ggplot2
Elegant graphics for Data Analysis

Johannes Bauer
Seminar (mit Bachelorarbeit)

20.1.2011

Outline

1. Introduction
2. The grammar of graphics
   - The grammar of graphics
   - Aesthetic attributes
   - Geometric objects
   - Faceting
3. Layers
4. Demo

Introduction

- ggplot2 is a plotting system for R

Introduction

- ggplot2 is a plotting system for R
- developed by Hadley Wickham

Introduction

- ggplot2 is a plotting system for R
- developed by Hadley Wickham
- based on the grammar of graphics (Wilkenson, 2005)

Introduction

- ggplot2 is a plotting system for R
- developed by Hadley Wickham
- based on the grammar of graphics (Wilkenson, 2005)

Installation

```
install.packages(”ggplot2”)
```

The grammar of graphics

The grammar tells us that a statistical graphic is a mapping from data to **aesthetic attributes** of **geometric objects**.

The grammar of graphics

aesthetic attributes "aesthetics"

- colour
- shape
- size

geometric objects "geoms"

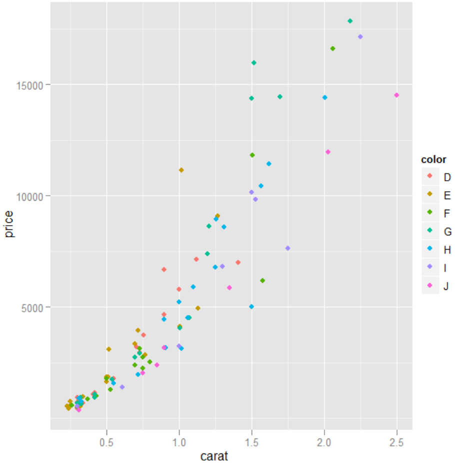
- points
- lines
- bars

aesthetic attributes

example: **qplot**(carat, price, data=dsmall, colour=color)

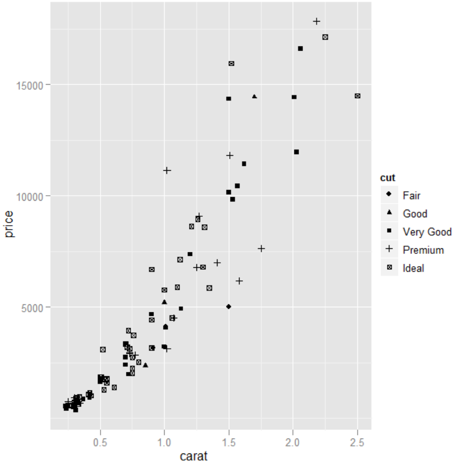
aesthetic attributes

example: **qplot**(carat, price, data=dsmall, colour=color)



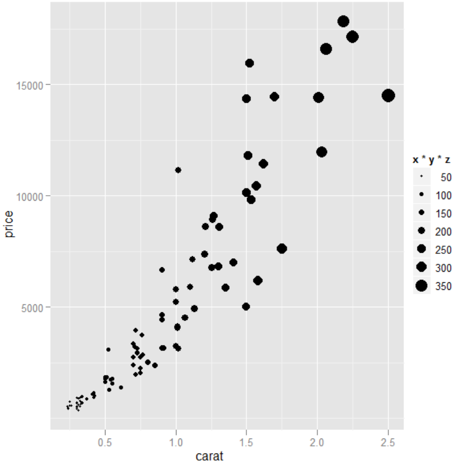
aesthetic attributes

example: qplot(carat, price, data=dsmall, shape=cut)



aesthetic attributes

example: qplot(carat, price, data=dsmall, size=x*y*z)

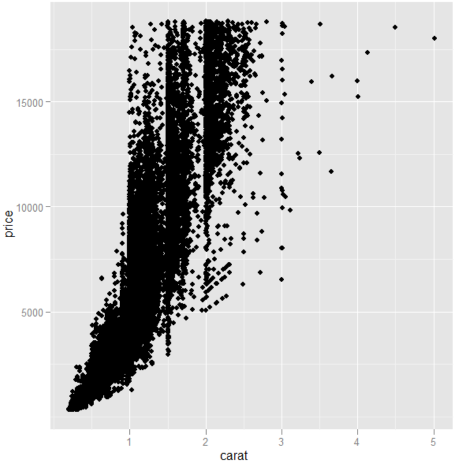


other aesthetic attributes

example: qplot(carat, price, data=diamonds)

other aesthetic attributes

example: qplot(carat, price, data=diamonds)

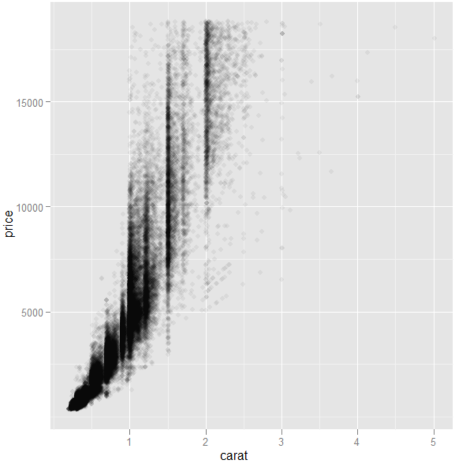


other aesthetic attributes

example: qplot(carat, price, data=diamonds, alpha=I(1/20))

other aesthetic attributes

example: qplot(carat, price, data=diamonds, alpha=I(1/20))



geometric objects

Geometric objects describe the type of object that is used to display the data.

some geoms

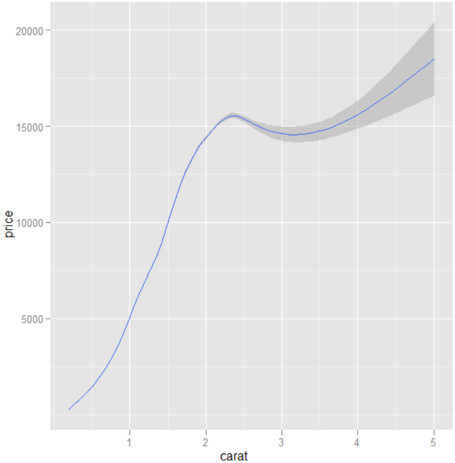
- points (default)
- smooth
- bars
- histogram
- density
- ...

Adding a smoother to a scatterplot

example: qplot(carat, price, data=diamonds, geom=”smooth”)

Adding a smoother to a scatterplot

example: qplot(carat, price, data=diamonds, geom="smooth")



Adding a smoother to a scatterplot

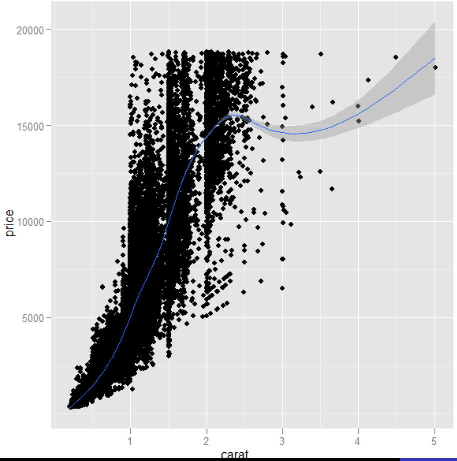
example:

qplot(carat, price, data=diamonds, geom=c("points", "smooth"))

Adding a smoother to a scatterplot

example:

qplot(carat, price, data=diamonds, geom=c(”points”, ”smooth”))



Adding a smoother to a scatterplot

Methods

Adding a smoother to a scatterplot

Methods

- method = "loess" (default)
  uses a smooth local regression

Adding a smoother to a scatterplot

Methods

- method = "loess" (default)
  uses a smooth local regression
- method = "lm"
  fits a linear model

Adding a smoother to a scatterplot

Methods

- method = "loess" (default)
  uses a smooth local regression
- method = "lm"
  fits a linear model
- method = "rlm"
  works like lm but uses a robust fitting algorithm so that outliers don't affect the fit as much.

Adding a smoother to a scatterplot

Methods

- method = "loess" (default)
  uses a smooth local regression
- method = "lm"
  fits a linear model
- method = "rlm"
  works like lm but uses a robust fitting algorithm so that outliers don't affect the fit as much.
- ...

Faceting

- With aesthetics we can compare different subgroups of our dataframe on **the same plot**.

Faceting

- With aesthetics we can compare different subgroups of our dataframe on **the same plot**.
- Faceting also splits the data in subgroups but displays it on **multiple plots**.

Faceting

```
qplot(carat, data=diamonds, facets = color ∼ . ,
geom = ”histogram”, binwidth = 0.1, xlim=c(0,3))
```

Faceting

qplot(carat, data=diamonds, facets = color ∼ . , geom = "histogram", binwidth = 0.1, xlim=c(0,3))

Methods

- facets = color ∼ .

Faceting

qplot(carat, data=diamonds, facets = color ∼ . ,
geom = "histogram", binwidth = 0.1, xlim=c(0,3))

Methods

- facets = color ∼ .
- geom = "histogram" has options like the binwidth

Faceting

qplot(carat, data=diamonds, facets = color ∼ . ,
geom = "histogram", binwidth = 0.1, xlim=c(0,3))

Methods

- facets = color ∼ .
- geom = "histogram" has options like the binwidth
- xlim=c(0,3) sets the limits of the x-axes

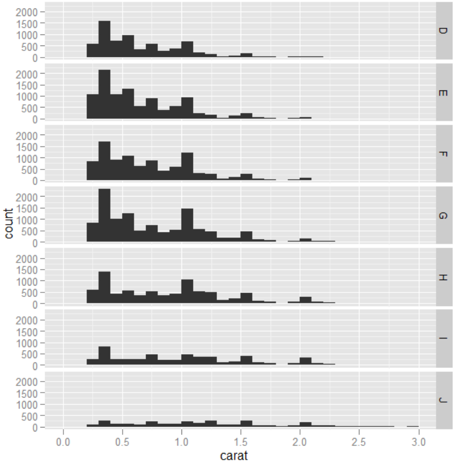
Faceting

qplot(carat, data=diamonds, facets = color ~ . , geom = "histogram", binwidth = 0.1, xlim=c(0,3))

Methods

- facets = color ~ .
- geom = "histogram" has options like the binwidth
- xlim=c(0,3) sets the limits of the x-axes
- attributes xlim, ylim, main, xlab, ylab are equivalent to basic plot

Faceting



Differences from the basic plot function

- qplot() is not generic.
- We cannot use points(), lines() or text() to add further graphic elements. We need additional layers.

creating a plot

When we used qplot(), it did a lot of things for us:

creating a plot

When we used qplot(), it did a lot of things for us:

- it created a plot object
- it added layers
- displays the result

- To create the plot ourself, we use **ggplot()**.
- ggplot() has two arguments
    - the dataframe
    - aesthetic mapping

creating a plot - example

Creates a plot but there is nothing to see.

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))
```

creating a plot - example

Creates a plot but there is nothing to see.

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))
```

Add a layer to the plot.

```
p <- p + layer(geom = "point")
```

creating a plot - example

Creates a plot but there is nothing to see.

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))
```

Add a layer to the plot.

```
p <- p + layer(geom = "point")
```

Show the plot.

```
p
```

creating a plot - complex example

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))
p <- p + layer(
geom = "bar",
geom_params = list(fill = "steelblue"),
stat = "bin",
stat_params = list(binwidth = 2)
)

p
```

creating a plot - complex example

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))
p <- p + layer(
geom = "bar",
geom_params = list(fill = "steelblue"),
stat = "bin",
stat_params = list(binwidth = 2)
)

p
```

We can simplify this layer definition by using **shortcuts**.

creating a plot - simplified complex example

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))
```

creating a plot - simplified complex example

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))

p <- p + geom_histogram(binwidth = 2, fill = "steelblue")
```

creating a plot - simplified complex example

```
p <- ggplot(diamonds, aes(carat, price, colour = cut))

p <- p + geom_histogram(binwidth = 2, fill = "steelblue")
```

All the shortcut functions have the same basic form, beginning with geom_ or stat_:

- geom_XXX(...)
- stat_XXX(...)

- geom_bar: Bars, rectangles with bases on y-axis
- geom_boxplot: Box and whiskers plot
- geom_line: Connect observations, in ordered by x value

more examples in R

References:

- ggplot2 - Elegant Graphics for Data Analysis
  Hadley Wickham, Springer 2009
- Online Reference
  http://had.co.nz/ggplot2

Thank you.
Questions?