

定位

# 本节内容

- 分面
  - 分面即在一个页面上自动摆放多幅图形：先将数据划分为多个子集，然后将每个子集依次绘制到页面的不同版面中。这类图形也称“小联号图”（**small multiples**）。
- 坐标系
  - 坐标系即通过两个独立的位置标度来生成一个二维的坐标系。最常见的是笛卡尔坐标系。

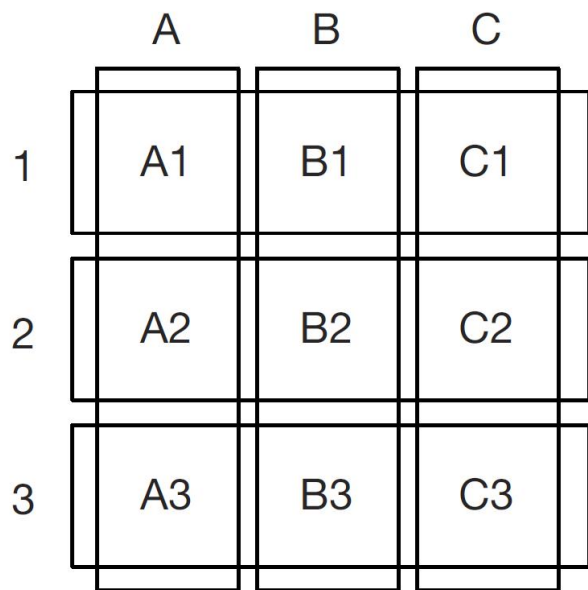
分面

# 分面

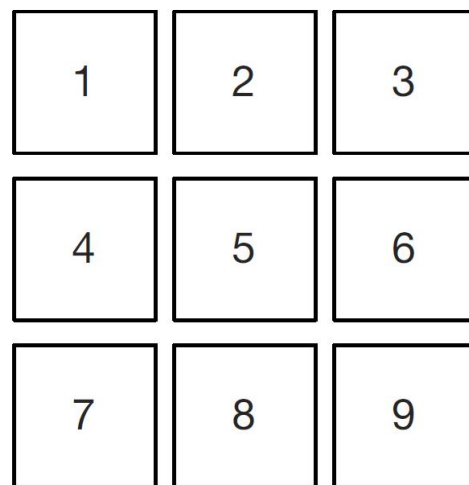
- 对于探索性分析来说，分面是一个强大的工具。它能帮我们快速地分析出数据各自模式的异同。

# 网格型VS封装型

- **ggplot2**提供两种分面类型：
  - 网格型（**facet\_grid**）：左图。本质上为二维，由两个独立的部分组成。
  - 封装型（**facet\_wrap**）：右图。本质上为一维，为节省空间而封装成二维。



**facet\_grid**



**facet\_wrap**

# 数据集

- 使用mpg的子集:

```
mpg2 <- subset(mpg, cyl != 5 & drv %in% c("4",  
"f"))
```

# 网格分面

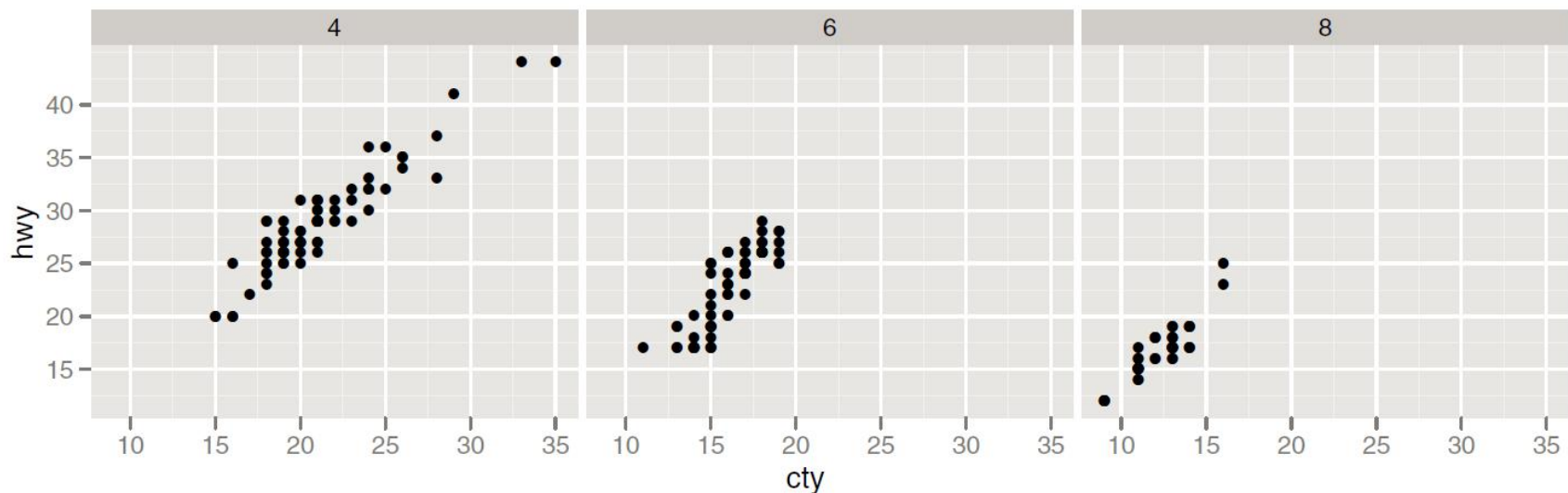
# 网格分面

- 网格分面在二维网格中展示图形。输入分面表达式时，需要设定哪些变量作为分面绘图的行，哪些变量作为列。
- 分面规则见随后几页。



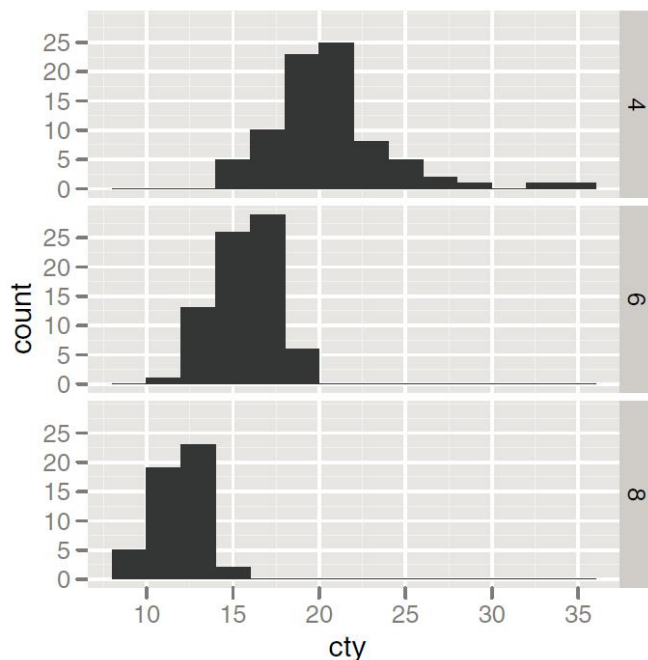
# 一行多列

- 一行多列“`. ~ a`”：电脑屏幕通常是宽的，因此这个方向适合展示数据。由于纵坐标相同，也利于y位置的比较。
  - `qplot(cty, hwy, data = mpg2) + facet_grid(. ~ cyl)`
  - 当然也可以用`ggplot()`: `ggplot(data=mpg2, aes(cty,hwy))+geom_point()+facet_grid(.~cyl)`



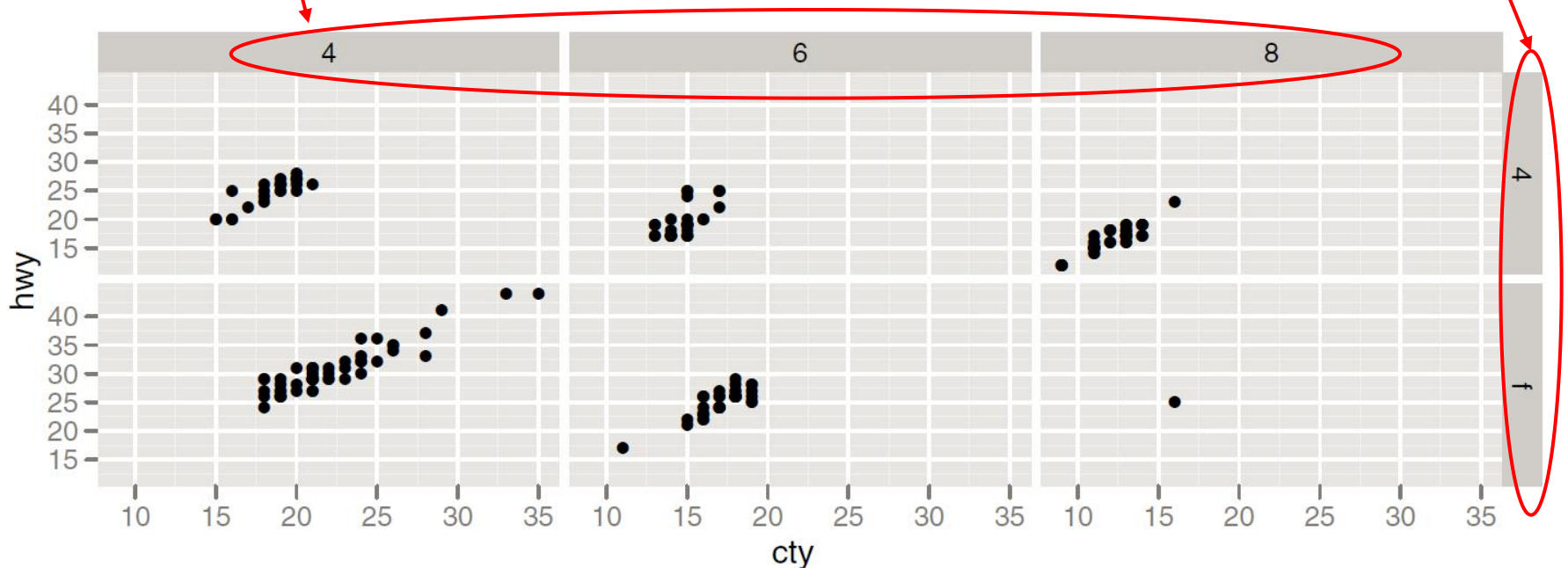
# 一行多列

- 一行多列“`b ~ .`”：横坐标相同，利于x位置的比较，尤其是对数据分布的比较。
  - `qplot(cty, data = mpg2, geom="histogram", binwidth = 2) + facet_grid(cyl ~ .)`



# 多行多列

- 多行多列“a ~ b”：通常将因子水平数目最大的变量按列排放，以利用屏幕的宽高比。
  - `qplot(cty, hwy, data = mpg2) + facet_grid(drv ~ cyl)`



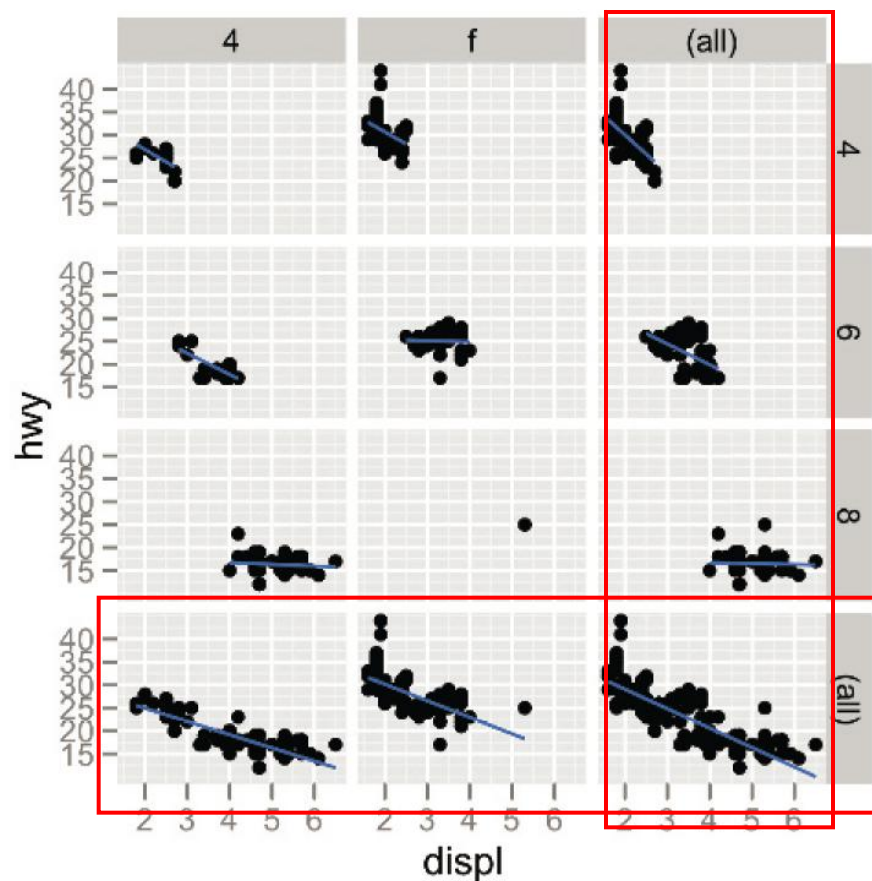
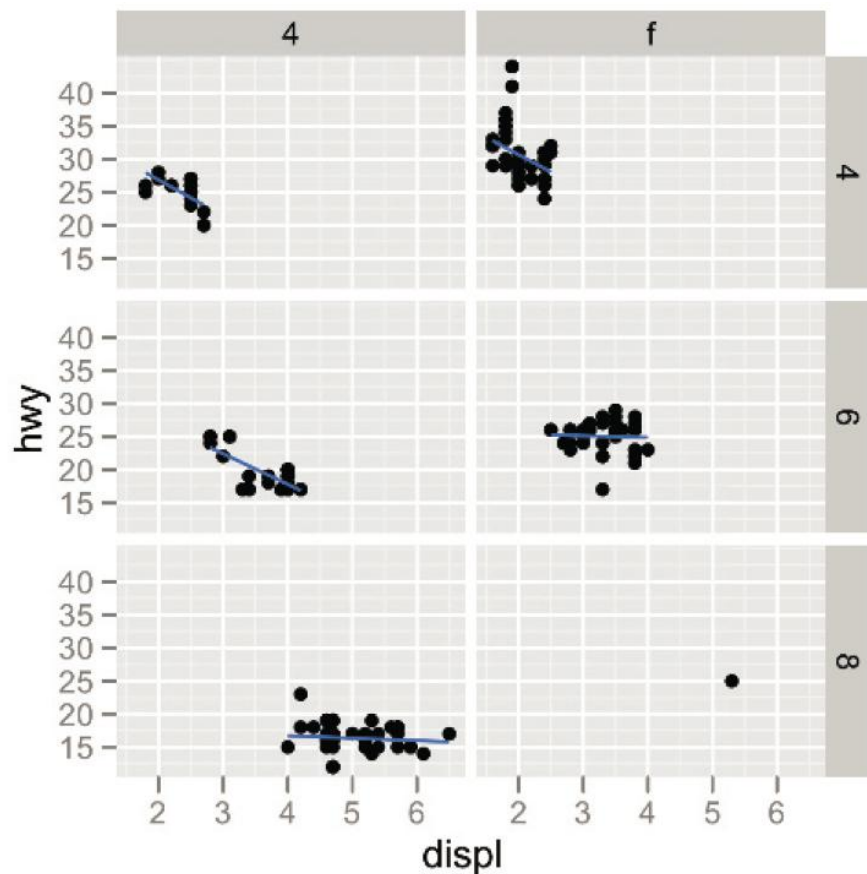
# 边际图 (Margins)

- 边际图就相当于一个列联表。
- 设定 `margins = TRUE` 可展示所有的边际图，或通过如 `margins = c("sex", "age")` 列出所要展示的边际图的变量名称。
- 例子见下页。

```
p <- qplot(displ, hwy, data = mpg2) +  
geom_smooth(method = "lm", se = F)
```

```
p + facet_grid(cyl ~ drv) #左图： 没有边际图
```

```
p + facet_grid(cyl ~ drv, margins = T) #右图： 展示了所有的边际图
```



封装分面

# 封装分面

- **facet\_wrap**并不是生成一个二维网格，而是先生成一个长的面板条，然后将它封装在二维中。
- 在处理单个多水平变量时，这种处理方式非常有用。可以有效地利用空间来安放图形。
- 例子见下页。

```
library(ggplot2movies)
```

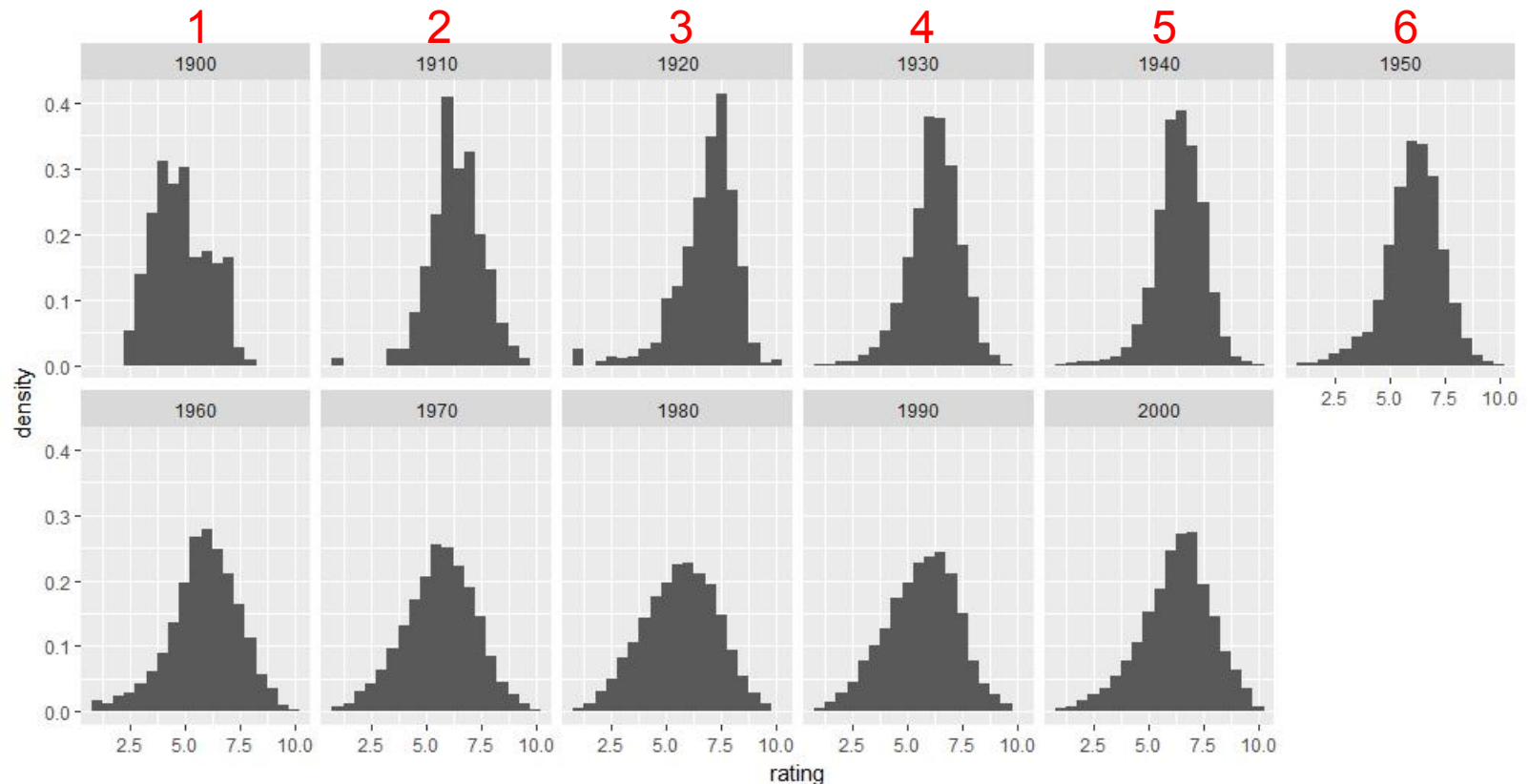
```
library(plyr)
```

```
movies$decade <- round_any(movies$year, 10, floor)
```

```
qplot(rating, ..density.., data=subset(movies, decade > 1890),
```

```
  geom="histogram", binwidth = 0.5) +
```

```
  facet_wrap(~ decade, ncol = 6)
```





# 标度控制

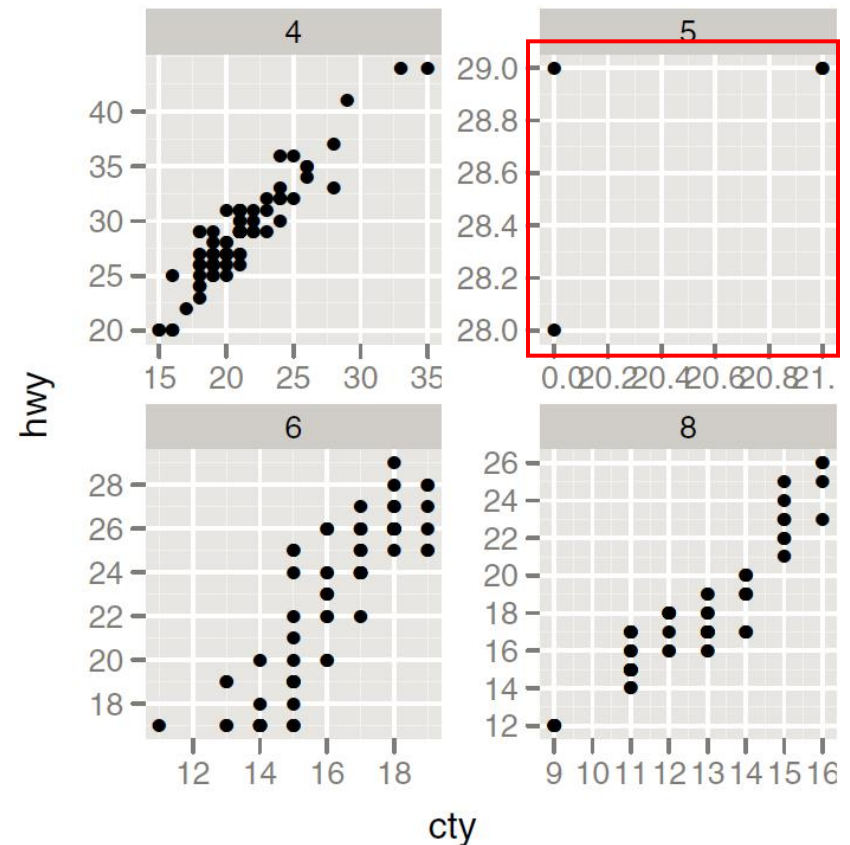
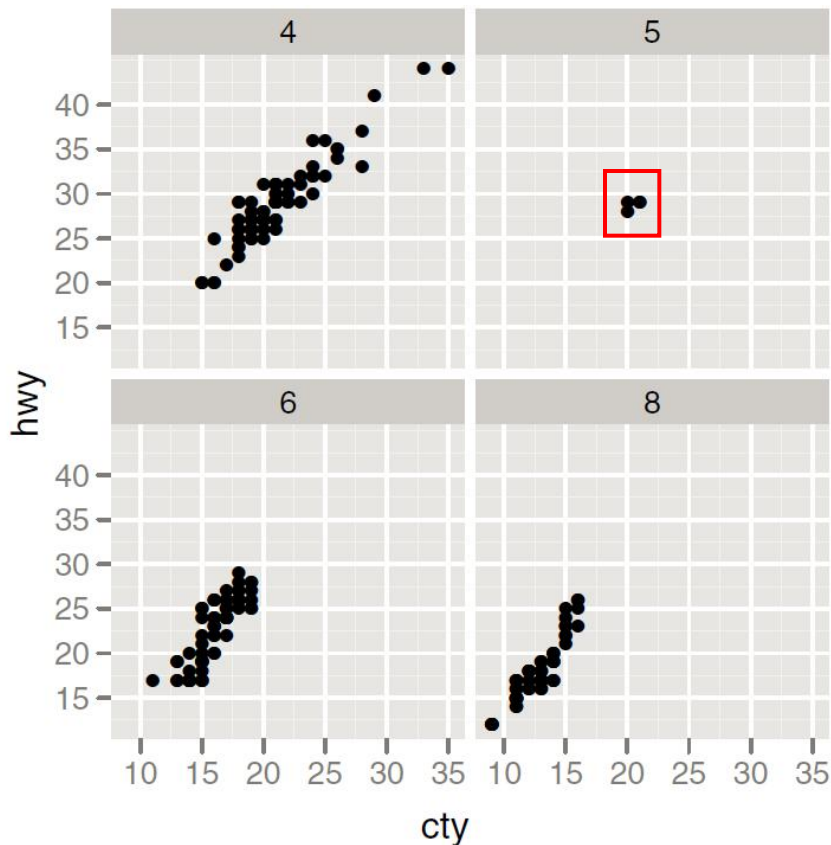
# 标度控制

- 对于两种分面，可以通过参数**scales**来控制所有面板的位置标度是相同的（固定）还是允许不同的（自由）。
  - scales = "**fixed**": x and y scales are fixed across all panels.
  - scales = "**free**": x and y scales vary across panels.
  - scales = "**free\_x**": the x scale is free, and the y scale is fixed.
  - scales = "**free\_y**": the y scale is free, and the x scale is fixed.

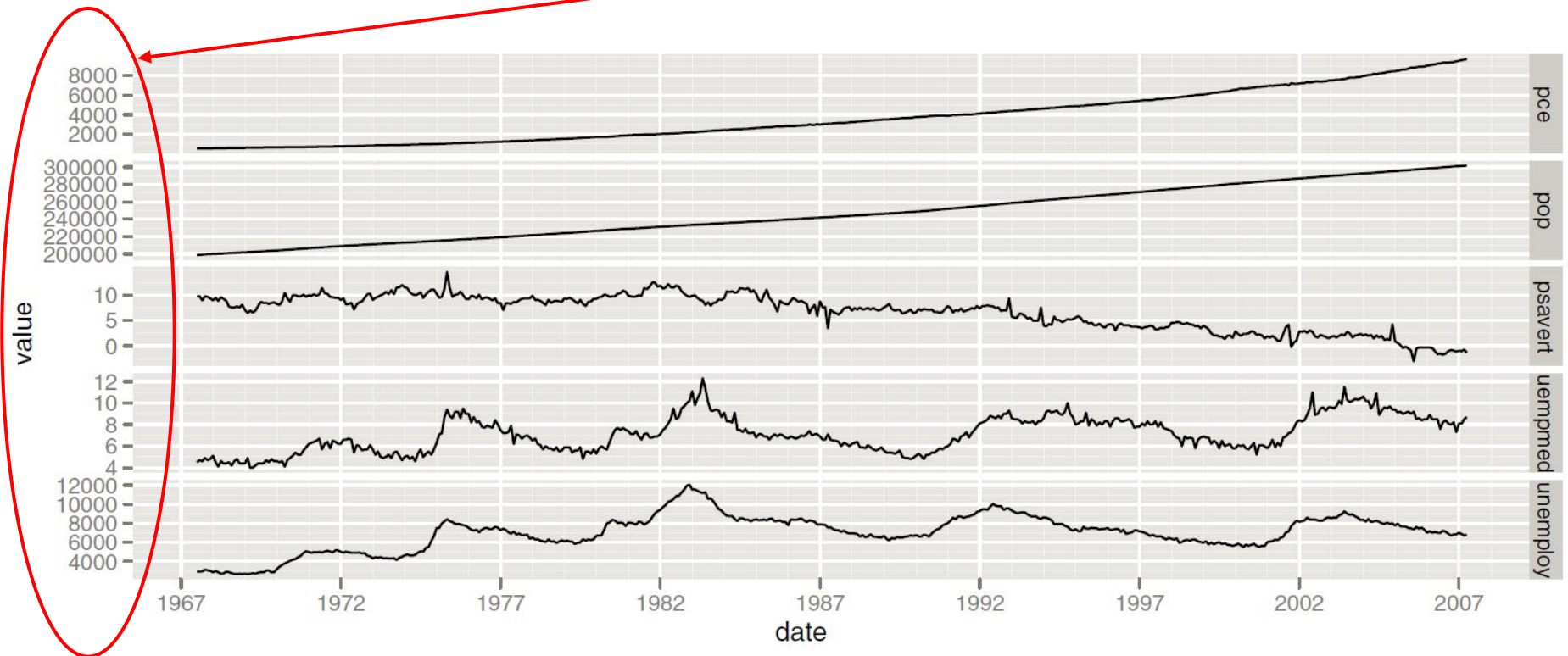
```
p <- qplot(cty, hwy, data = mpg)
```

```
p + facet_wrap(~ cyl) #左图
```

```
p + facet_wrap(~ cyl, scales = "free") #右图
```



```
library(reshape2)
em <- melt(economics, id = "date")
qplot(date, value, data = em, geom = "line", group =
variable) + facet_grid(variable ~ ., scale = "free_y")
```



# 网格分面facet\_grid额外注意事项

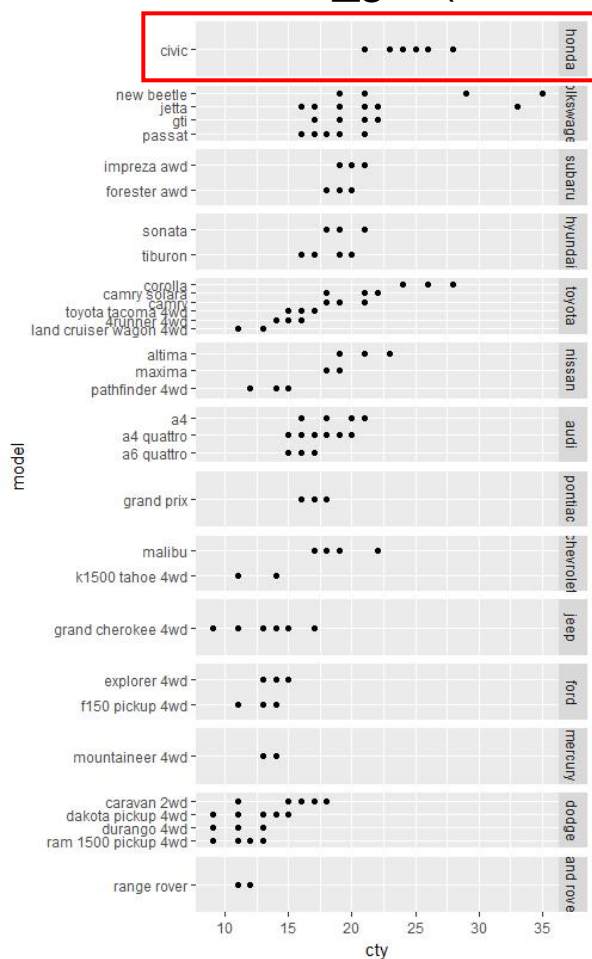
- 使用网格分面facet\_grid有个额外限制：同列的面板必须有相同的x标度，同行的面板必须有相同的y标度。
- facet\_grid还有一个额外参数space，值可为“free”、“fixed”、“free\_x”、“free\_y”。将其设定为free时，会使得所有面板的标度比例相同——这对分类标度非常有用。
- 例子见下页。

```
mpg3 <- within(mpg2, {
  model <- reorder(model, cty)
  manufacturer <- reorder(manufacturer, -cty)
})
```

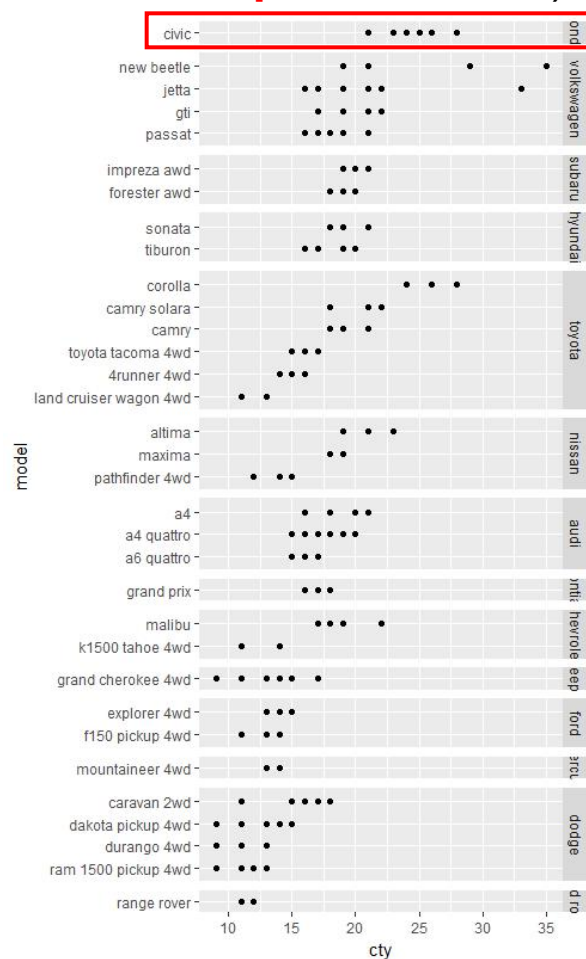
```
models <- qplot(cty, model, data = mpg3)
```

```
models + facet_grid(manufacturer ~ ., scales = "free") #左图
```

```
models + facet_grid(manufacturer ~ ., scales = "free", space = "free") #右图
```



所有面板一样大



所有面板的标度比例相同

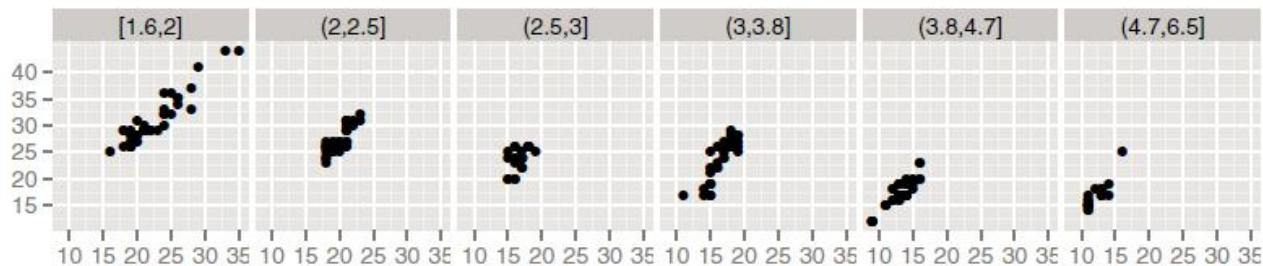
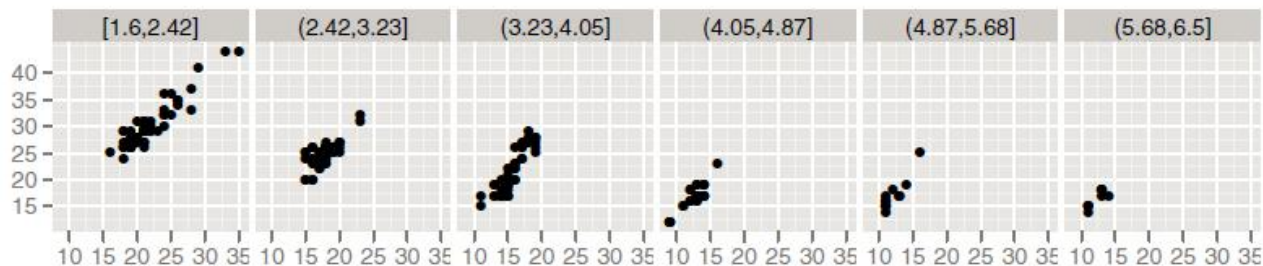
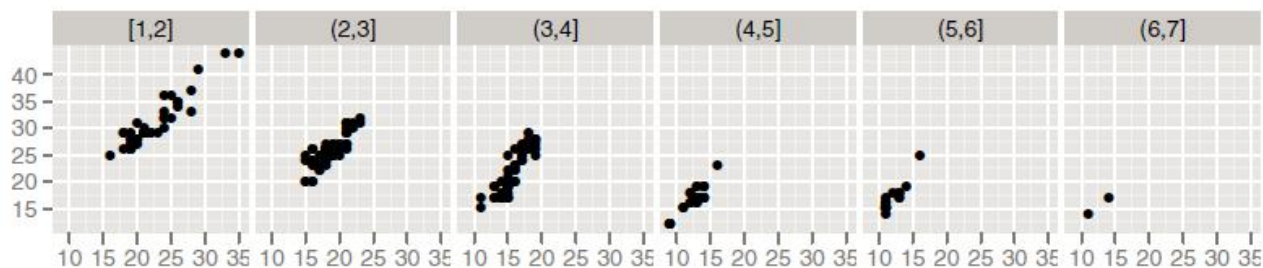
# 连续型变量

# 连续型变量

- 对连续型变量进行分面，首先需要将其转换为离散型。有两种转化方法：
  - 将数据分为n个长度相同的部分：用 `cut_interval(x, n = 10)` 控制划分数目，或用 `cut_interval(x, length = 1)` 控制每个部分的长度。
  - 将数据分为n个具有相同数目点的部分：  
`cut_number(x, n = 10)`
- 例子见下页



```
mpg2$disp_ww <- cut_interval(mpg2$displ, length = 1) #对应上图： n个长度相同的部分
mpg2$disp_wn <- cut_interval(mpg2$displ, n = 6) #对应中图： n个长度相同的部分
mpg2$disp_nn <- cut_number(mpg2$displ, n = 6) #对应下图： n个具有相同数目点的部分
plot <- qplot(cty, hwy, data = mpg2) + labs(x = NULL, y = NULL)
plot + facet_wrap(~ disp_ww, nrow = 1)
plot + facet_wrap(~ disp_wn, nrow = 1)
plot + facet_wrap(~ disp_nn, nrow = 1)
```



# 坐标系

# 坐标系（1/2）

- 坐标系是将两种位置标度结合在一起组成的二维定位系统。ggplot2包含了6种不同的坐标系。

名字	描述	
cartesian	笛卡尔坐标系	
equal	同尺度笛卡尔坐标系	} 本质都是笛卡尔型的坐标系
flip	翻转的笛卡尔坐标系	
trans	变换的笛卡尔坐标系	
map	地图射影	
polar	极坐标系	

- 坐标系命名规则为**coord\_**加上坐标系名字。
- 大部分图形都默认为笛卡尔坐标系，命名为**coord\_cartesian()**。

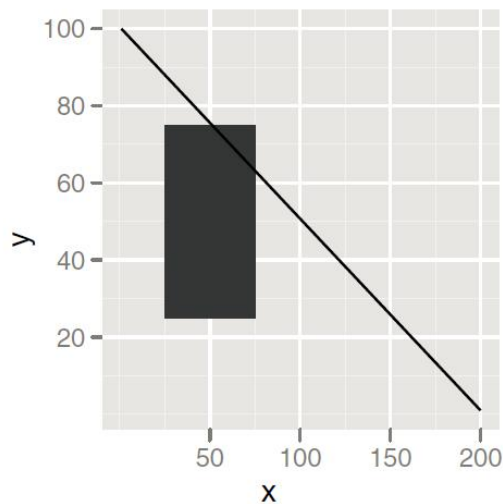
# 坐标系（2/2）

- 坐标系有两大功能：
  - 将两个位置图形属性组合起来在图形中形成二维方位系统。
  - 配合分面，坐标系可以绘出坐标轴和面板背景。

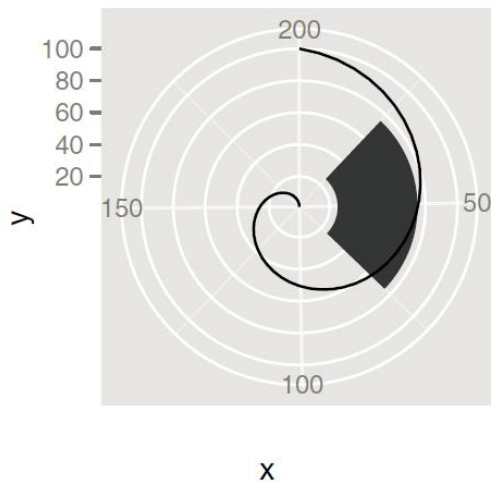
# 变换（1/2）

- 与数据变换和标度变换不同，坐标系变换将改变图形的几何形状：在极坐标系中，矩形变成了圆环的一部分；在地图中，两点间的最短路径将不是直线。
- 下页例子将展示直线和矩形在不同坐标系中的变换。

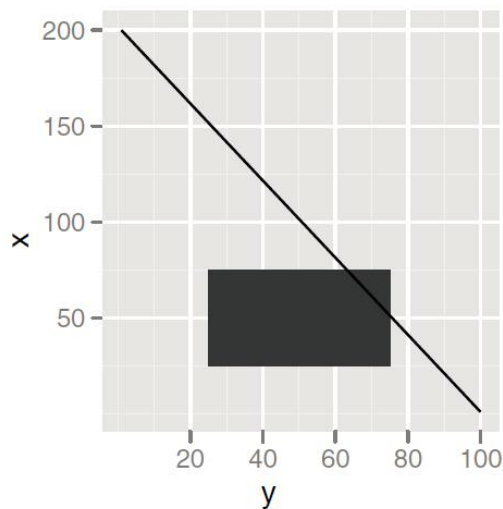
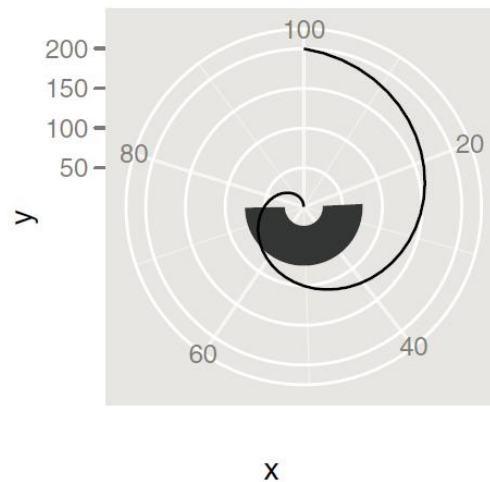
笛卡尔坐标系  
(x、y轴标度不同)



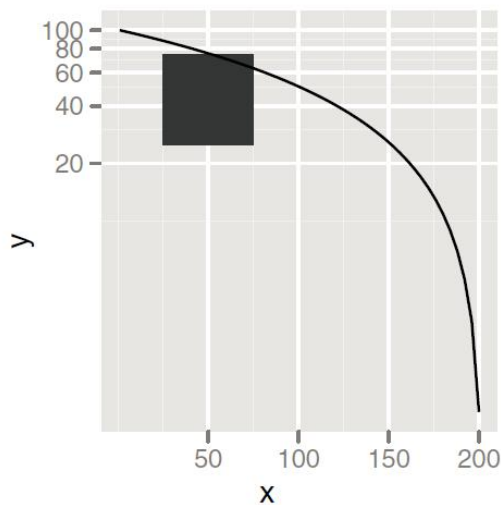
将x映射为角度的  
极坐标系



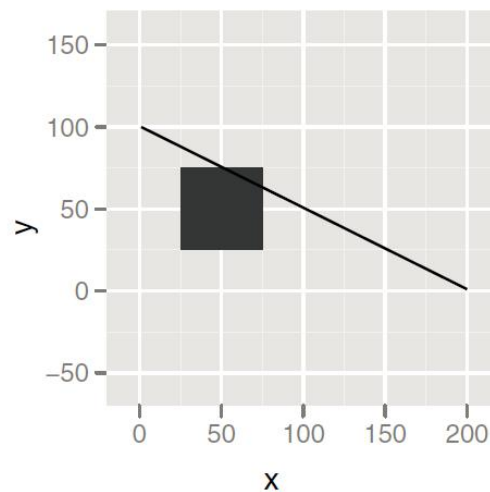
将y映射为角度的  
极坐标系



翻转的笛卡尔坐标系



y对数变换后的  
笛卡尔坐标系

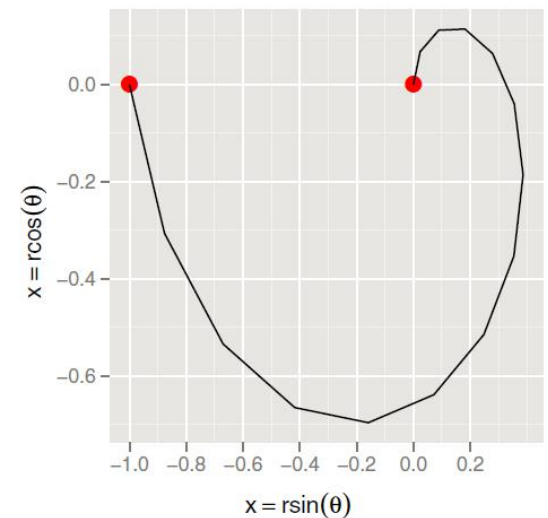
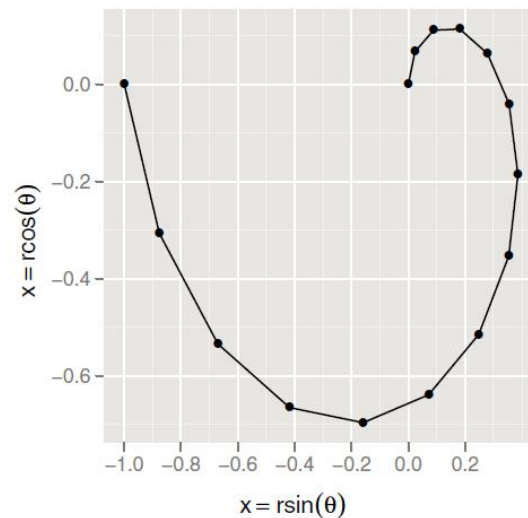
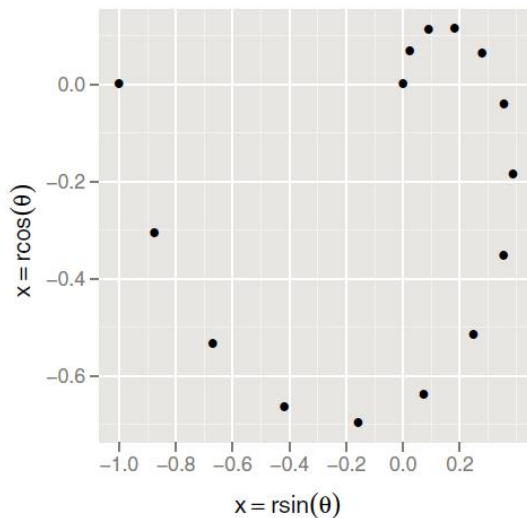
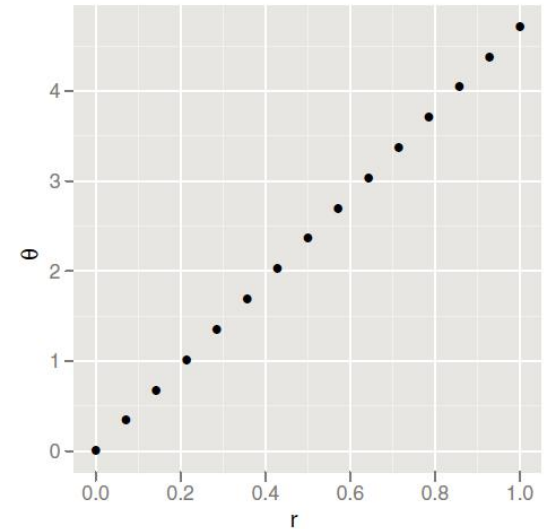
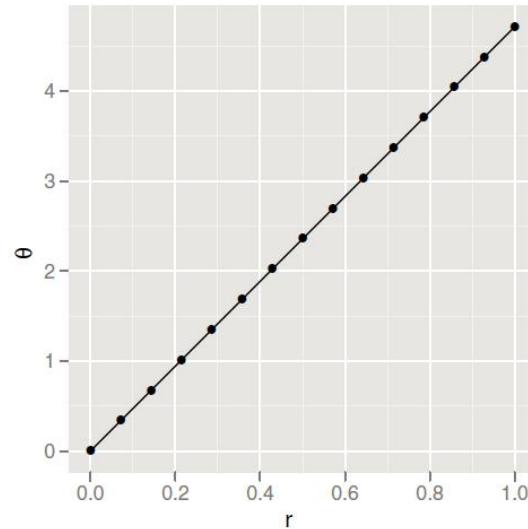
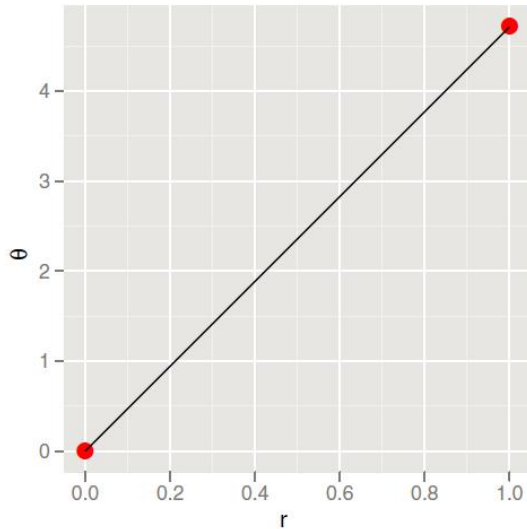


等标度坐标系

# 变换（2/2）

- 坐标系变换分为两步：
  - 将几何形状变为基于定位的表现形式。从而把所有的几何形状转化为点、线和多边形的组合。几何形状的参数变换只依据定位，而不是定位和维度。
  - 将每个位置转化到新的坐标系中。
    - 点的转化比较简单
    - 线和多边形可能会困难：因为直线在一个坐标系中不一定是直线：将线和多边形切割为许多小的线段后再进行变换，这个过程称为“分割再组合”（munching）。见下页图。

例：笛卡尔坐标系中的直线转化为极坐标中的曲线：  
先通过两个端点使线条参数化，然后将线切割为多个很小的线段，每个都有两个端点，再将这些点转化到新的坐标系中并重新连接。





# 统计量

# 统计量

- 尽管统计变换使用的统计方法应依赖坐标系，但`ggplot2`仍是在笛卡尔坐标系进行计算。所得结果是还是非常近似真实值的。

# 笛卡尔坐标系

- 有四种基于笛卡尔的坐标系：  
`coord_cartesian`, `coord_equal`, `coord_flip`  
和`coord_trans`。这四种坐标系本质上都是  
笛卡尔型的。

# 设置范围

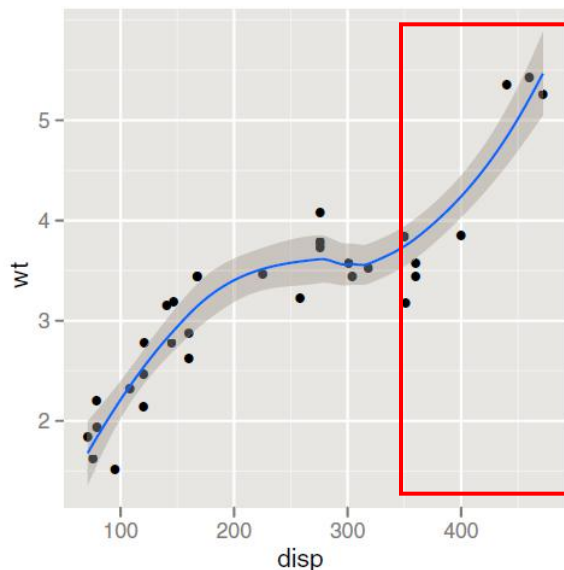
- `coord_cartesian`有两个参数`xlim`和`ylim`。该参数与标度范围参数的**区别在于其工作原理**：当设定标度范围参数时，超出范围的数据都会被删除；当设定笛卡尔坐标系的范围时，我们使用的仍是所有的数据，只不过只展示一小片图形区域（就像放大镜一样）。
- 例子见下页。

`(p <- qplot(displ, wt, data=mtcars) + geom_smooth())` #左图

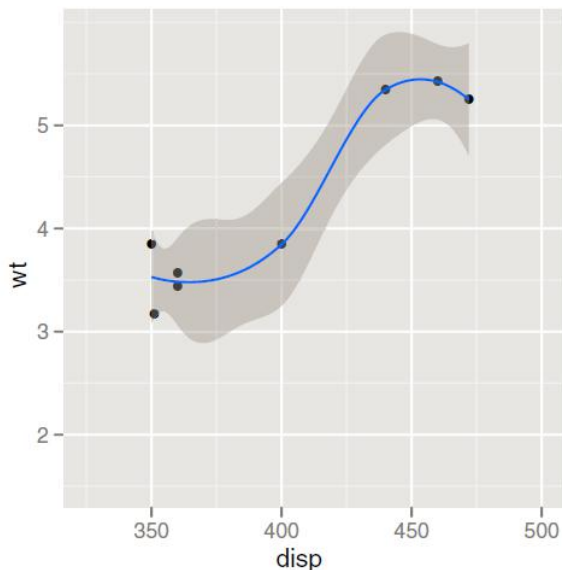
`p + scale_x_continuous(limits = c(325, 500))` #中图

`p + coord_cartesian(xlim = c(325, 500))` #右图

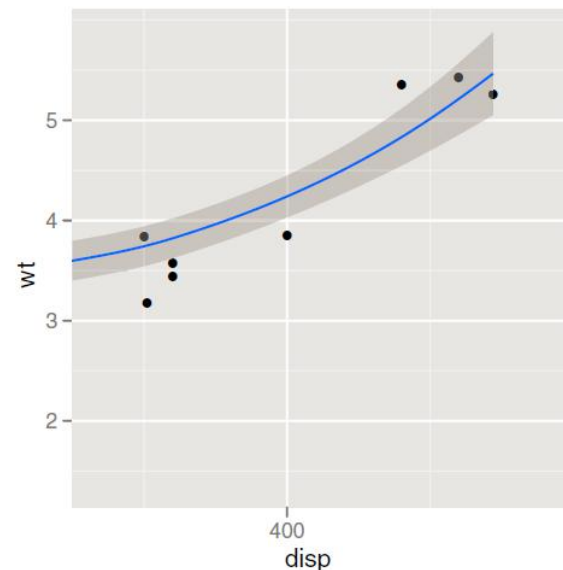
完整的数据集，  
红色为缩放范围。



标度的设置范围是对  
数据取子集，然后再  
重新拟合曲线。



坐标系的缩放就  
是图像的缩放。



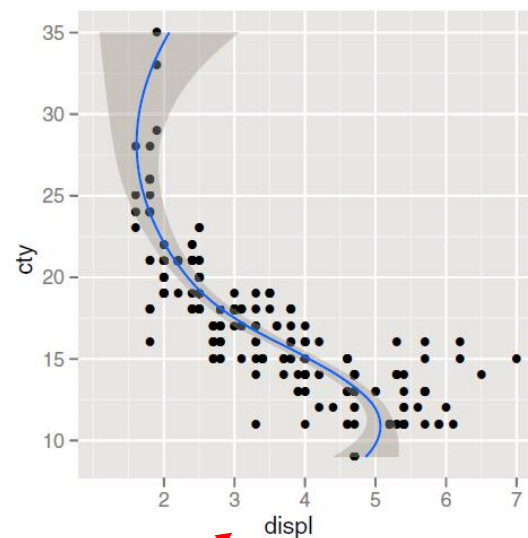
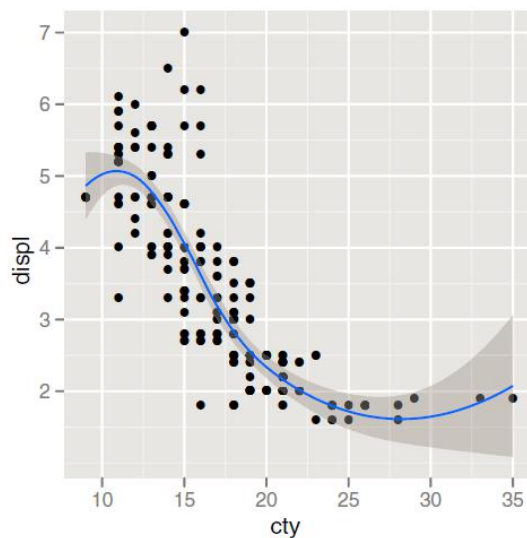
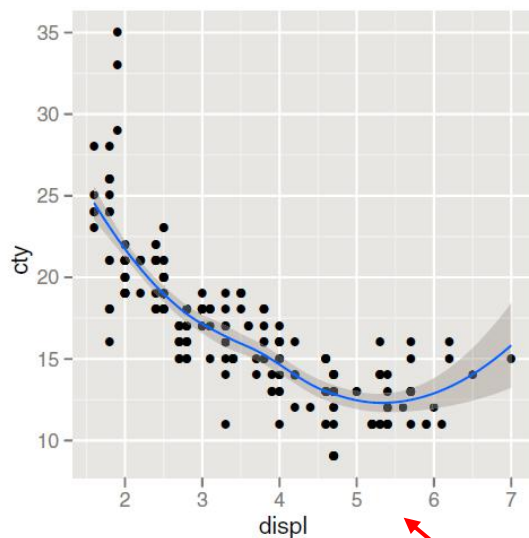
# 坐标轴翻转

- 大多数统计量和几何形状都假定我们只对x条件下的y值感兴趣，且都假定x的值测量无误差。
- 如果需要，可以使用**coord\_flip**调换x和y轴。

`qplot(displ, cty, data = mpg) + geom_smooth()` #左图

`qplot(cty, displ, data = mpg) + geom_smooth()` #中图

`qplot(cty, displ, data = mpg) + geom_smooth() + coord_flip()` #右图



尽管散点图一样，但拟合曲线不同

# 变换

- 可在变换的尺度上建模，然后再反演到变换前的图形，以便于解释——这是常见的分析模式。例：

```
qplot(carat, price, data = diamonds, log = "xy") +
```

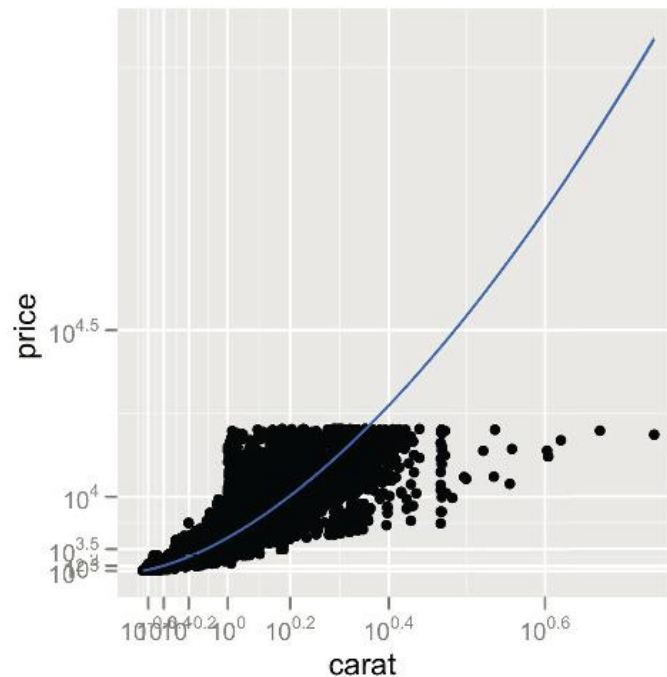
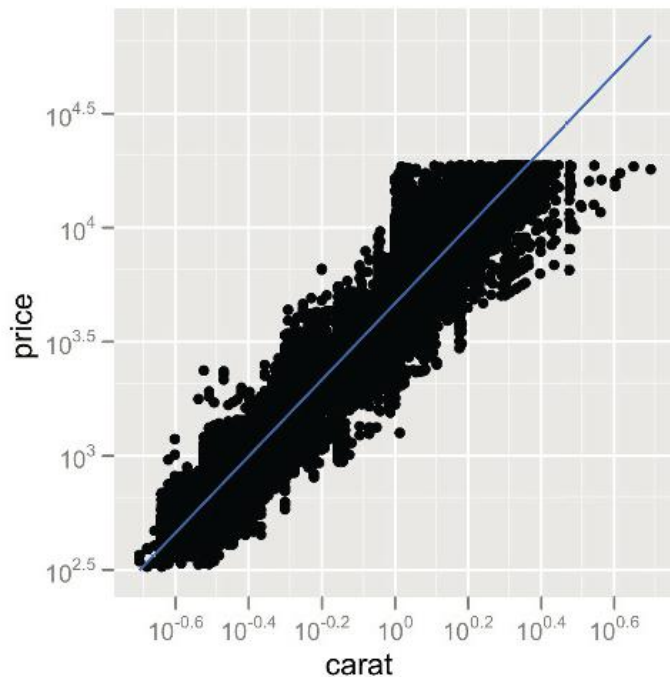
```
  geom_smooth(method = "lm") #左图
```

```
library(scales)
```

```
last_plot() + coord_trans(x = exp_trans(10), y = exp_trans(10))
```

上一次  
绘的图 ↘

左图：  
克拉与  
价格对  
数变换  
后的散  
点图。  
直线为  
回归曲  
线。



右图：  
将左面  
的图变  
换回去，  
标度还  
原，因  
此线性  
趋势变  
成指数  
形式。