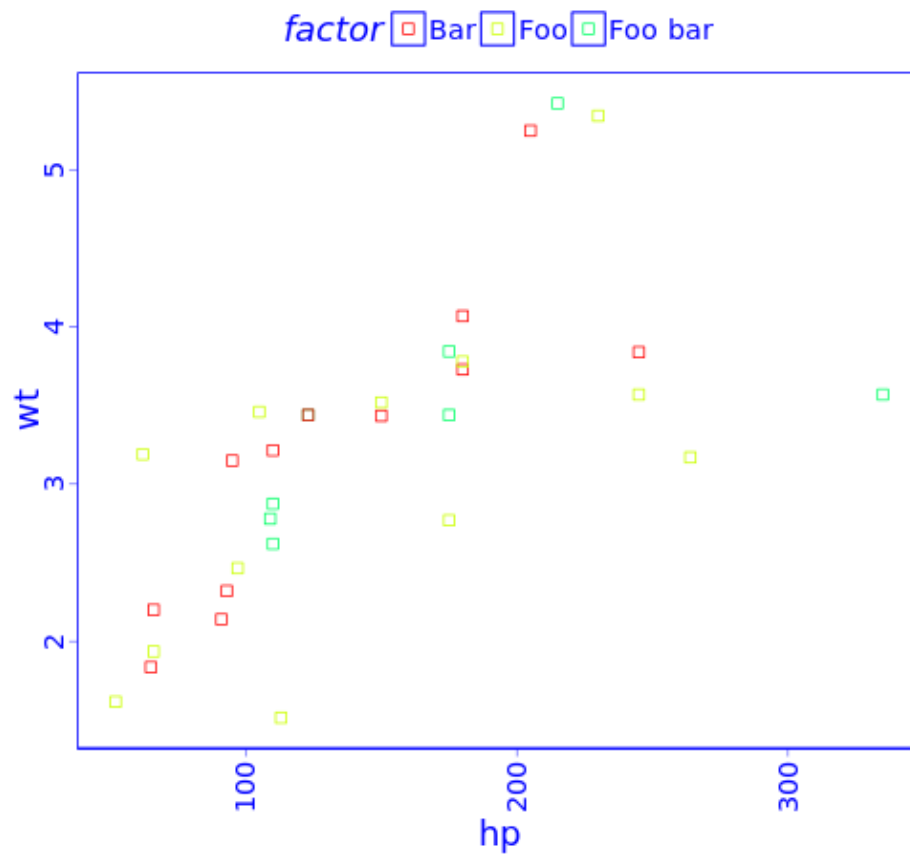


Figure 27:

## Faceted barplot with lattice

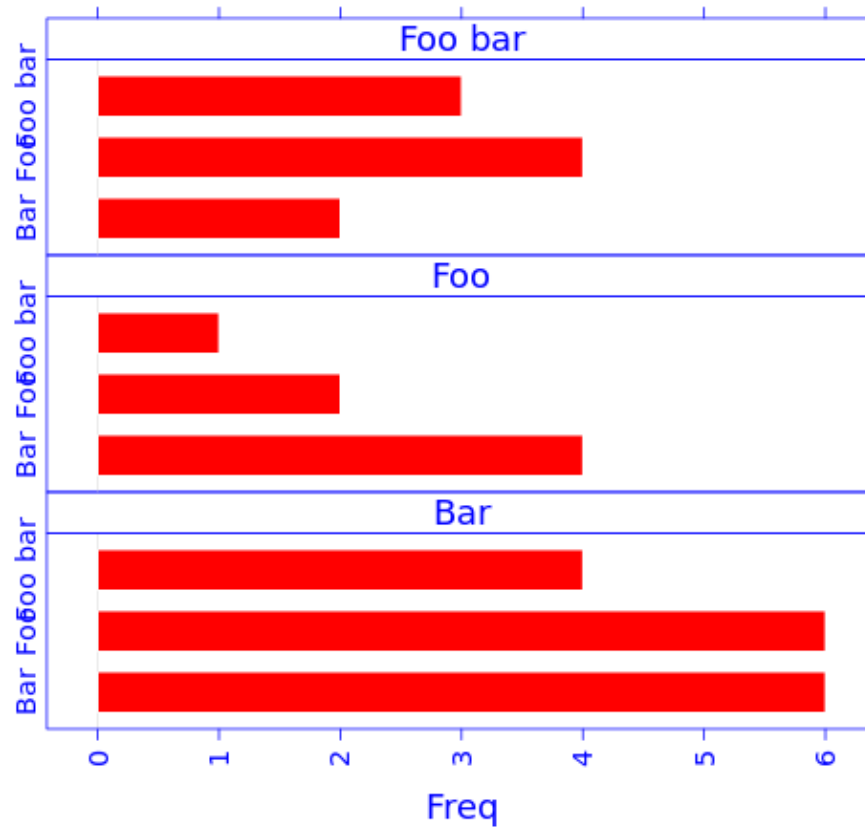


Figure 28:

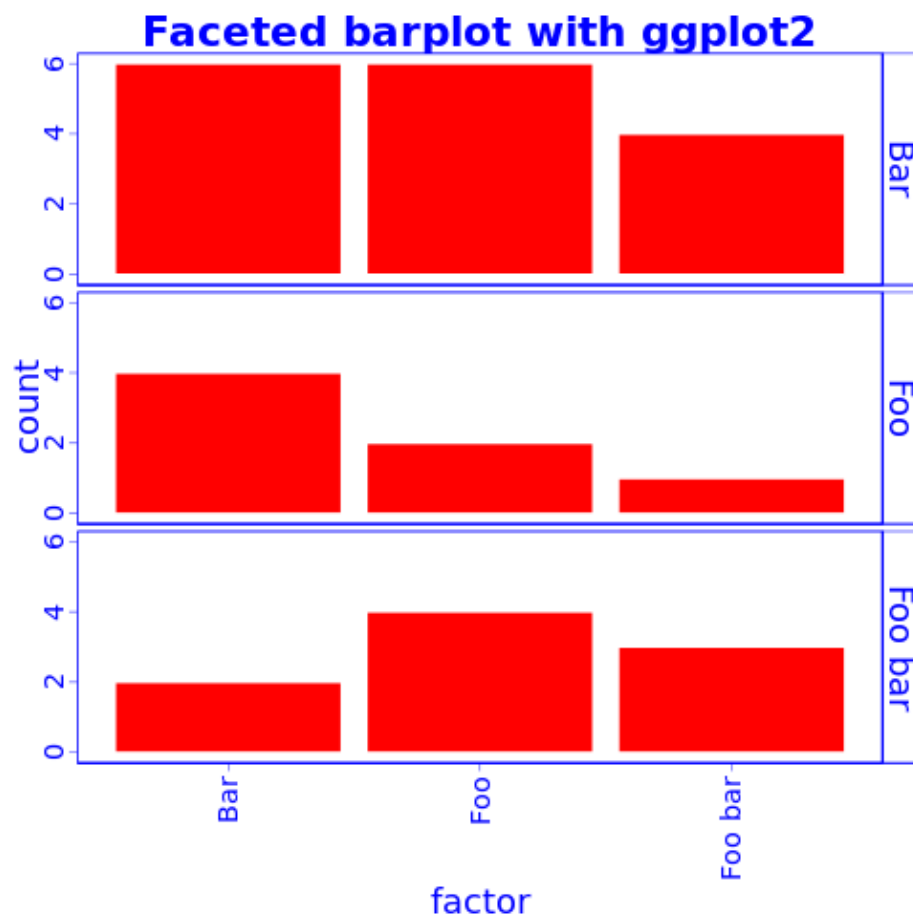


Figure 29:

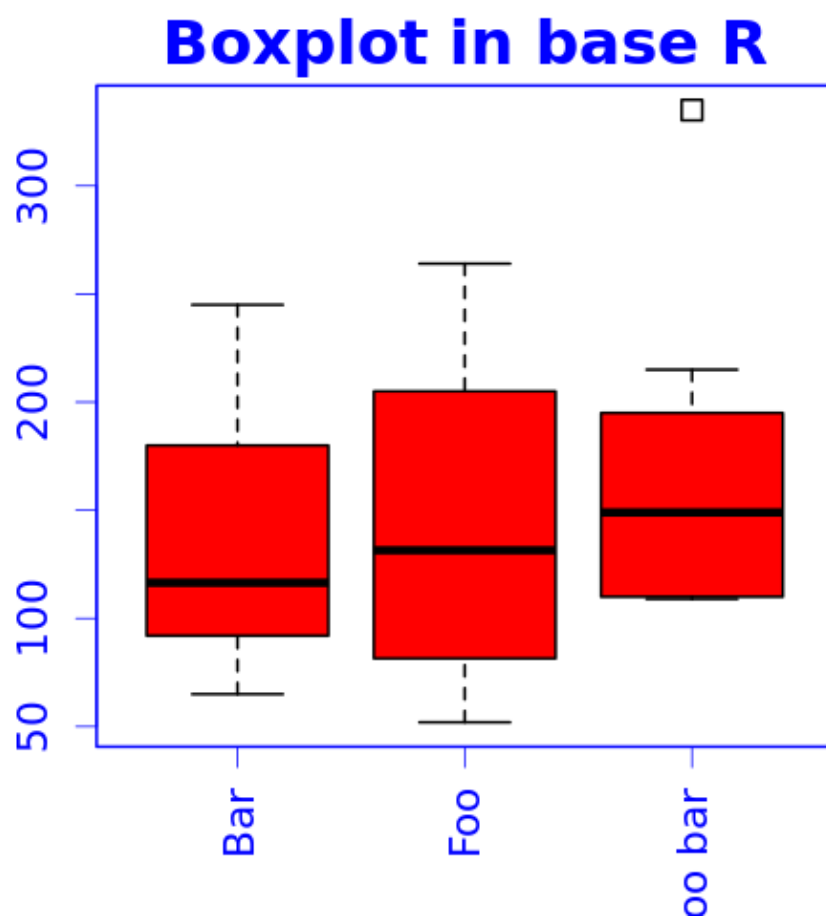


Figure 30:

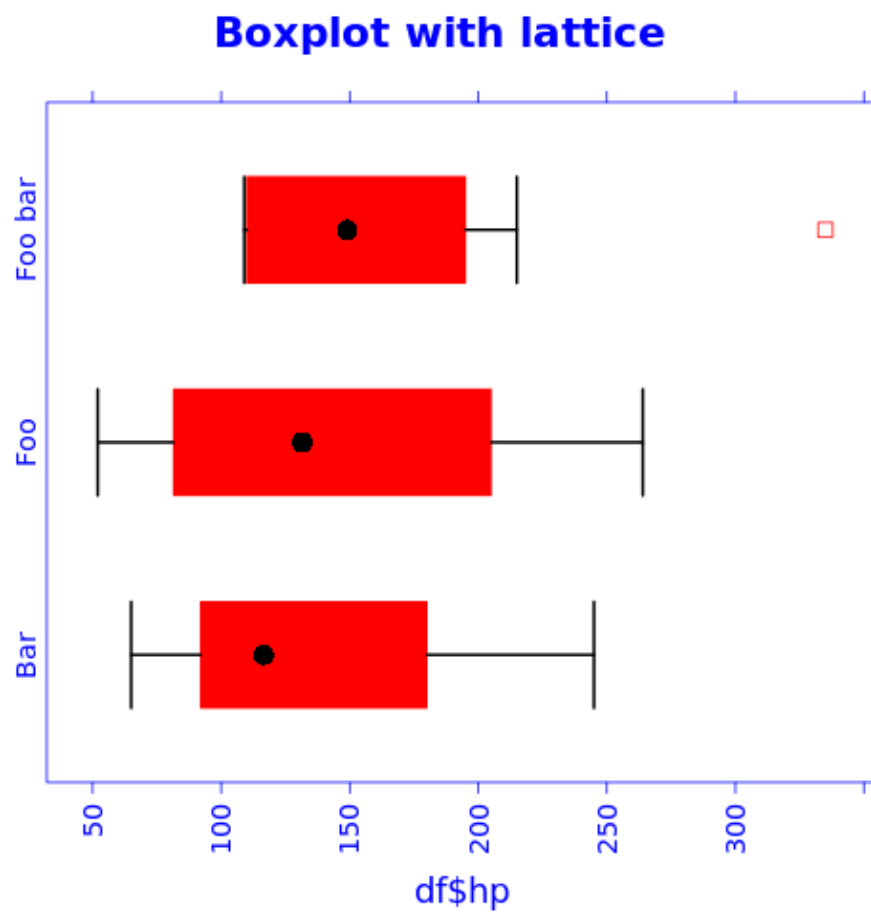


Figure 31:



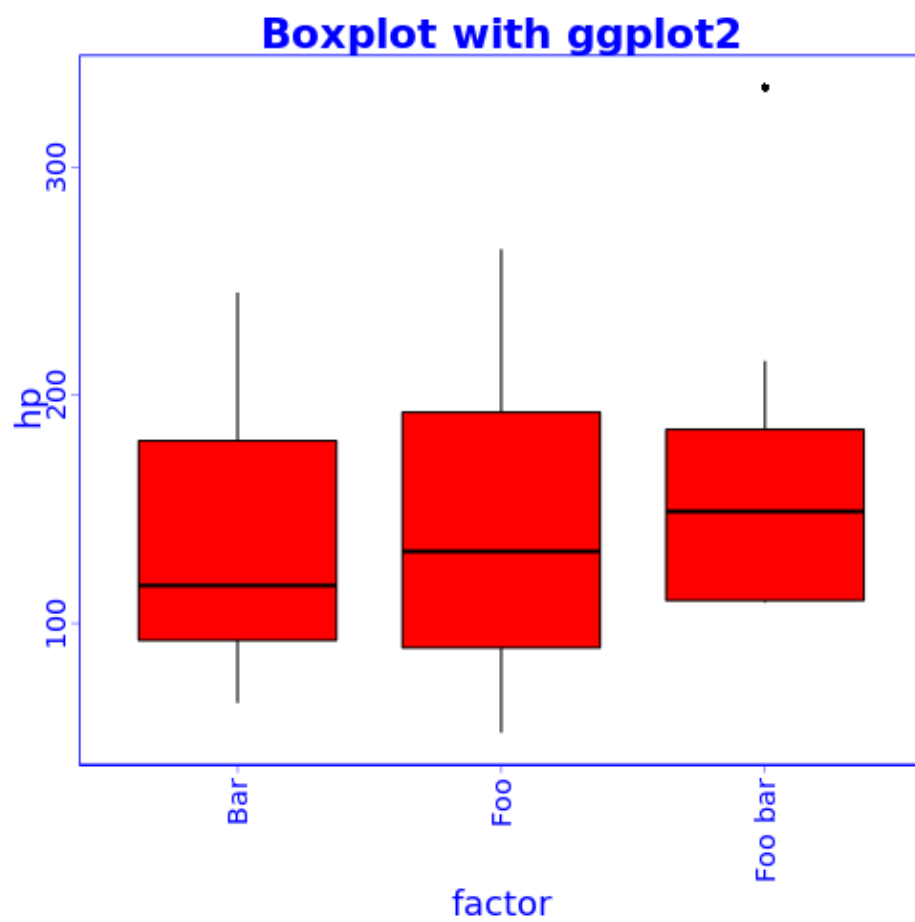


Figure 32:

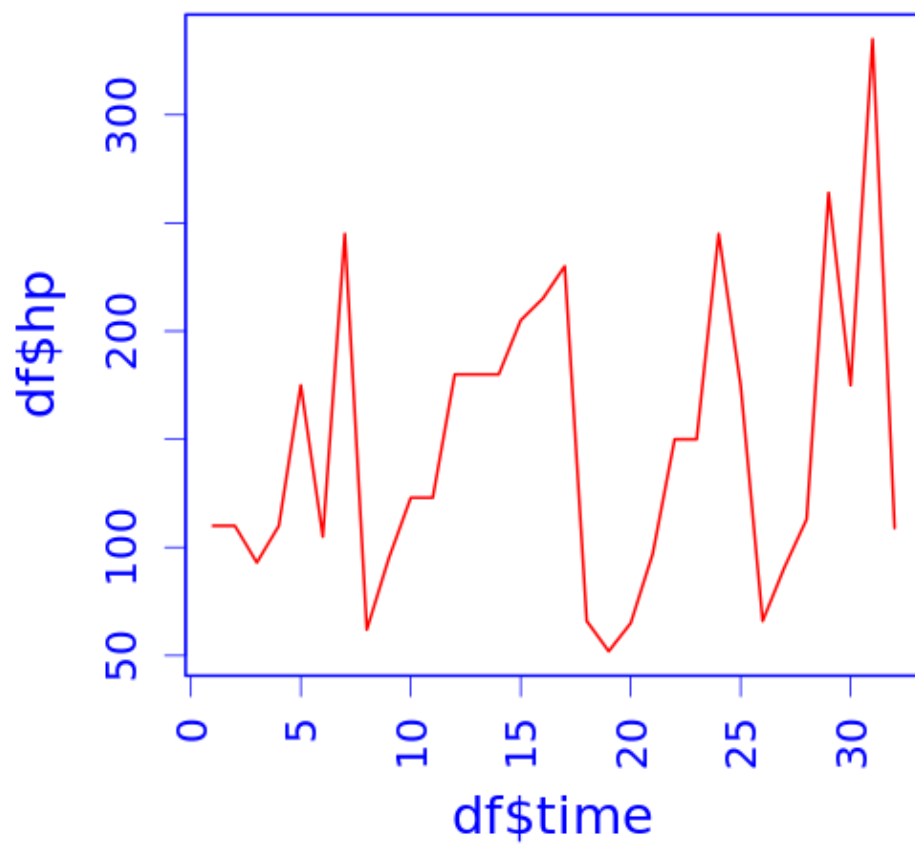


Figure 33:

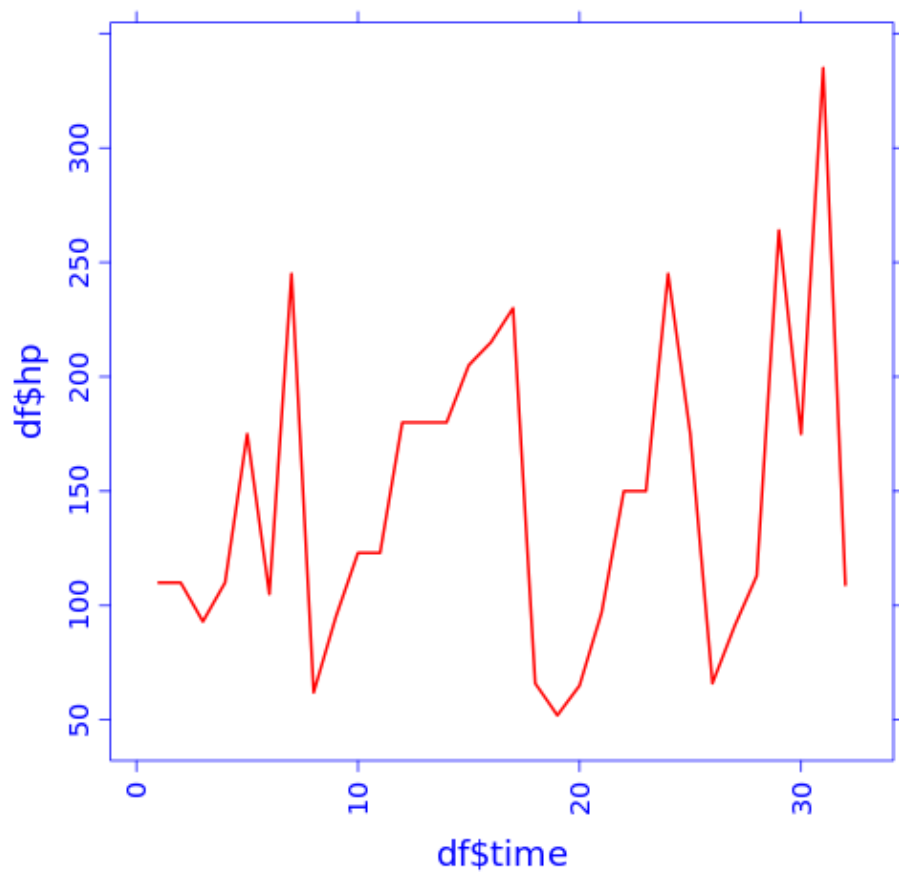


Figure 34:

**lattice**

**ggplot2**

**Boxplot**

**Base R plot**

**lattice**

**ggplot2**

**Lines**

**Base R plot**

**lattice**

**ggplot2**

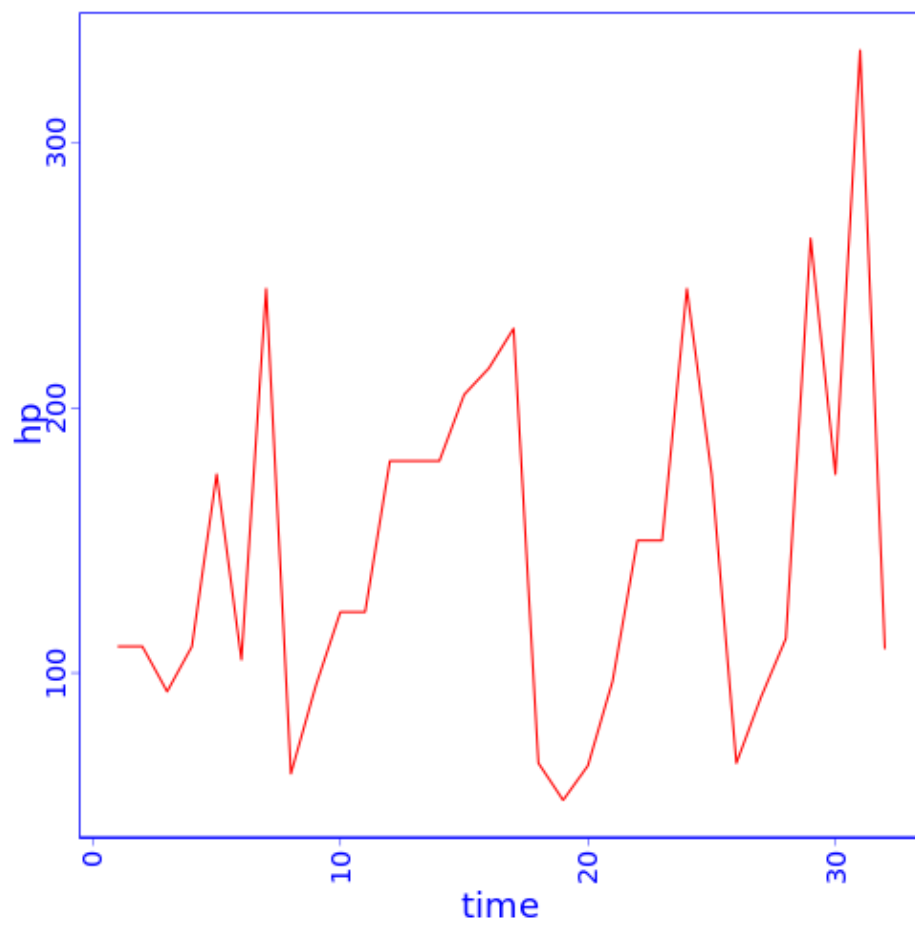


Figure 35: