
ggplot2による データ可視化

里 洋平 (@yokkuns)

第42回Tokyo.R 初心者セッション

2014.08.30

AGENDA

- 自己紹介
- ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図

- 自己紹介
 - ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図
-
-

里 洋平 (@yokkuns)

- 略歴
 - ヤフー
 - DeNA
 - ドリコム
- やつてること
 - いろんなデータ分析
 - 分析環境の整備



バックグラウンド

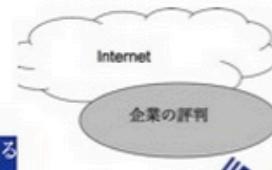
システム開発
Webアプリ開発
Androidアプリ開発

プロモーション
CM時系列
各KPIの時系列

購買履歴
購買履歴データ
欠測
広告接触データ
共通で得られている属性データ

Data Fusion

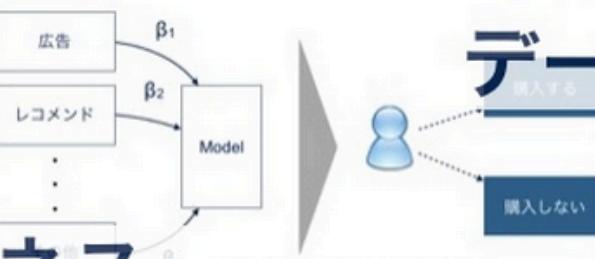
動画レコメンド



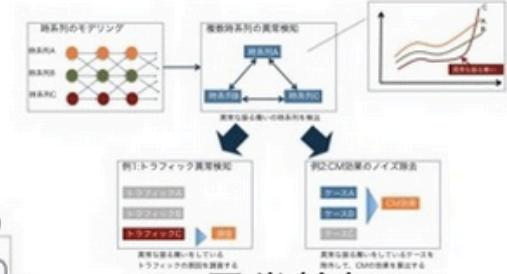
株価市場予測

Web上にある評判情報から
市場予測！

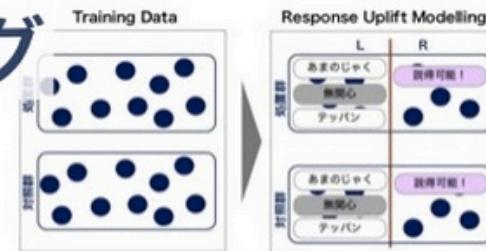
統計モデリング データマイニング



選択行動予測



異常検知



分類結果
"R"の場合
良いターゲット

行動変化モデリング

Tokyo.Rの主催やっています

第42回R勉強会@東京 (#TokyoR)

 Tweet 28

 Like 1

 g+1 4

 B! 0



日時： 2014/08/30 14:30 to 19:30

定員： 130 人

会場： ヤフー株式会社 11階 イベントスペース（東京都港区赤坂9-7-1）

URL： <http://groups.google.com/group/r-study-tokyo>

管理者： yokkuns



ログインしてください

参加するためにはログインが必要です

参加希望者 **138** / 130 人

参加：130 / 補欠：8 / キャンセル：13

著書



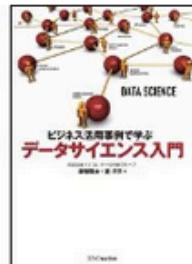
戦略的データマイニング (シリーズ Useful R 4) 里 洋平、金 明哲 (2014/8/23)

¥ 3,888 単行本 ✓[プライム](#)

一時的に在庫切れ；入荷時期は未定です

こちらからもご購入いただけます - 単行本

¥ 5,740 中古品 (4 出品)



ビジネス活用事例で学ぶ データサイエンス入門 酒巻 隆治、里 洋平 (2014/6/25)

¥ 2,376 単行本 ✓[プライム](#)

2014/8/29 金曜日中にお届け

¥ 2,200 Kindle版

今すぐダウンロード

★★★★★ (23)



Rではじめる ビジネス統計分析 末吉 正成、里 洋平、酒巻 隆治、小林 雄一郎 (2014/7/18)

¥ 3,456 大型本 ✓[プライム](#)

2014/8/29 金曜日中にお届け

¥ 3,200 Kindle版

今すぐダウンロード

こちらからもご購入いただけます - 大型本

¥ 2,878 中古品 (2 出品)

★★☆☆☆ (4)



データサイエンティスト養成読本 [ビッグデータ時代のビジネスを支えるデータ分析力が身につく!] (Software Design plus)
佐藤 洋行、原田 博植、下田 倫大、大成 弘子 (2013/8/8)

¥ 2,138 大型本 ✓[プライム](#)

18点在庫あり。ご注文はお早めに。

こちらからもご購入いただけます - 大型本

¥ 1,700 中古品 (6 出品)

★★★★☆ (6)



Rパッケージガイドブック 岡田 昌史、荒木 孝治、伊藤 康広、里 洋平 (2011/4/9)

¥ 4,104 単行本 ✓[プライム](#)

1点在庫あり。ご注文はお早めに。

こちらからもご購入いただけます - 単行本

¥ 3,800 中古品 (7 出品)

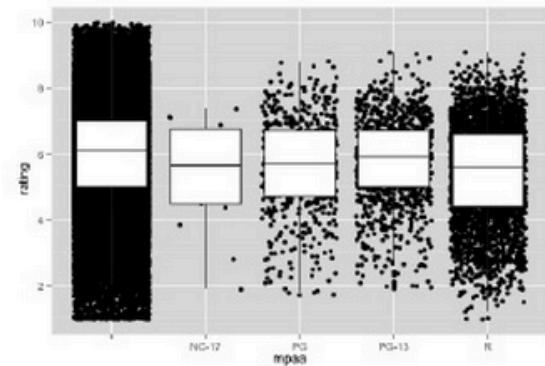
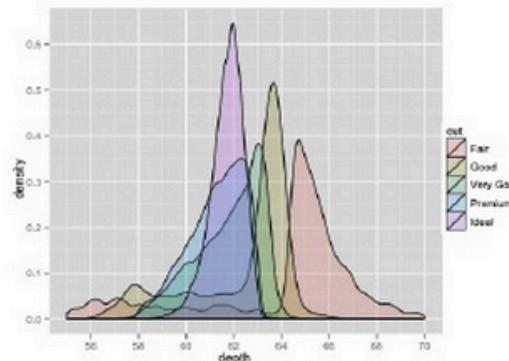
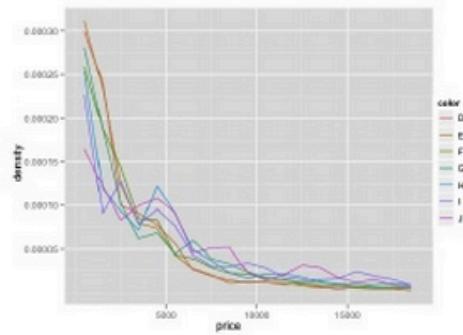
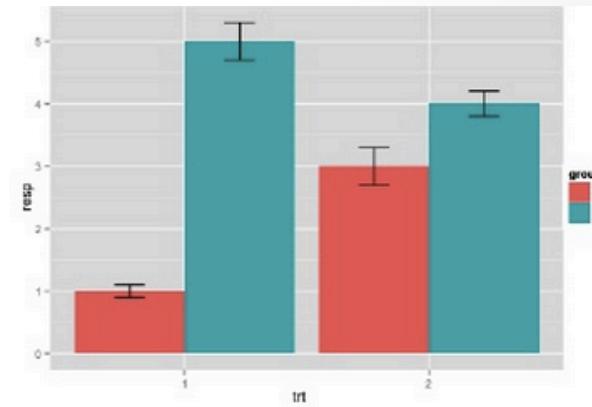
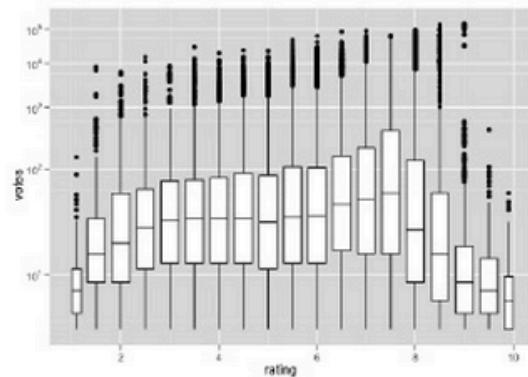
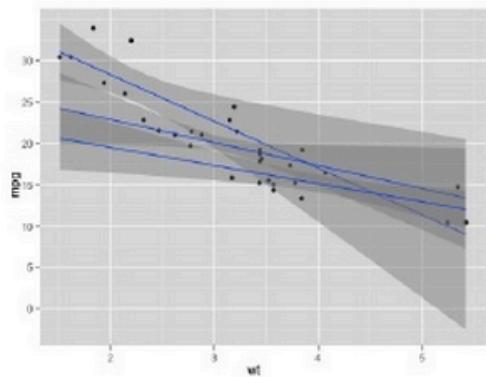
★★☆☆☆ (2)

AGENDA

- 自己紹介
- ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図

ggplot2パッケージとは

統一されたインターフェースとオプションで
簡単にきれいでパワフルなグラフが作成出来るパッケージ



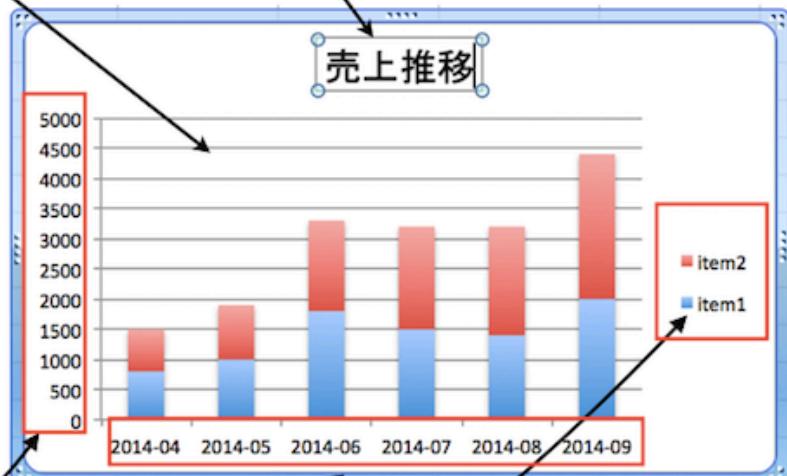
ggplot2によるグラフ作成のイメージ

```
ggplot(data, aes(x=month, y=sales, fill=item))  
+ geom_bar() + ggtitle("売上推移")
```

データリスト形式の表

	A	B	C	D
1	item	month	month	sales
2	item1	4	2014-04	800
3	item1	5	2014-05	1000
4	item1	6	2014-06	1800
5	item1	7	2014-07	1500
6	item1	8	2014-08	1400
7	item1	9	2014-09	2000
8	item2	4	2014-04	700
9	item2	5	2014-05	900
10	item2	6	2014-06	1500
11	item2	7	2014-07	1700
12	item2	8	2014-08	1800
13	item2	9	2014-09	2400

作成されるグラフ



y=sales

x=month

fill=item

ggplot2によるグラフ作成の流れ

①データの選択

	A	B	C
1	speed	dist	
2	4	2	
3	4	10	
4	7	4	
5	7	22	
6	8	16	
7	9	10	
8	10	18	
9	10	28	
10	10	34	
11	11	17	

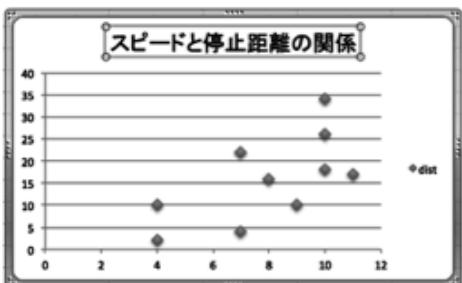
```
ggplot(cars, aes(x=speed, y=dist))
```

②グラフの選択



```
+ geom_point()
```

③グラフのタイトルや軸ラベルの設定



```
+ ggtitle("スピードと停止距離の関係")
```

仮想の売上データ

- 商品1の月次売上データ
- 商品1,2の月次売上データ
- 店舗1～3の商品3,4の日次売上データ

データの作成 ①：商品1の月次売上データ

```
sales.data1 <- data.frame(  
  item="item1",  
  m=4:9,  
  month=paste0("2014-0",c(4:9)),  
  sales=c(800,1000,1800,1500,1400,2000))  
  
sales.data1
```

```
##  item m  month sales  
## 1 item1 4 2014-04  800  
## 2 item1 5 2014-05 1000  
## 3 item1 6 2014-06 1800  
## 4 item1 7 2014-07 1500  
## 5 item1 8 2014-08 1400  
## 6 item1 9 2014-09 2000
```

データの作成 ②：商品1,2の月次売上データ

```
sales.data2 <- data.frame(  
  item="item2",  
  m=4:9,  
  month=paste0("2014-0",c(4:9)),  
  sales=c(700,900,1500,1700,1800,2400))  
sales.data12 <- rbind(sales.data1, sales.data2)  
sales.data12
```

```
##   item m  month sales  
## 1 item1 4 2014-04  800  
## 2 item1 5 2014-05 1000  
## 3 item1 6 2014-06 1800  
## 4 item1 7 2014-07 1500  
## 5 item1 8 2014-08 1400  
## 6 item1 9 2014-09 2000  
## 7 item2 4 2014-04  700  
## 8 item2 5 2014-05  900  
## 9 item2 6 2014-06 1500  
## 10 item2 7 2014-07 1700  
## 11 item2 8 2014-08 1800  
## 12 item2 9 2014-09 2400
```

データの作成 ③：店舗1～3の商品3,4の日次売上データ

```
set.seed(10)
sales.data34 <- data.frame(
  shop=sample(1:3,100,replace=T),
  item3=rnorm(100,100,30)
)

sales.data34$item4 <- round(0.9 * sales.data34$item3 +
  80 * sales.data34$shop +
  rnorm(100,30,15))

sales.data34$shop <- paste0("shop",sales.data34$shop)
head(sales.data34)
```

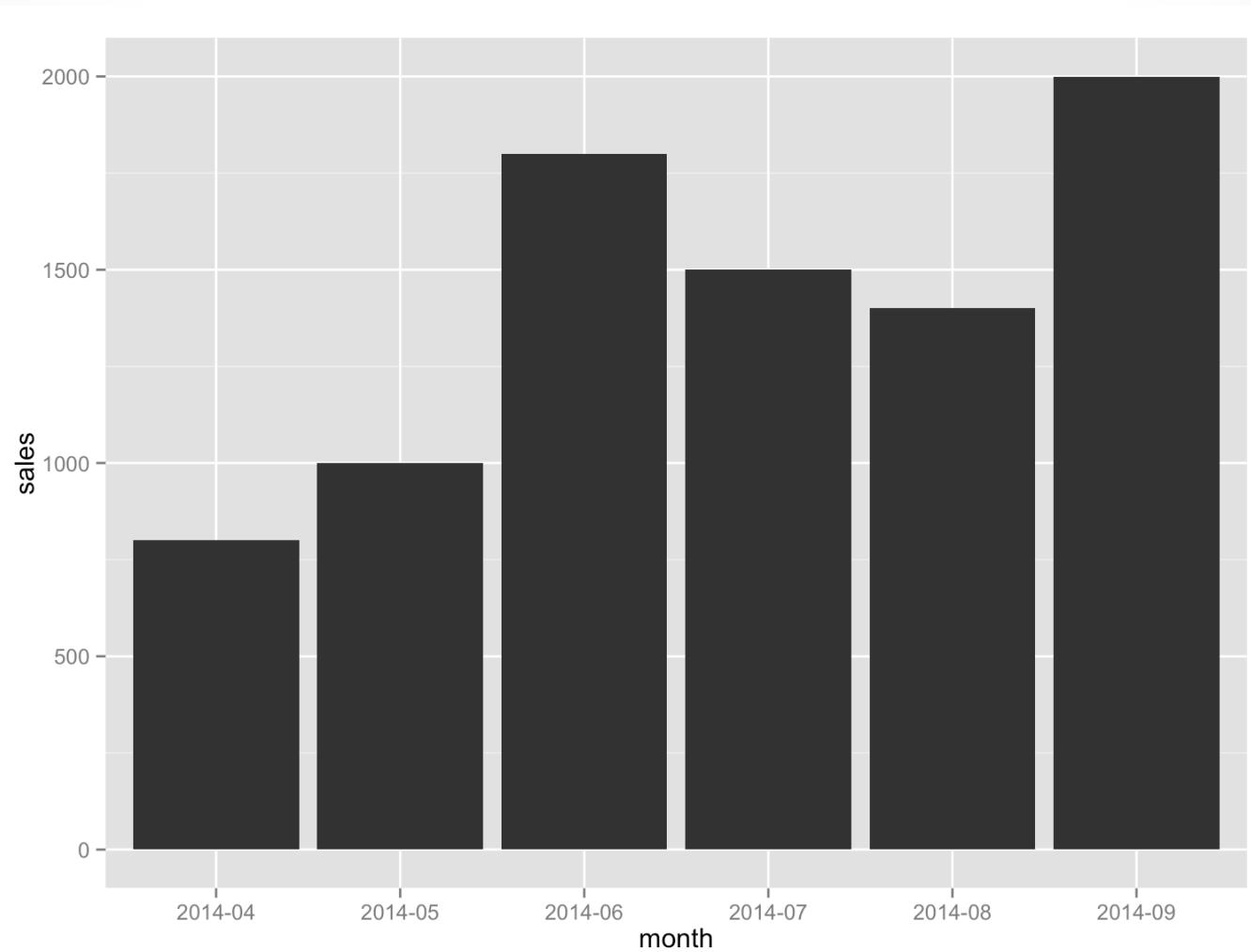
```
##   shop item3 item4
## 1 shop2  87.98  263
## 2 shop1  89.96  187
## 3 shop2 141.04  334
## 4 shop3 164.13  405
## 5 shop1 115.17  201
## 6 shop1 123.59  186
```

AGENDA

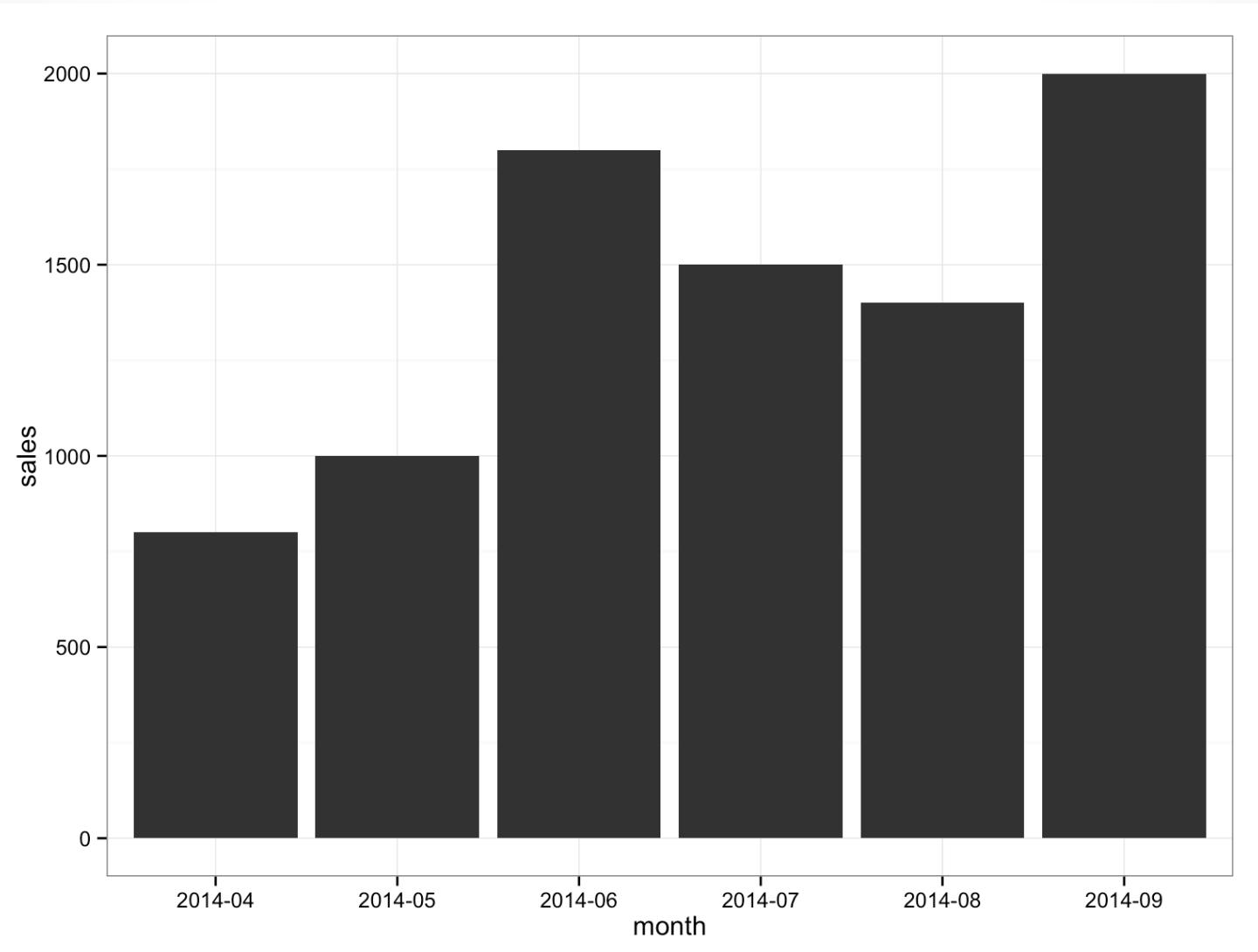
- 自己紹介
 - ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図
-
-

基本的な棒グラフ

```
ggplot(sales.data1, aes(x=month, y=sales)) +  
  geom_bar(stat="identity")
```



背景色を白にしたい



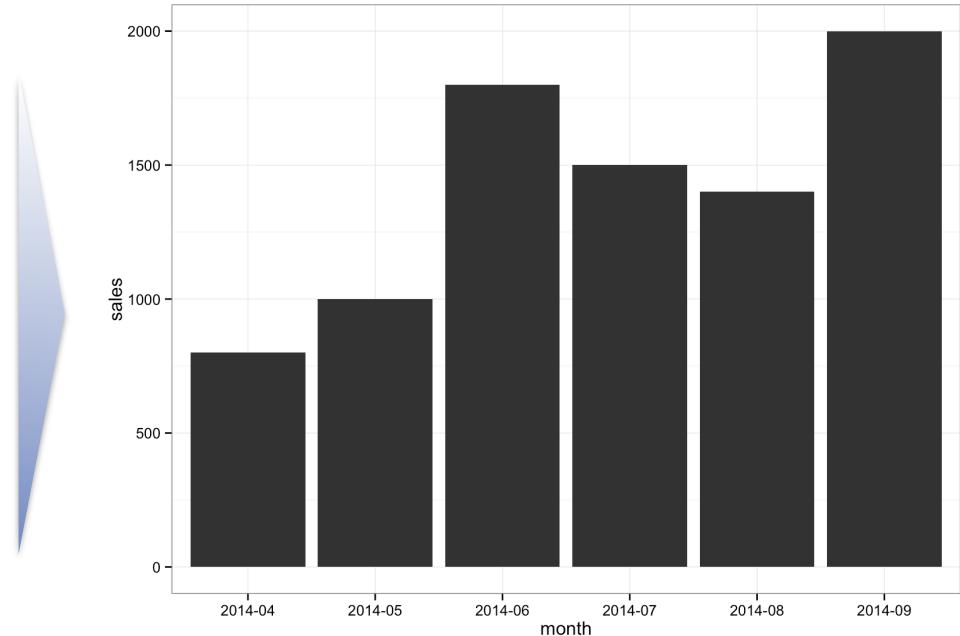
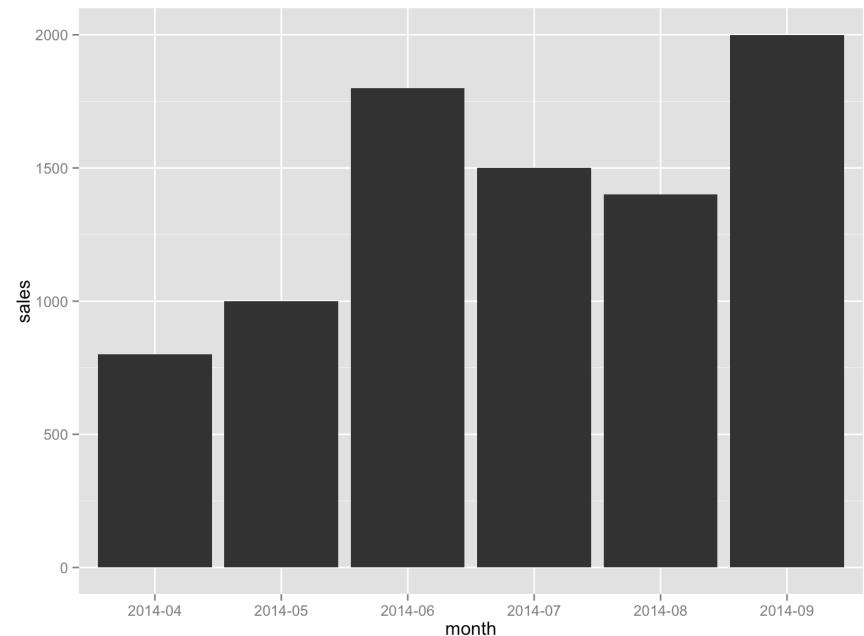
theme_bw関数を使う

```
# 背景を白にする  
ggplot(sales.data1, aes(x=month, y=sales)) +  
  geom_bar(stat="identity") +  
  theme_bw()
```

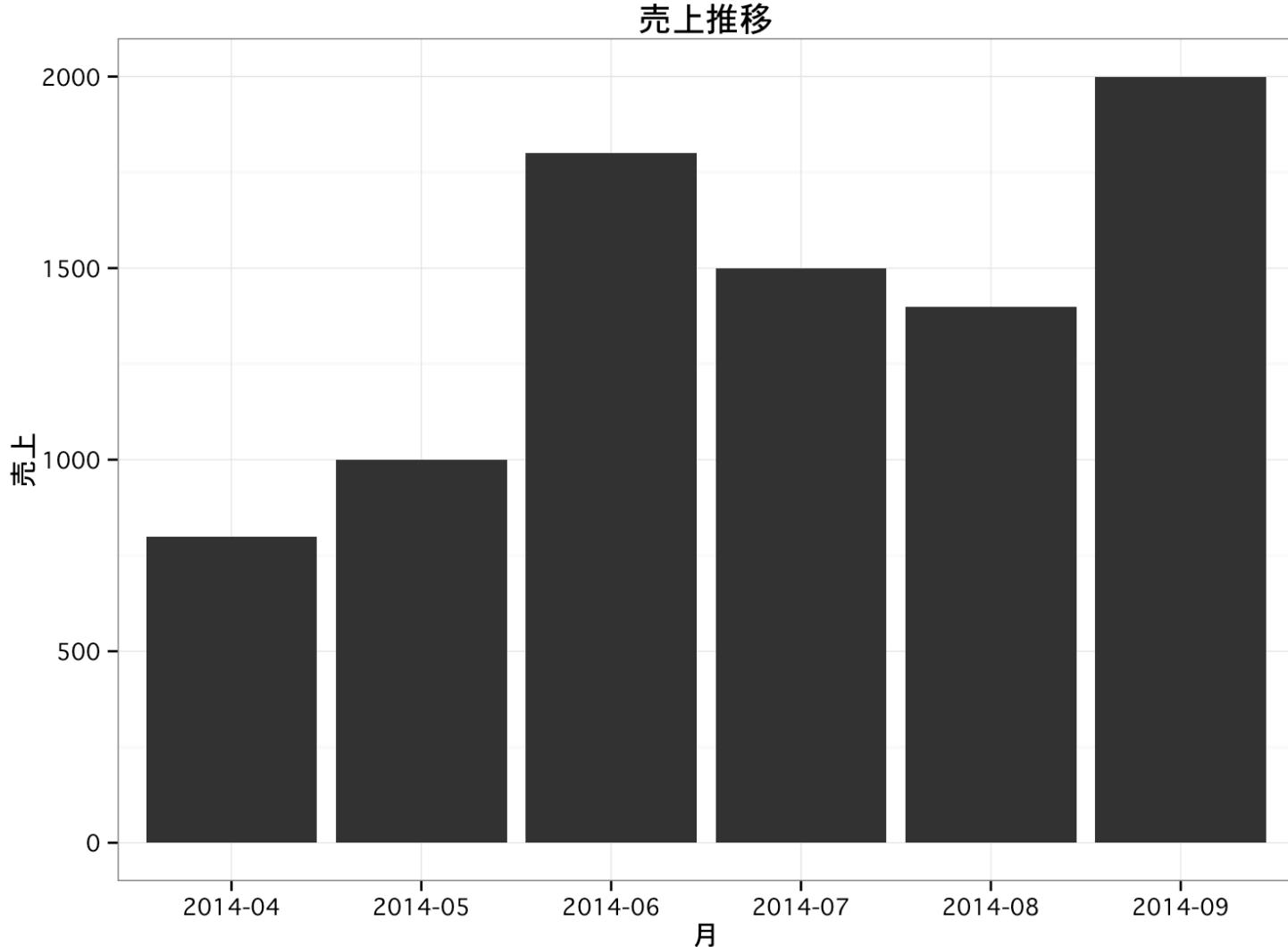
背景色の変更（結果）

背景を白にする

```
ggplot(sales.data1, aes(x=month, y=sales)) +  
  geom_bar(stat="identity") +  
  theme_bw()
```



軸ラベルやタイトルを日本語にしたい



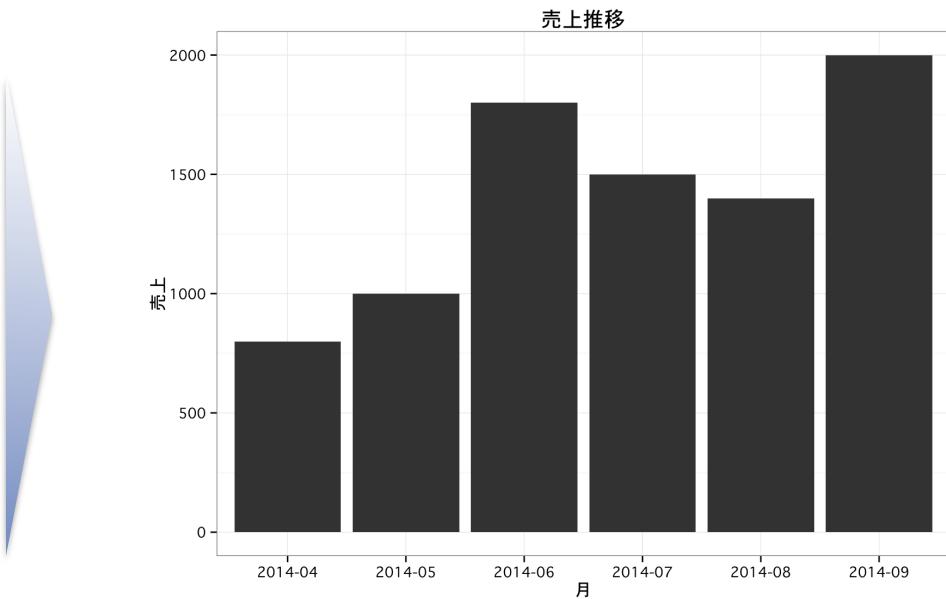
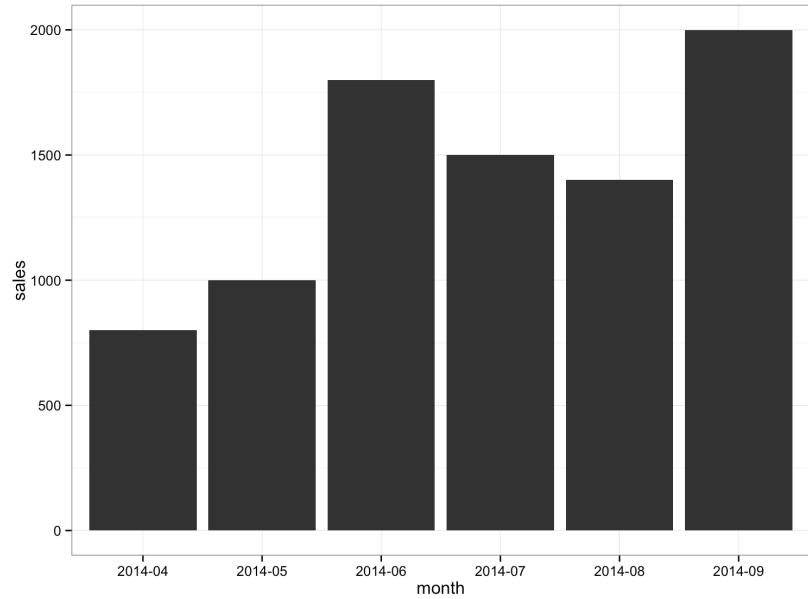
軸ラベル・タイトルの変更（Rコード）

xlab、ylab、ggttitle関数を使う

```
ggplot(sales.data1, aes(x=month, y=sales)) +  
  geom_bar(stat="identity") +  
  xlab("月") +  
  ylab("売上") +  
  ggttitle("売上推移") +  
  theme_bw(base_family="Osaka")
```

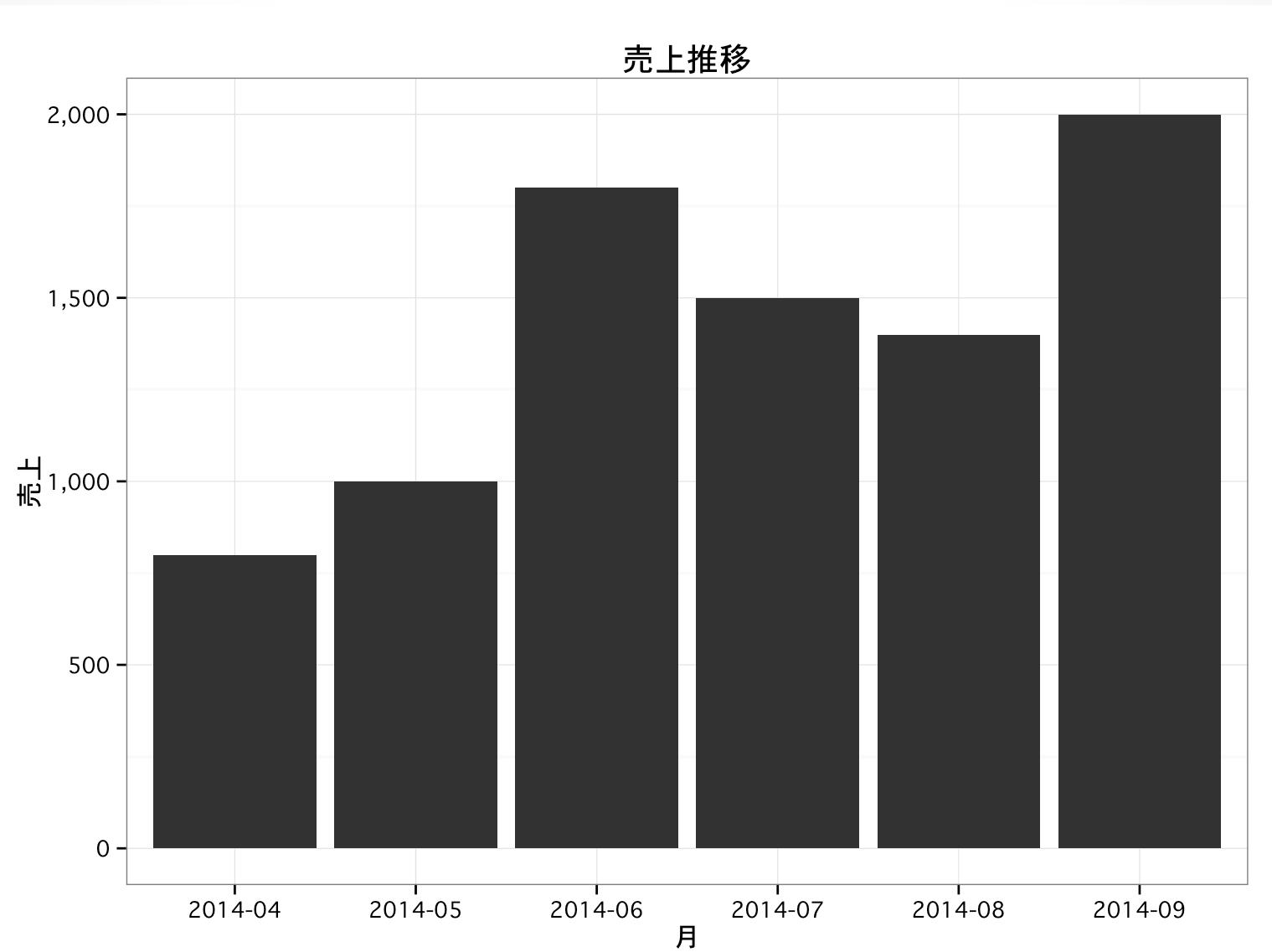
軸ラベル・タイトルの変更（結果）

```
ggplot(sales.data1, aes(x=month, y=sales)) +  
  geom_bar(stat="identity") +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  theme_bw(base_family="Osaka")
```



数字フォーマットの変更

y軸の数字をカンマ区切りにしたい



数字フォーマットの変更 (Rコード)

scalesパッケージのcomma関数を使う

```
# 数値フォーマット用のライブラリ
library(scales)

ggplot(sales.data1, aes(x=month, y=sales)) +
  geom_bar(stat="identity") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

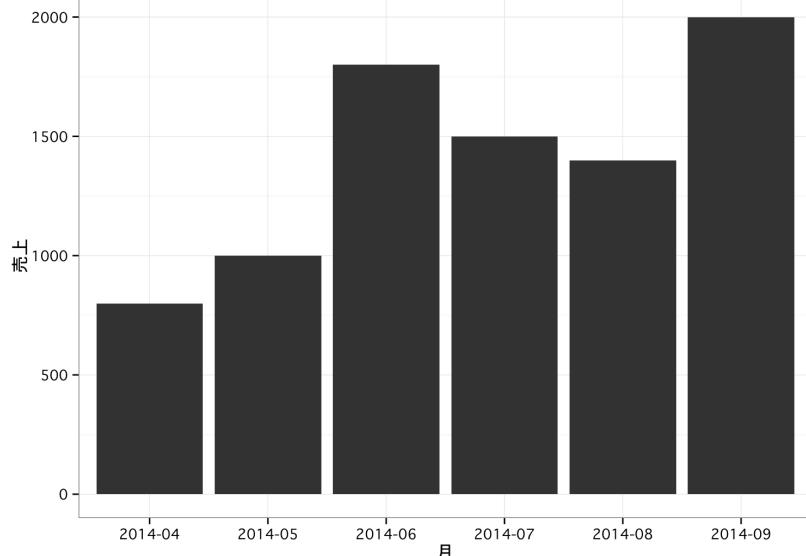
数字フォーマットの変更（結果）

```
# 数値フォーマット用のライブラリ
```

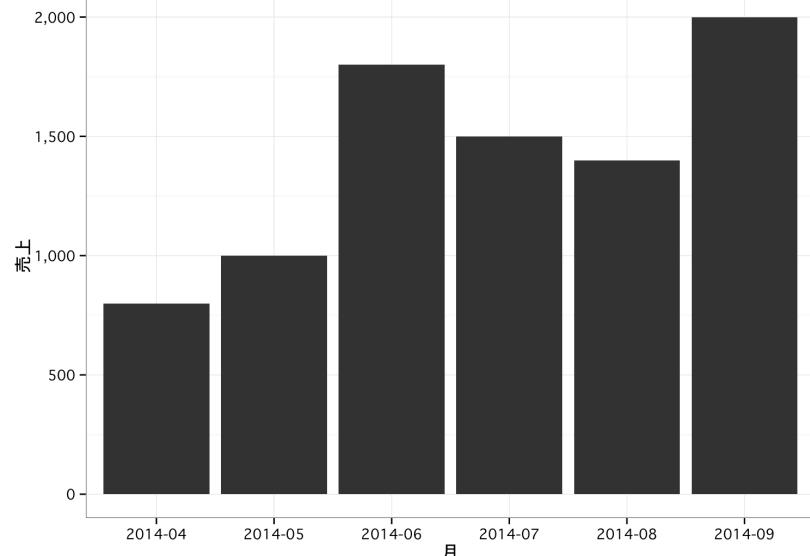
```
library(scales)
```

```
ggplot(sales.data1, aes(x=month, y=sales)) +  
  geom_bar(stat="identity") +  
  xlab("月") +  
  ylab("売上") +  
  qqttitle("売上推移") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```

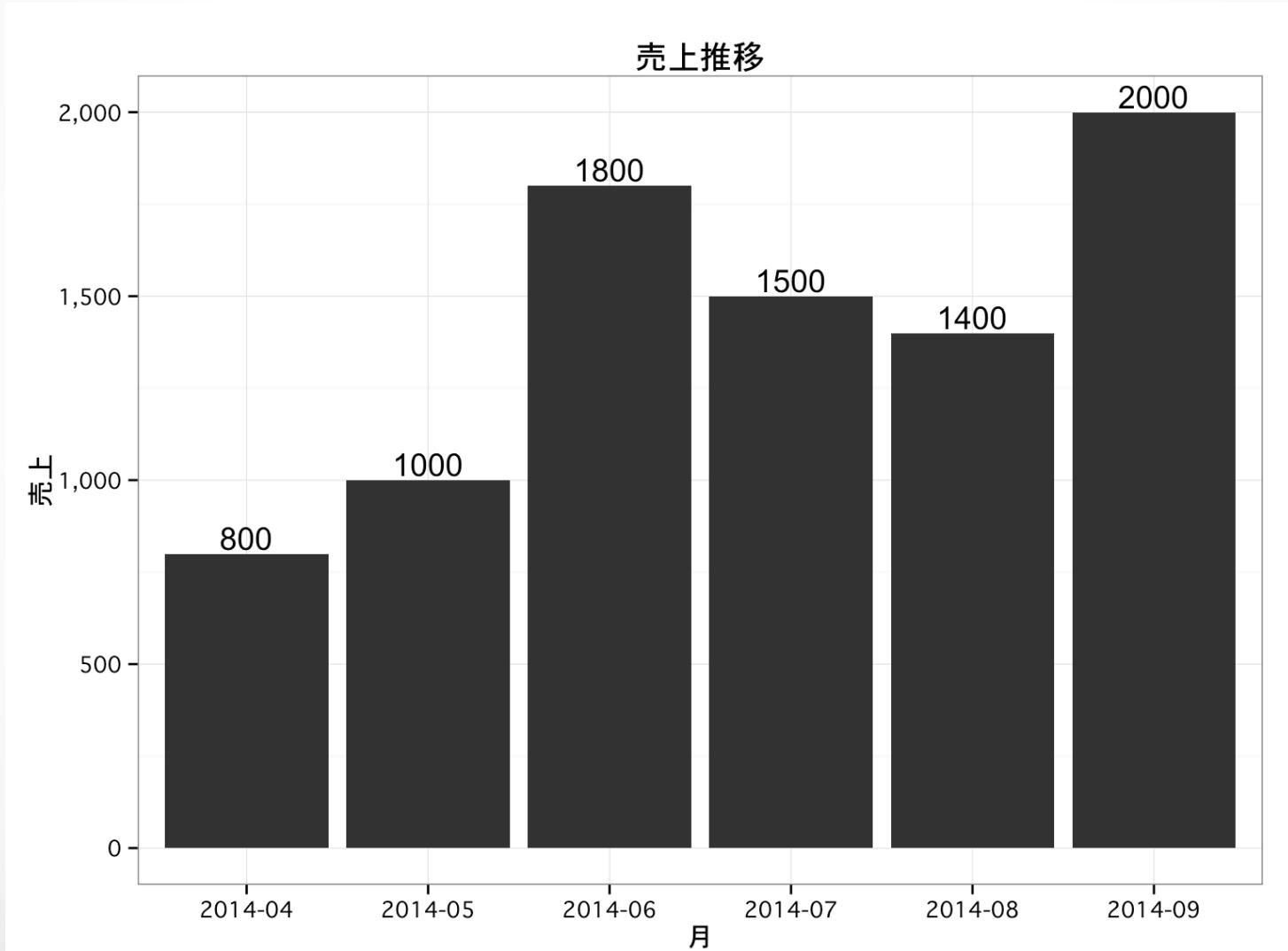
売上推移



売上推移



棒グラフの上に数字を載せたい

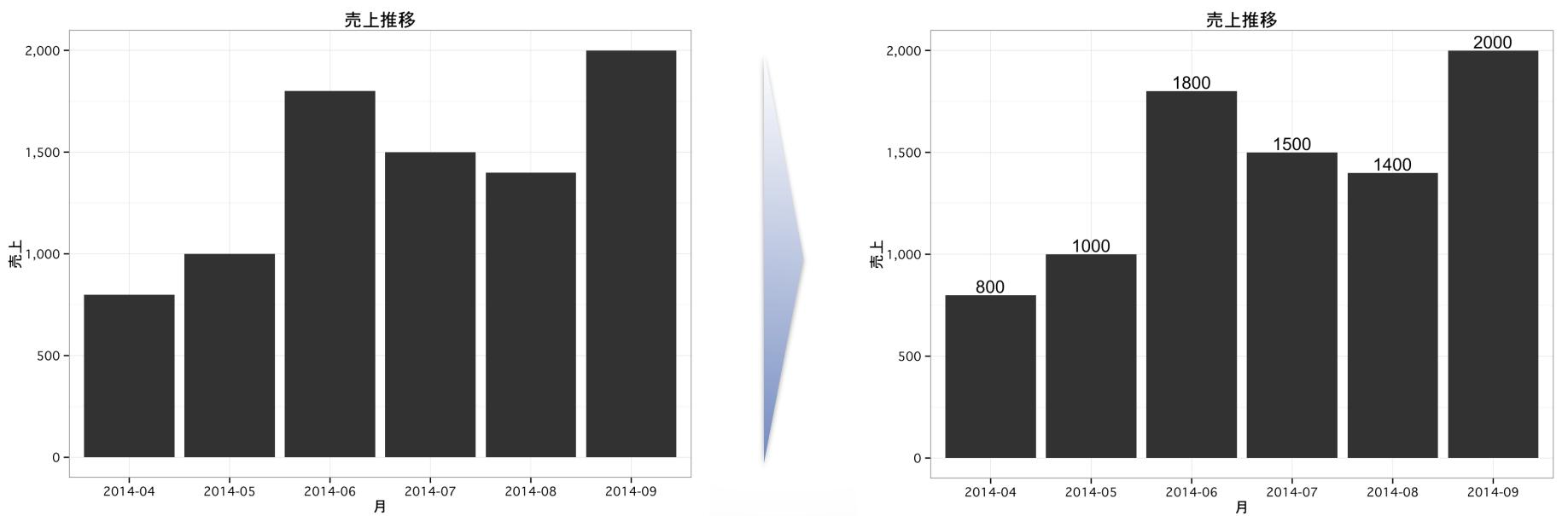


geom_text関数を使う

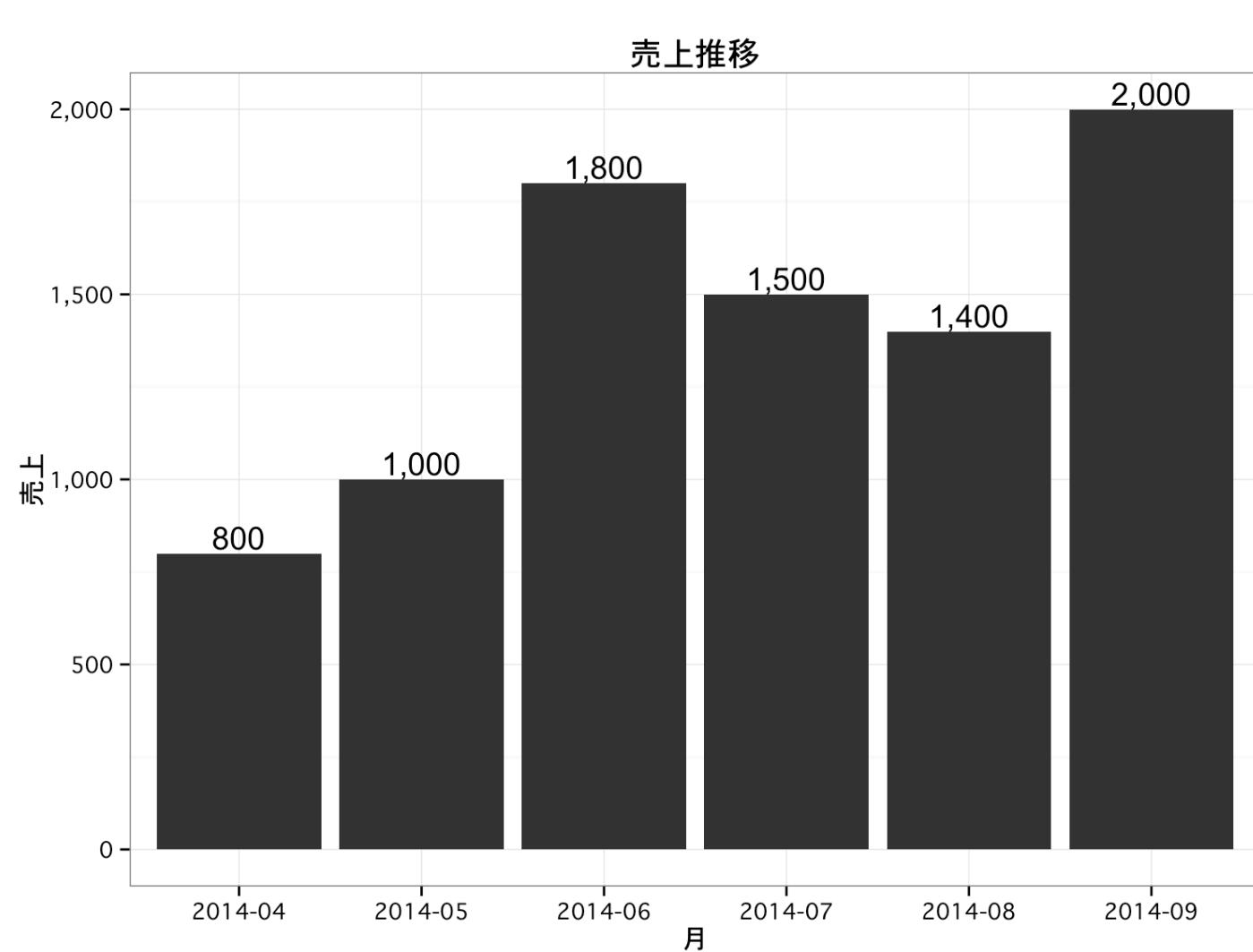
```
## 数値を出力する
ggplot(sales.data1, aes(x=month, y=sales)) +
  geom_bar(stat="identity") +
  geom_text(aes(label=sales), vjust=-0.2) + [Red box]
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

テキストの追加（結果）

```
## 数値を出力する
ggplot(sales.data1, aes(x=month, y=sales)) +
  geom_bar(stat="identity") +
  geom_text(aes(label=sales), vjust=-0.2) +
  xlab("月") +
  ylab("売上") +
  ggttitle("売上推移") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```



テキスト（数字）をカンマ区切りにしたい



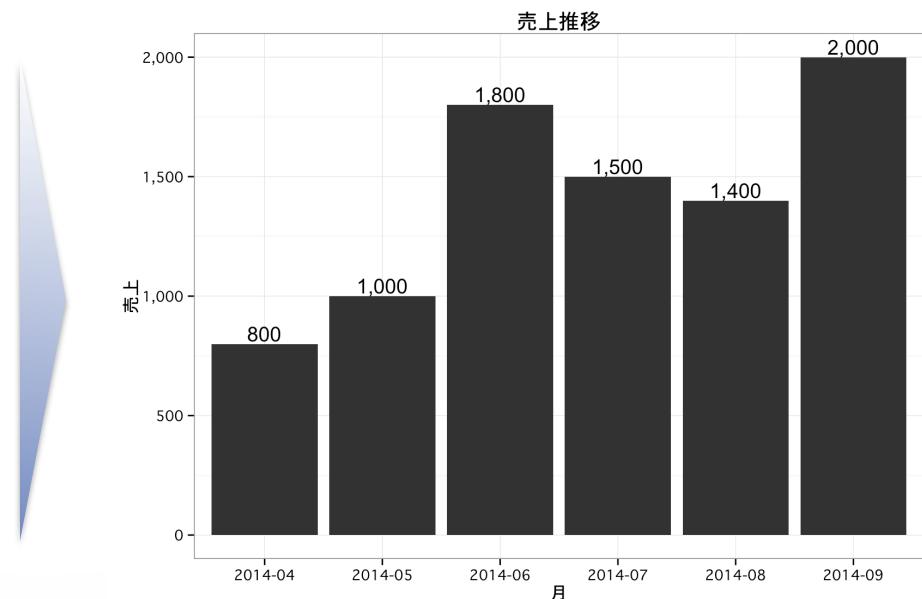
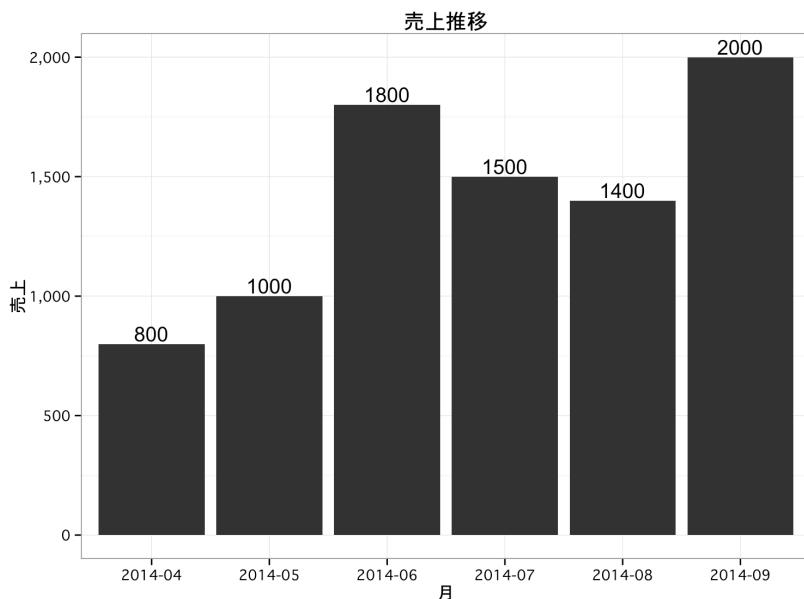
追加的ストの加工 (Rコード)

scalesパッケージのcomma関数を使う

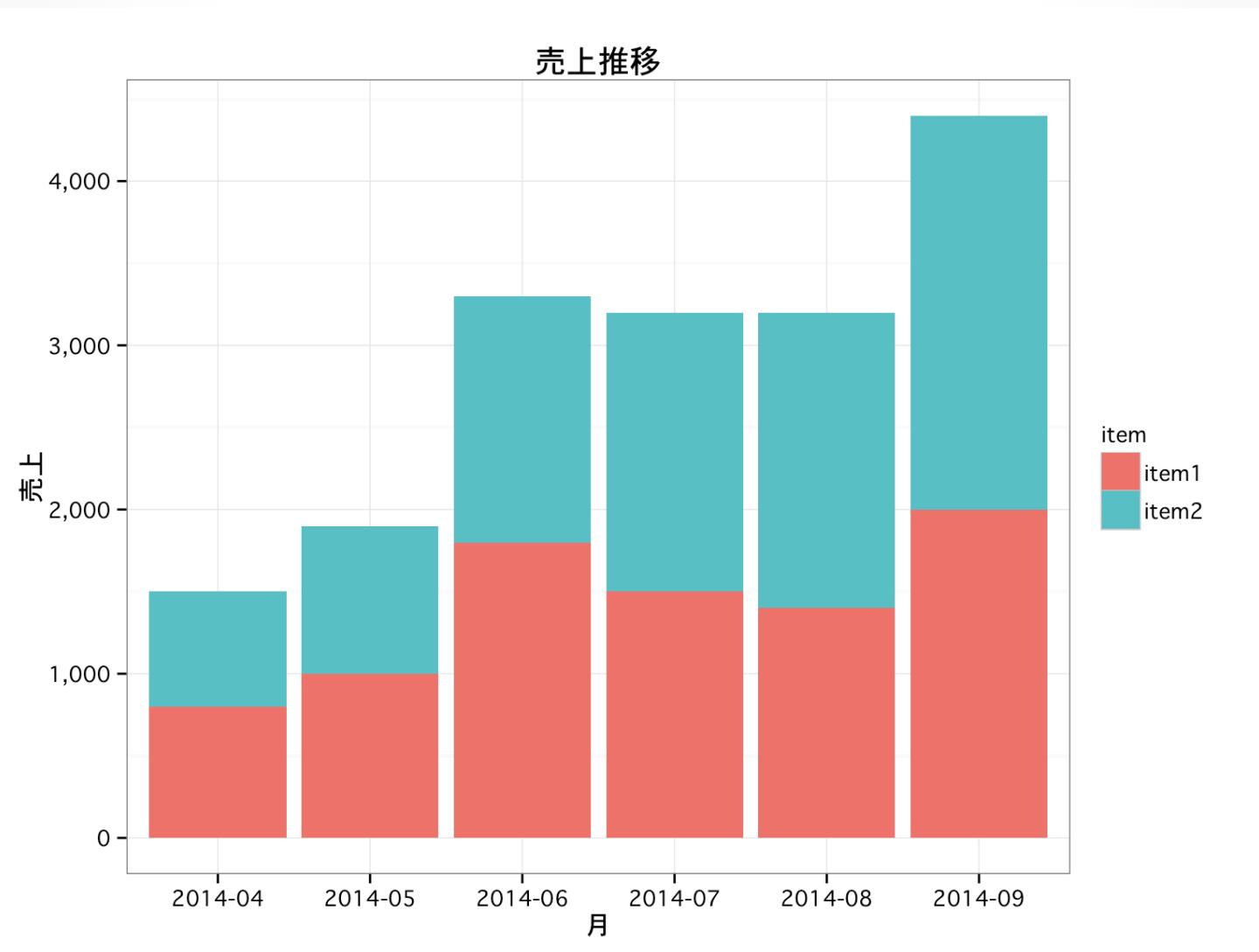
```
## 数値のフォーマットを3ヶタ区切りにする
ggplot(sales.data1, aes(x=month, y=sales)) +
  geom_bar(stat="identity") +
  geom_text(aes(label=comma(sales))), vjust=-0.2) +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

追加的ストの加工（結果）

```
## 数値のフォーマットを3ヶタ区切りにする
ggplot(sales.data1, aes(x=month, y=sales)) +
  geom_bar(stat="identity") +
  geom_text(aes(label=comma(sales)), vjust=-0.2) +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```



積み上げ棒グラフを作りたい



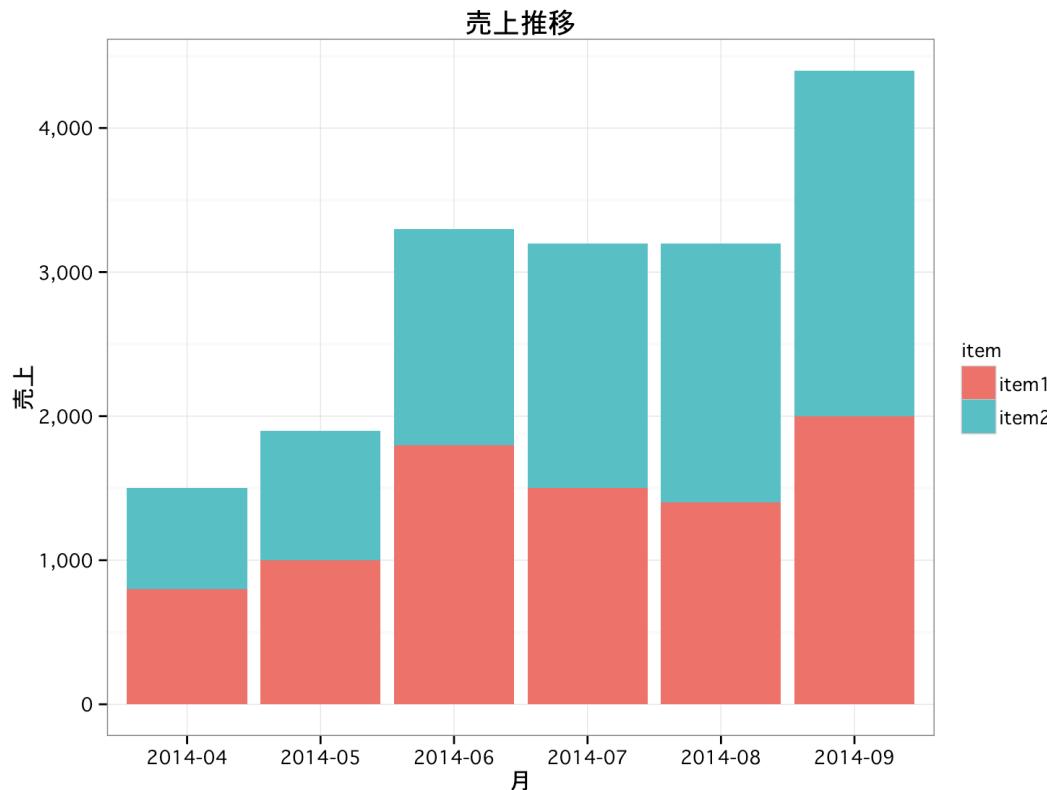
積み上げ棒グラフの作成（Rコード）

fillオプションにグループとなる変数を指定する

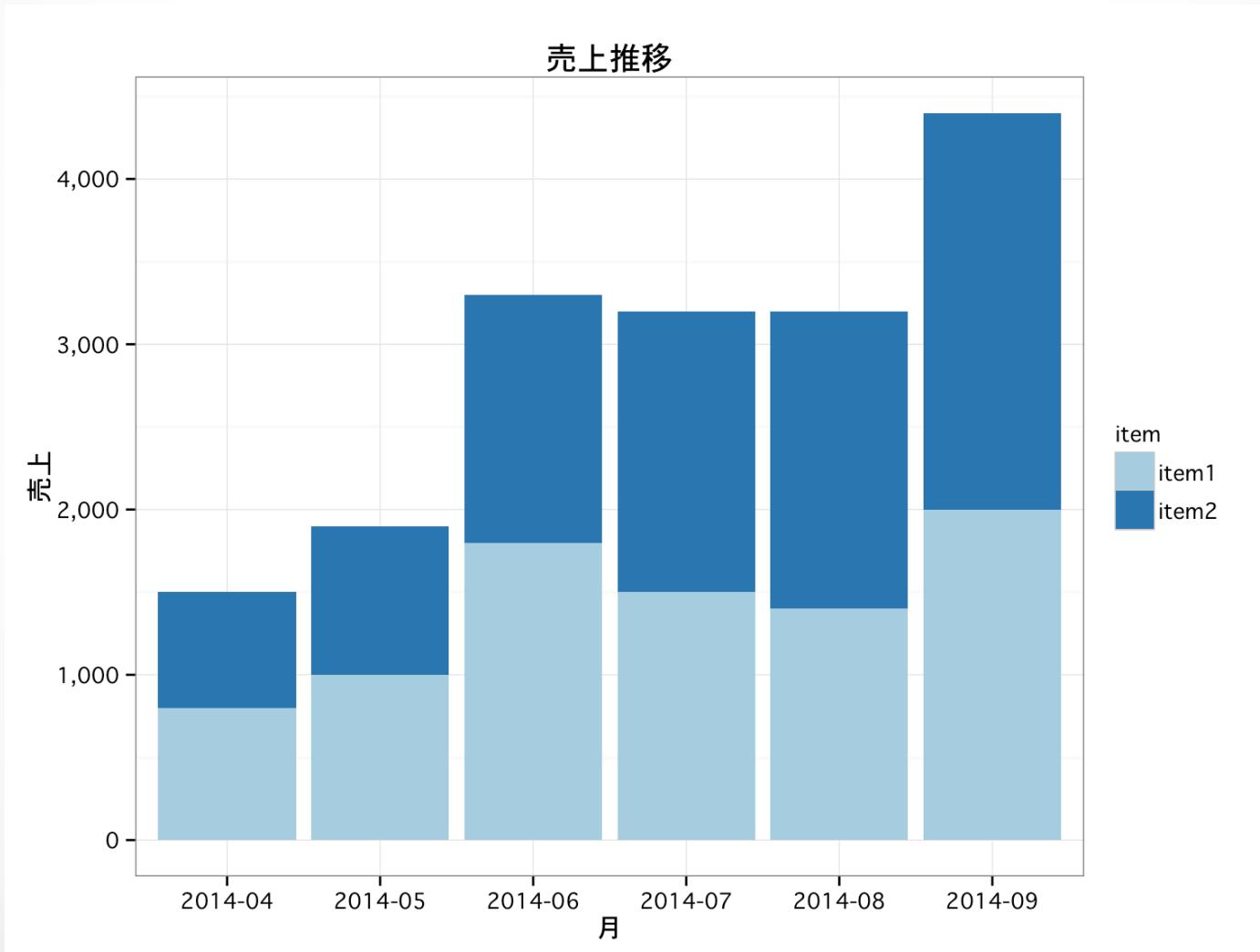
```
# 積み上げ棒グラフ
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +
  geom_bar(stat="identity") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

積み上げ棒グラフの作成（結果）

```
# 積み上げ棒グラフ  
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +  
  geom_bar(stat="identity") +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



棒の色セットを変更したい



色セットの変更（パレットの確認）

RColorBrewerパッケージ

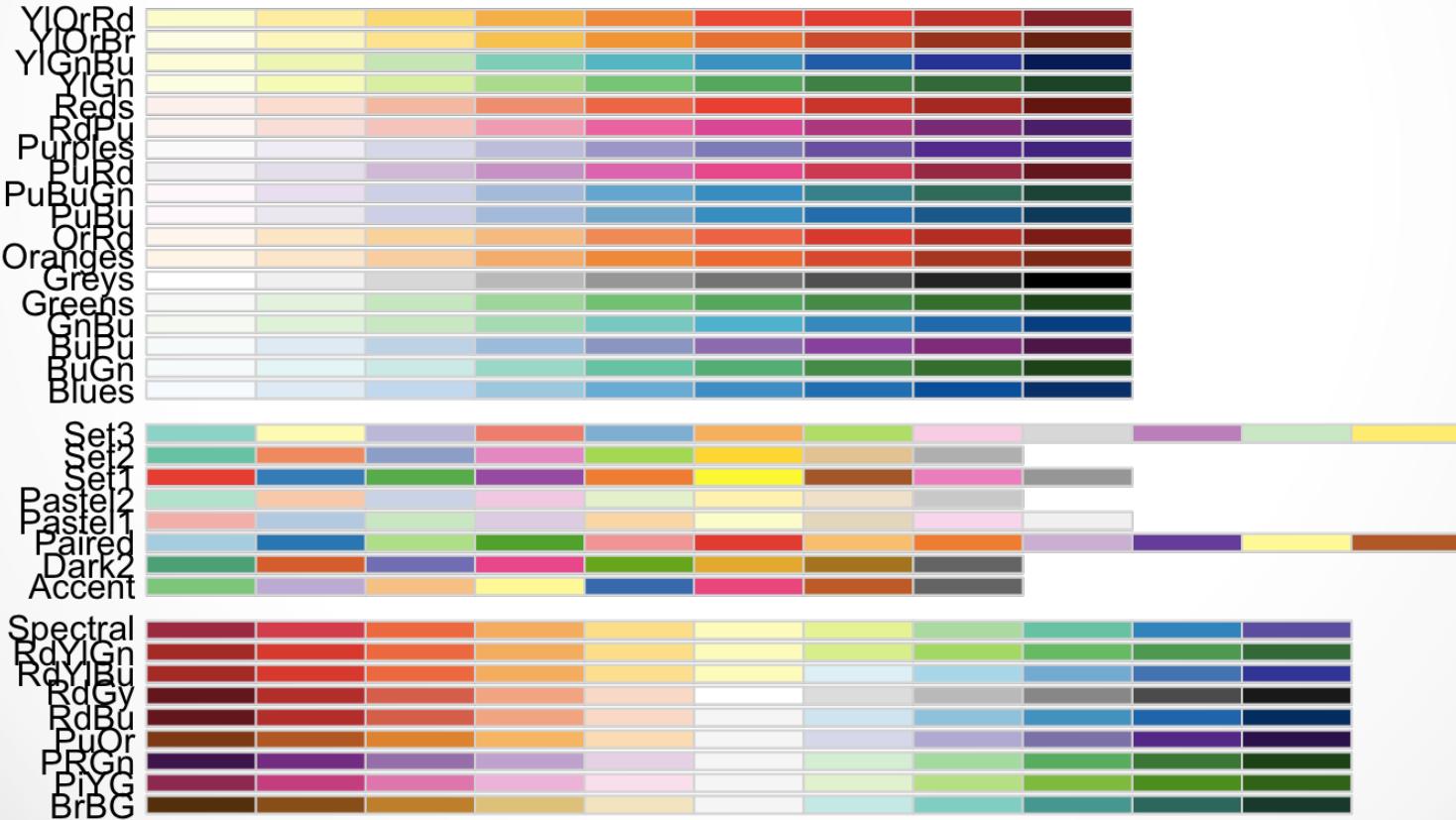
```
library(RColorBrewer)  
rownames(brewer.pal.info)
```

```
## [1] "BrBG"   "PiYG"   "PRGn"   "PuOr"   "RdBu"   "RdGy"  
## [7] "RdYlBu"  "RdYlGn"  "Spectral" "Accent"  "Dark2"  "Paired"  
## [13] "Pastel1"  "Pastel2"  "Set1"    "Set2"    "Set3"   "Blues"  
## [19] "BuGn"    "BuPu"    "GnBu"    "Greens"  "Greys"  "Oranges"  
## [25] "OrRd"    "PuBu"    "PuBuGn"  "PuRd"   "Purples" "RdPu"  
## [31] "Reds"    "YlGn"    "YlGnBu"  "YlOrBr" "YlOrRd"
```

色セットの変更（パレットの確認）

display.brewer.all関数でパレットを確認

```
display.brewer.all()
```



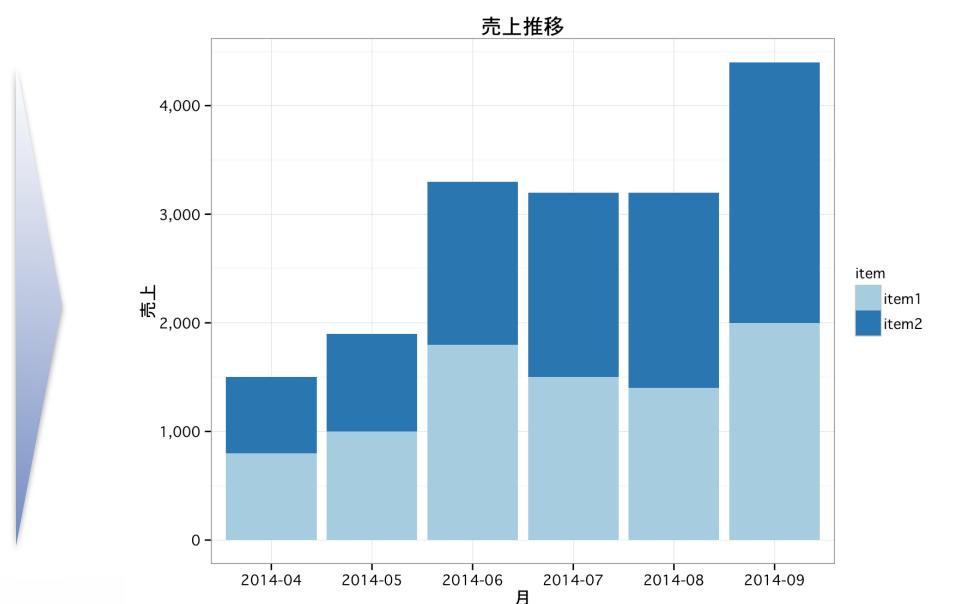
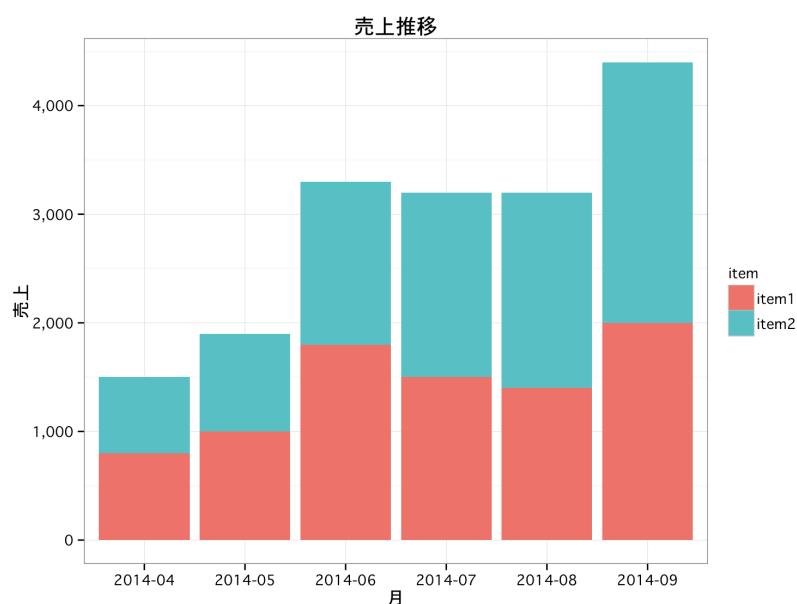
色セットの変更 (Rコード)

scale_fill_brewer関数を使う

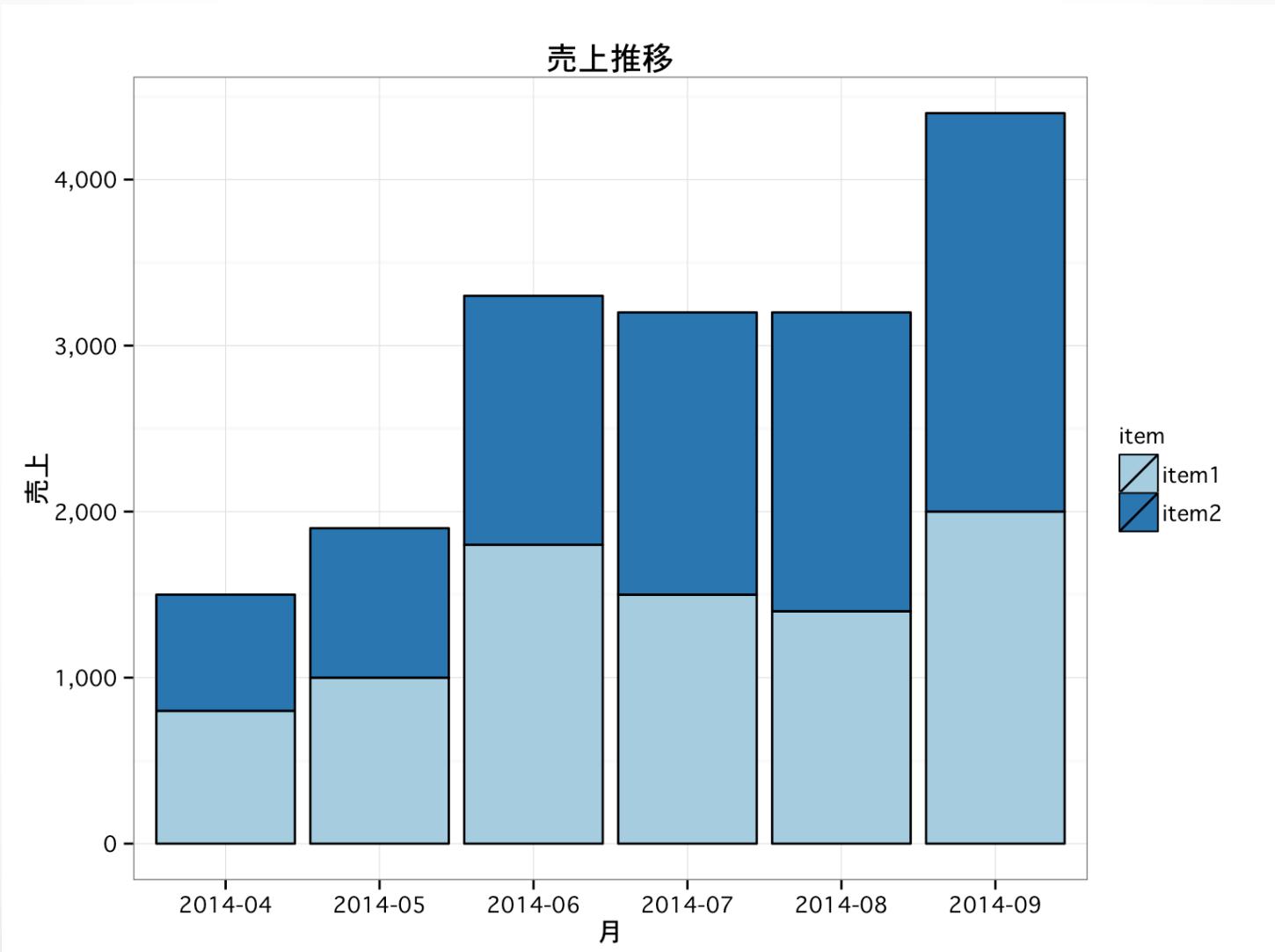
```
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +  
  geom_bar(stat="identity") +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  scale_y_continuous(label=comma) +  
  scale_fill_brewer(palette = "Paired") +  
  theme_bw(base_family="Osaka")
```

色セットの変更（結果）

```
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +  
  geom_bar(stat="identity") +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  scale_y_continuous(label=comma) +  
  scale_fill_brewer(palette = "Paired") +  
  theme_bw(base_family="Osaka")
```



棒に枠をつけたい



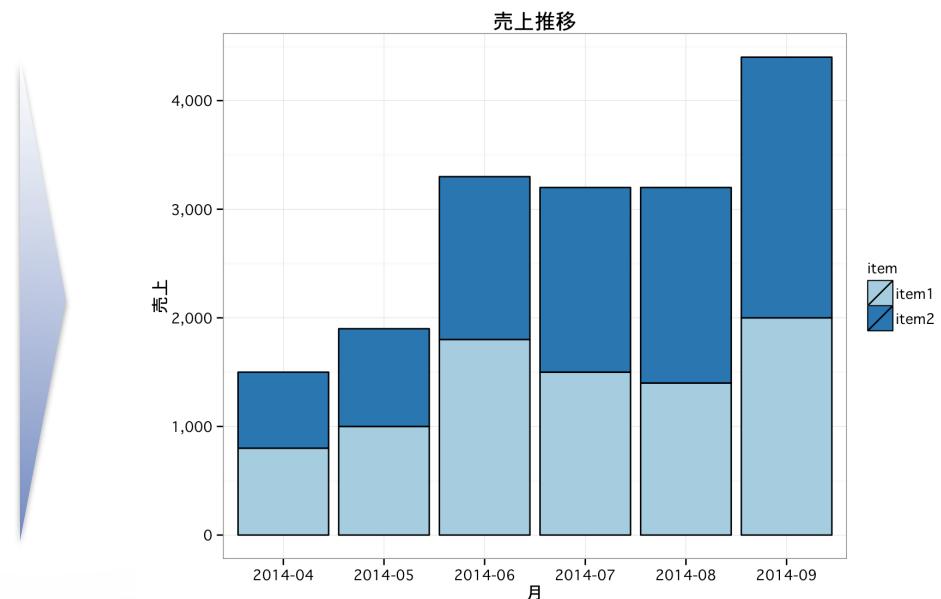
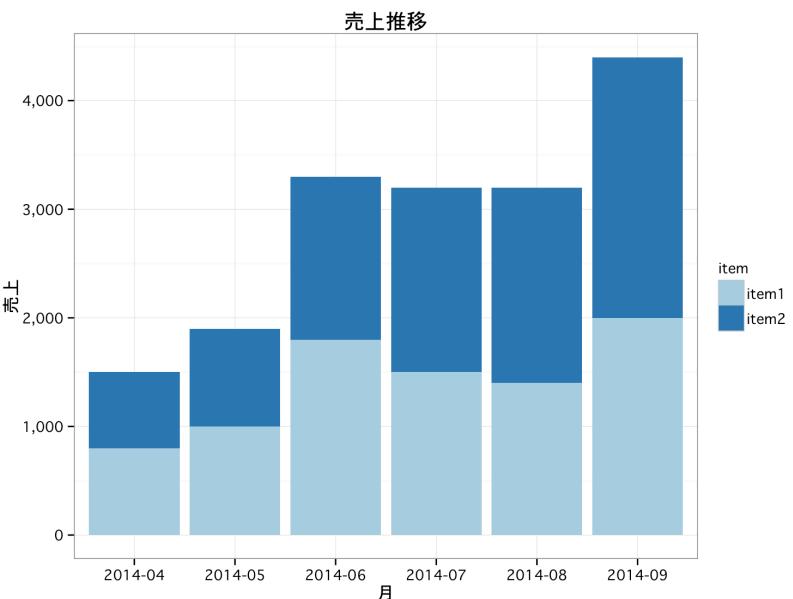
棒に枠をつける (Rコード)

colオプションに枠色を指定する

```
# 積み上げ棒グラフで枠を付ける
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +
  geom_bar(stat="identity", col="black") +  
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  scale_fill_brewer(palette = "Paired") +
  theme_bw(base_family="Osaka")
```

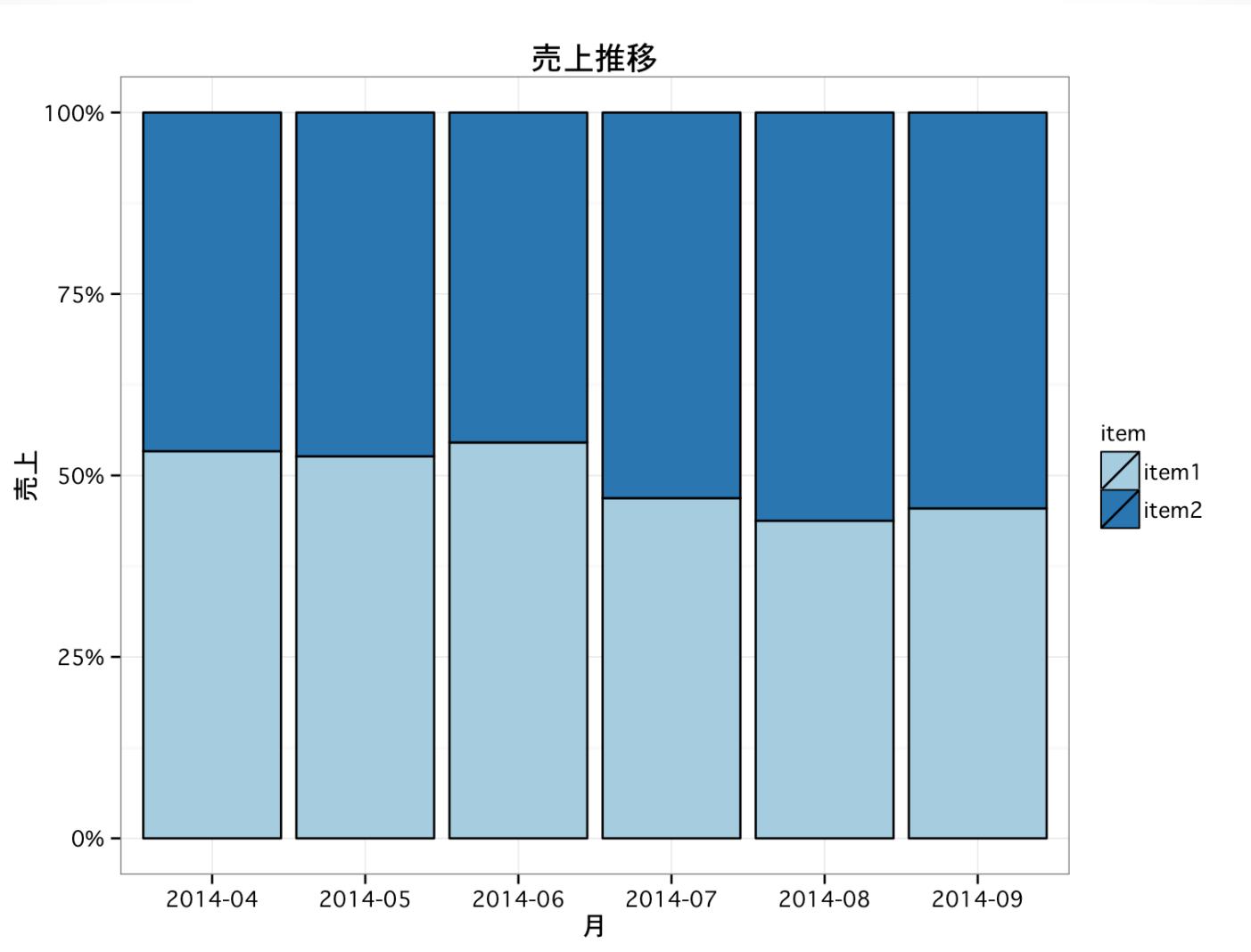
棒に枠をつける（結果）

```
# 積み上げ棒グラフで枠を付ける
ggplot(sales.data12, aes(x=month, v=sales, fill=item)) +
  geom_bar(stat="identity", col="black") + geom_bar(stat="identity", col="black") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  scale_fill_brewer(palette = "Paired") +
  theme_bw(base_family="Osaka")
```



100%積み上げ棒グラフの作成

100%積み上げ棒グラフを作りたい



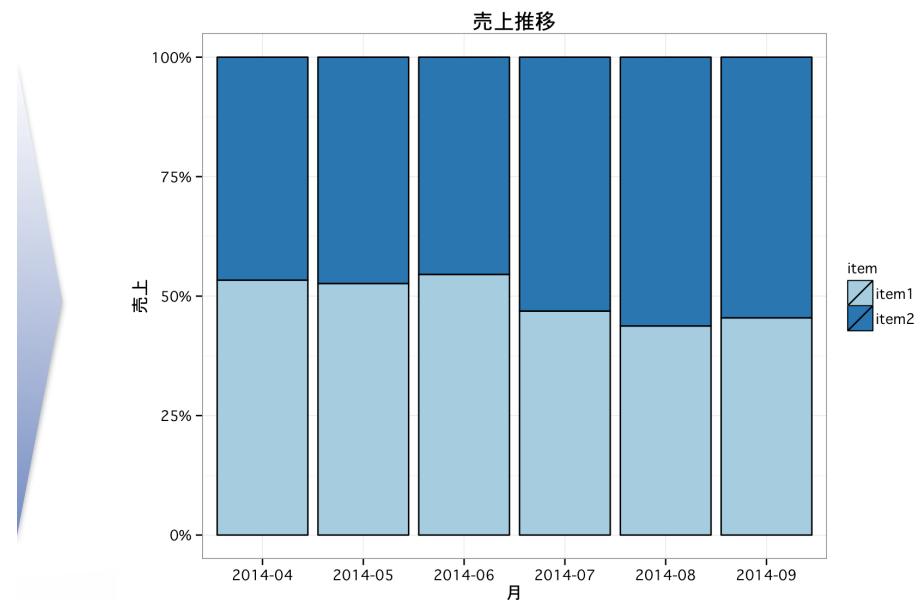
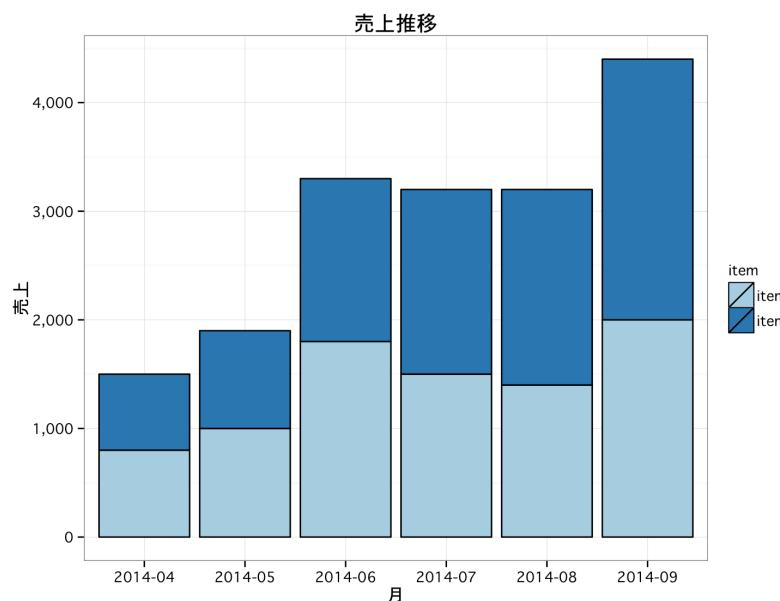
100%積み上げ棒グラフ（Rコード）

positionオプションにfillを指定する

```
# 100%積み上げグラフ
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +
  geom_bar(position="fill", stat="identity", col="black") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=percent) +
  scale_fill_brewer(palette = "Paired") +
  theme_bw(base_family="Osaka")
```

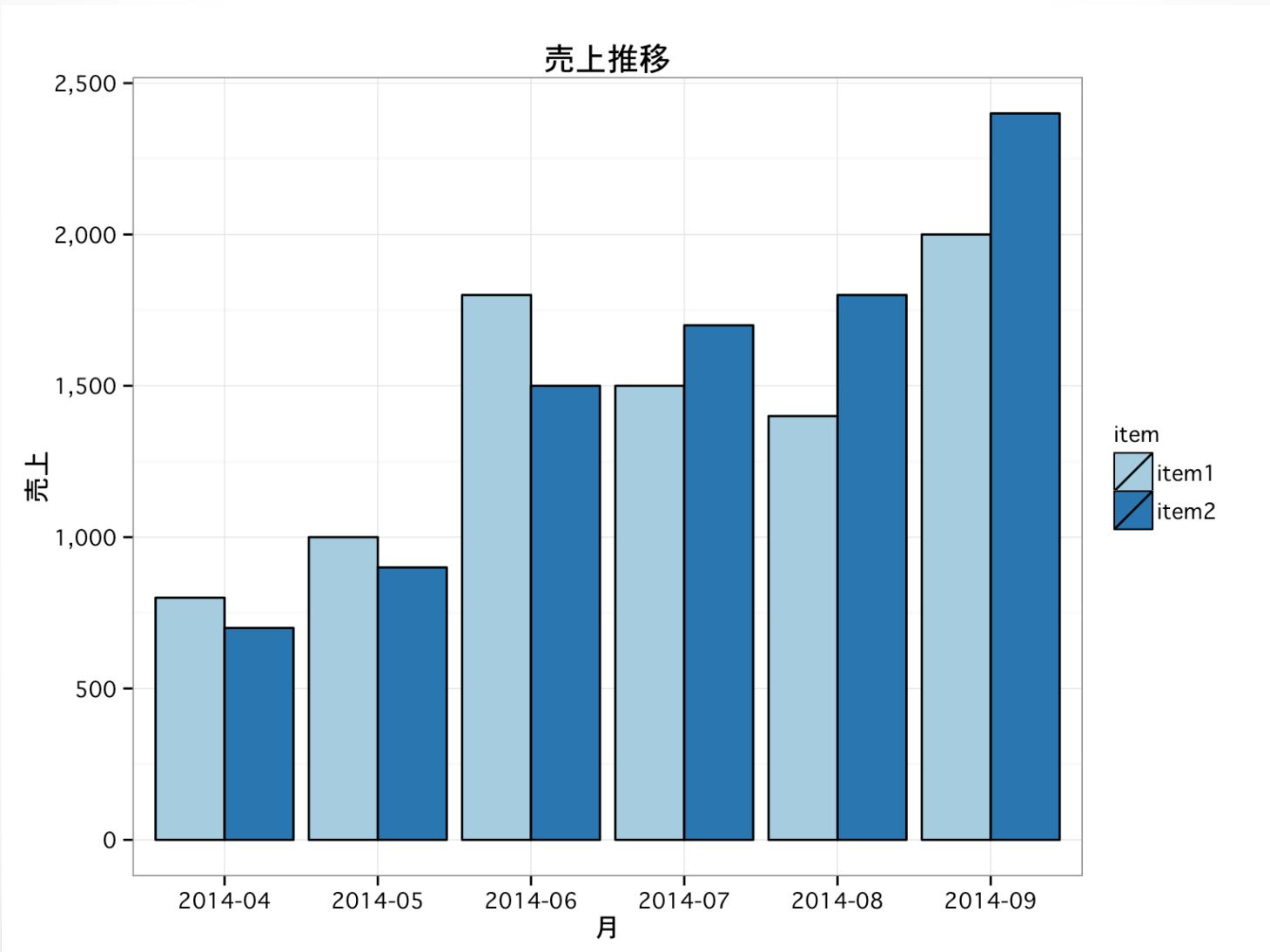
100%積み上げ棒グラフ（結果）

```
# 100%積み上げグラフ
ggplot(sales_data12, aes(x=month, y=sales, fill=item)) +
  geom_bar(position="fill", stat="identity", col="black") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=percent) +
  scale_fill_brewer(palette = "Paired") +
  theme_bw(base_family="Osaka")
```



棒を水平に並べる

棒を水平に並べたい



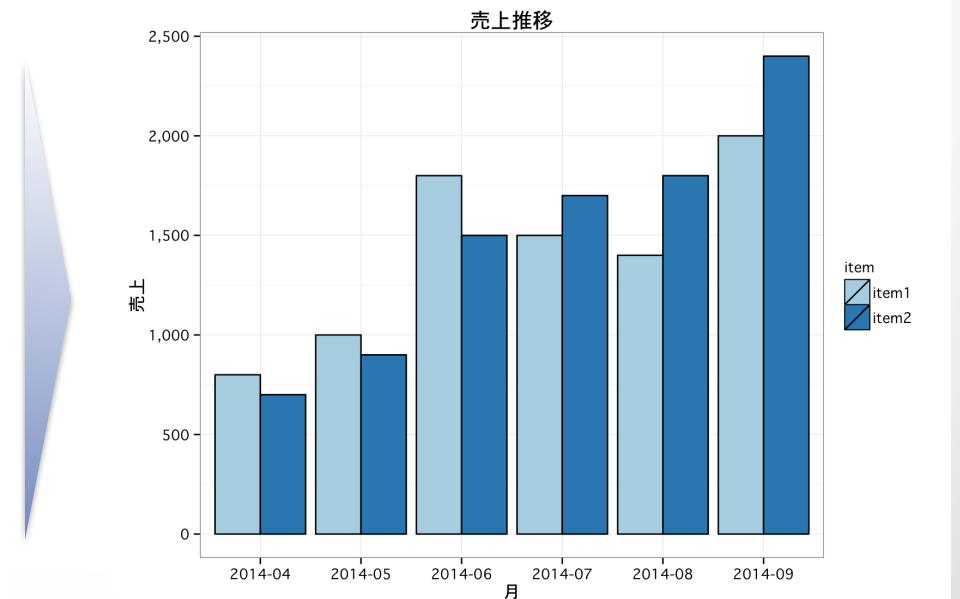
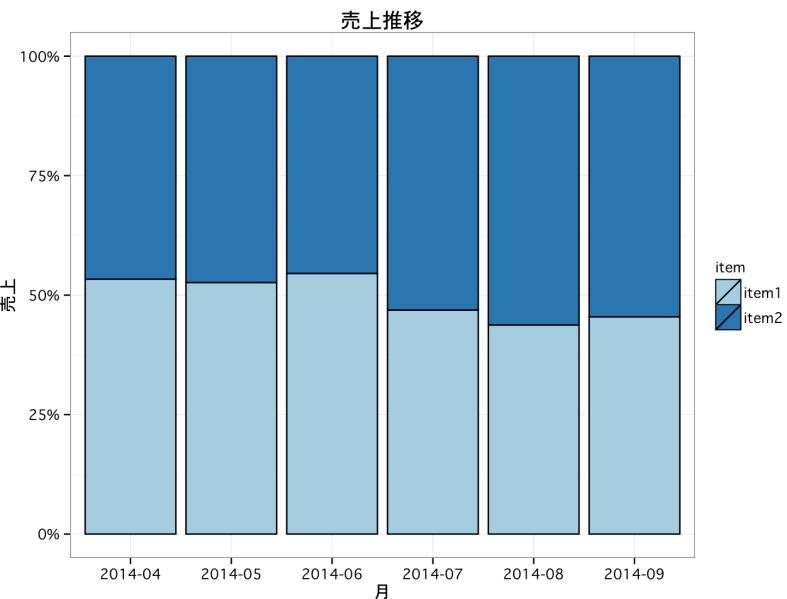
棒を水平に並べる (Rコード)

positionオプションにdodgeを指定する

```
## 水平に棒グラフを並べる
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +
  geom_bar(position="dodge", stat="identity", col="black") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  scale_fill_brewer(palette = "Paired") +
  theme_bw(base_family="Osaka")
```

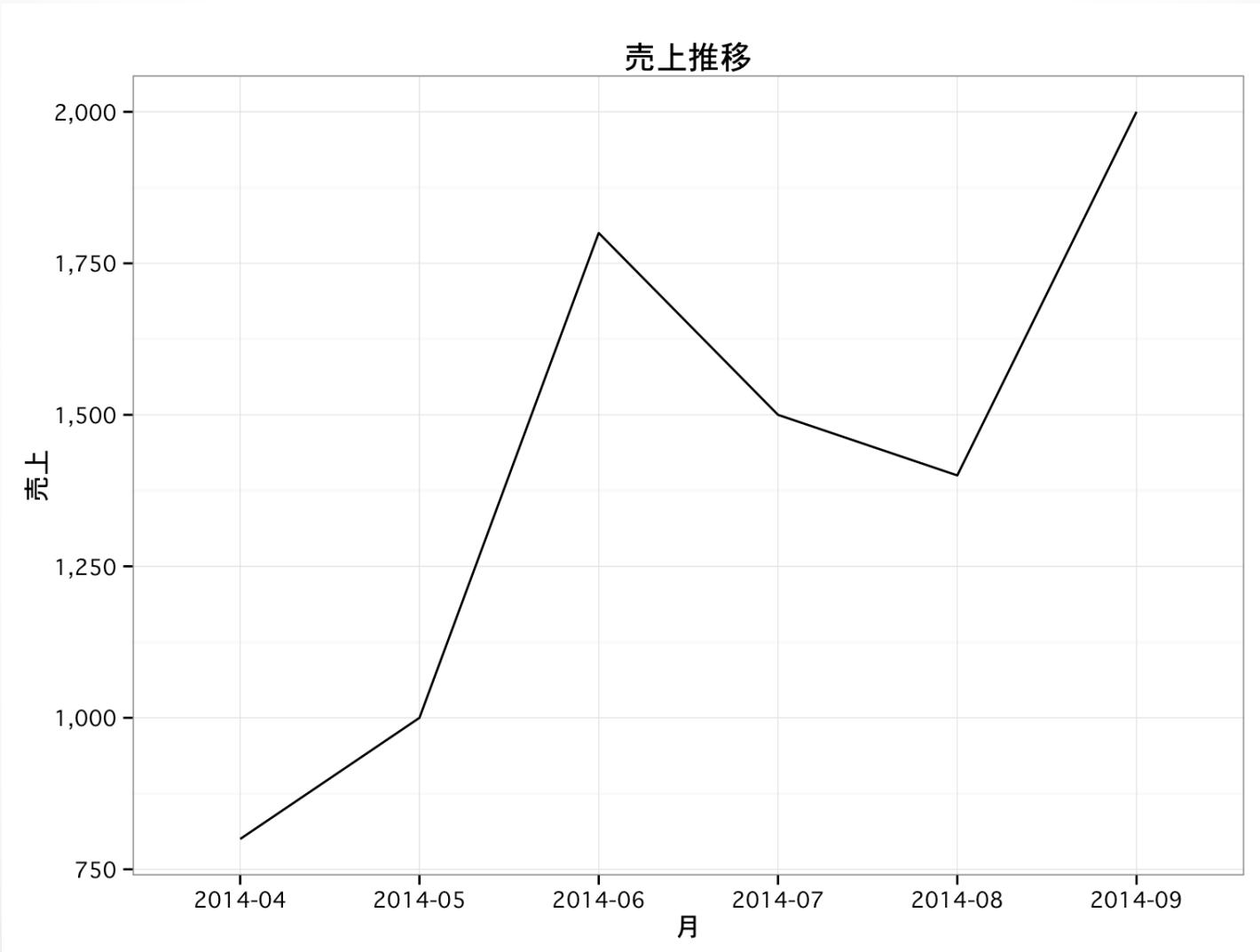
棒を水平に並べる（結果）

```
## 水平に棒グラフを並べる
ggplot(sales.data12, aes(x=month, y=sales, fill=item)) +
  geom_bar(position="dodge", stat="identity", col="black") +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma) +
  scale_fill_brewer(palette = "Paired") +
  theme_bw(base_family="Osaka")
```



- 自己紹介
 - ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図
-
-

折れ線グラフを作りたい

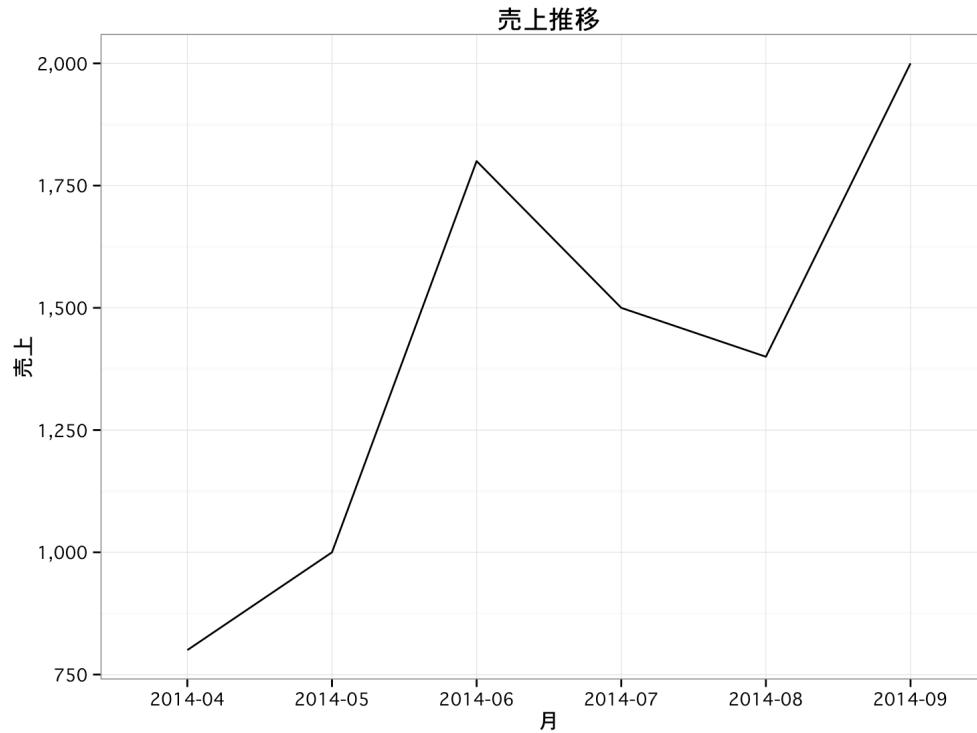


geom_line関数を使う

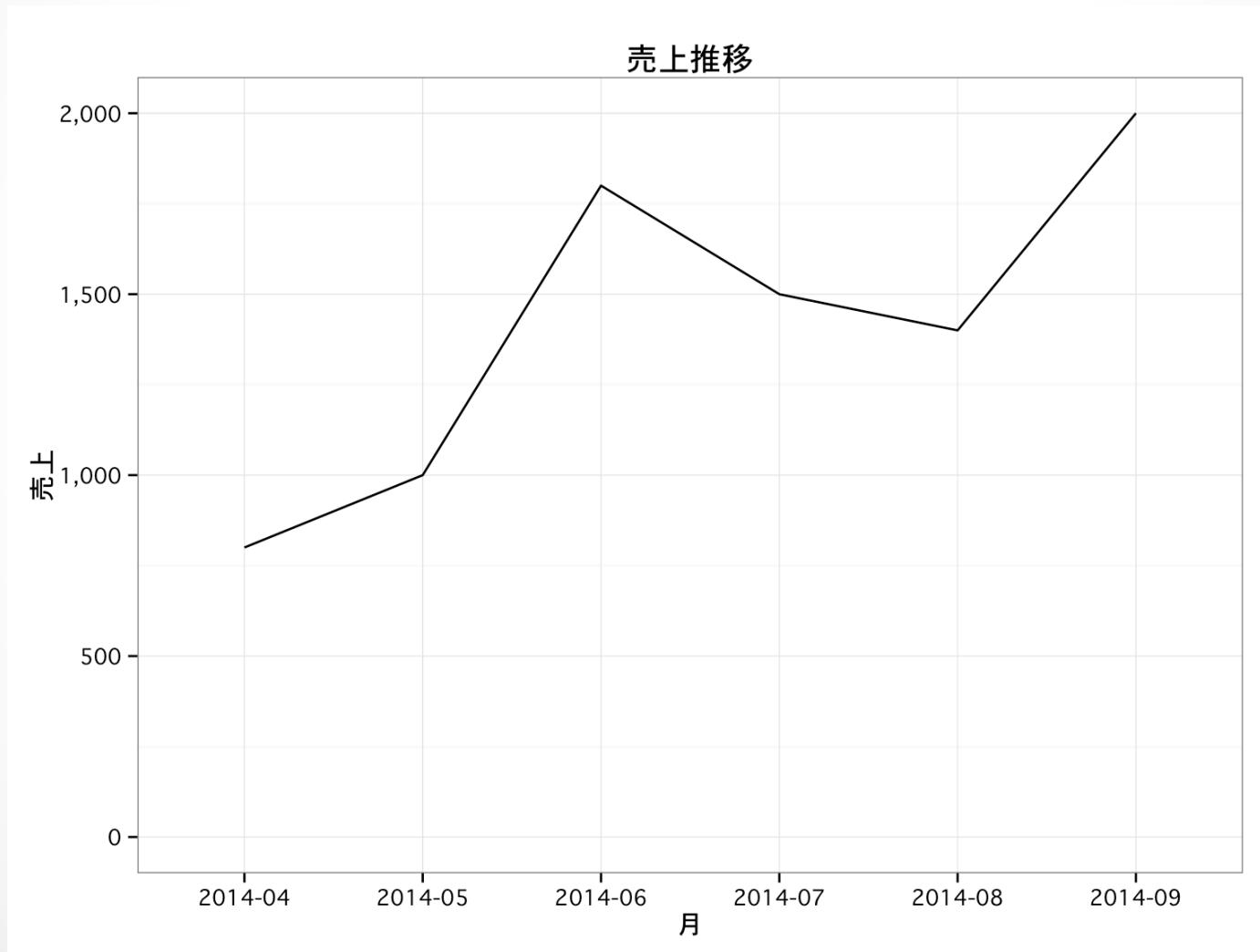
```
# xがファクター  
ggplot(sales.data1, aes(x=month, y=sales,group=1)) +  
  geom_line() +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```

折れ線グラフの作成（結果）

```
# xがファクター  
ggplot(sales_data1, aes(x=month, y=sales,group=1)) +  
  geom_line() +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



y軸の範囲を指定したい

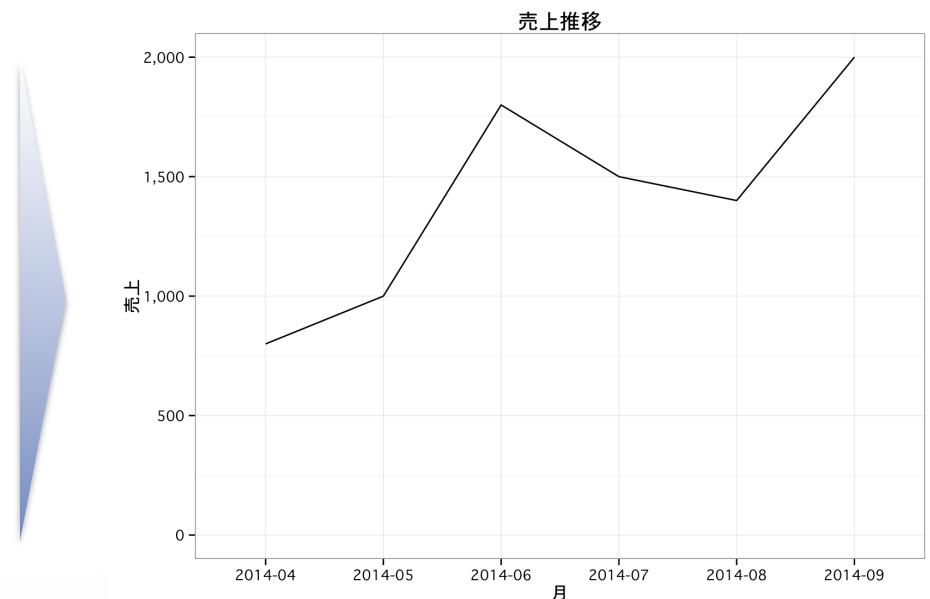
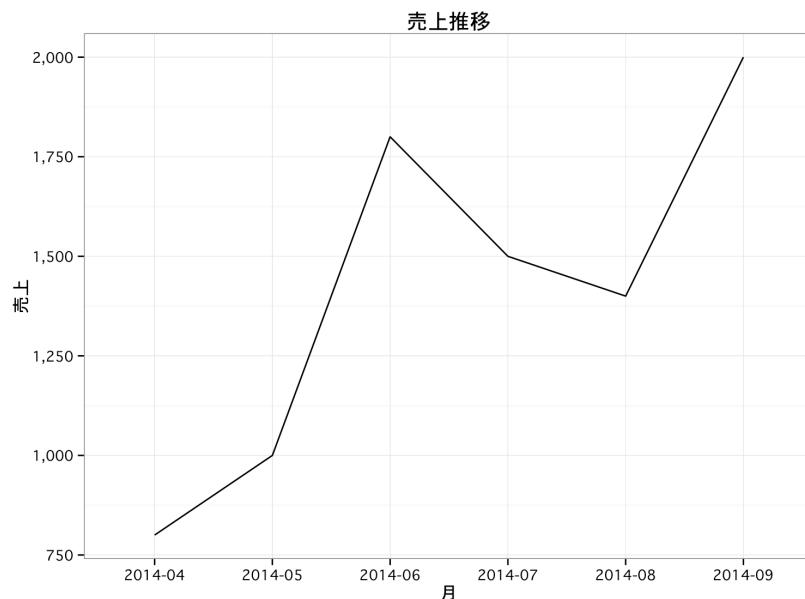


scale_y_continuous関数の limitsオプションを使う

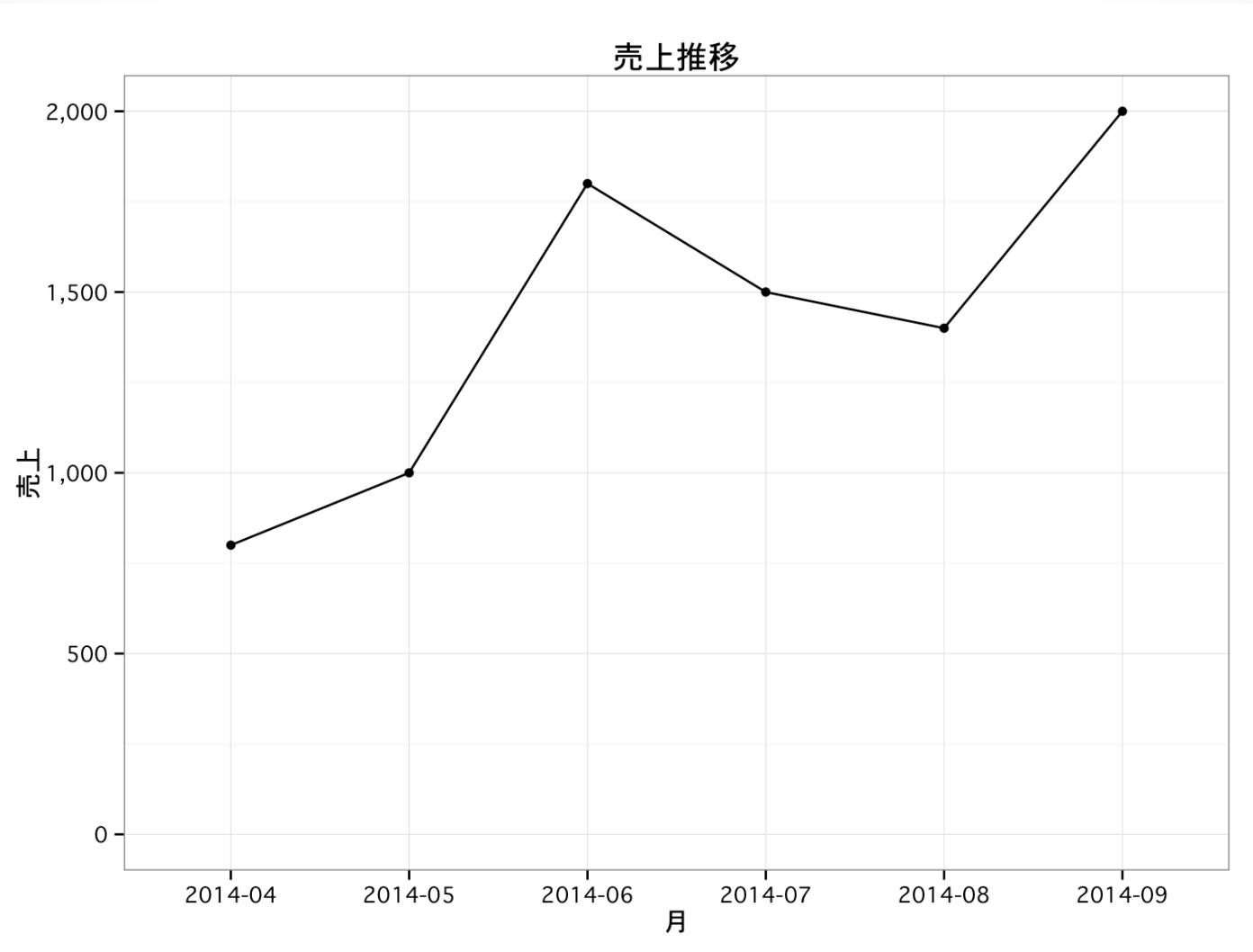
```
ylimit <- c(0,max(sales.data1$sales))
ggplot(sales.data1, aes(x=month, y=sales,group=1)) +
  geom_line() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma,limits=ylimit) +
  theme_bw(base_family="Osaka")
```

y軸の範囲指定（結果）

```
ylim <- c(0,max(sales.data1$sales))
ggplot(sales.data1, aes(x=month, y=sales,group=1)) +
  geom_line() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma,limits=ylim) +
  theme_bw(base_family="Osaka")
```



折れ線グラフに点を追加したい

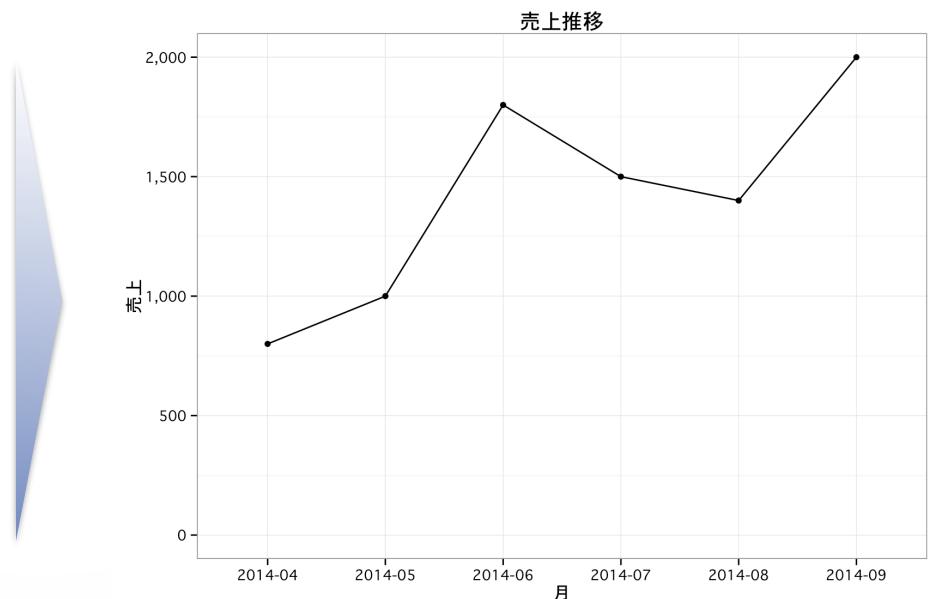
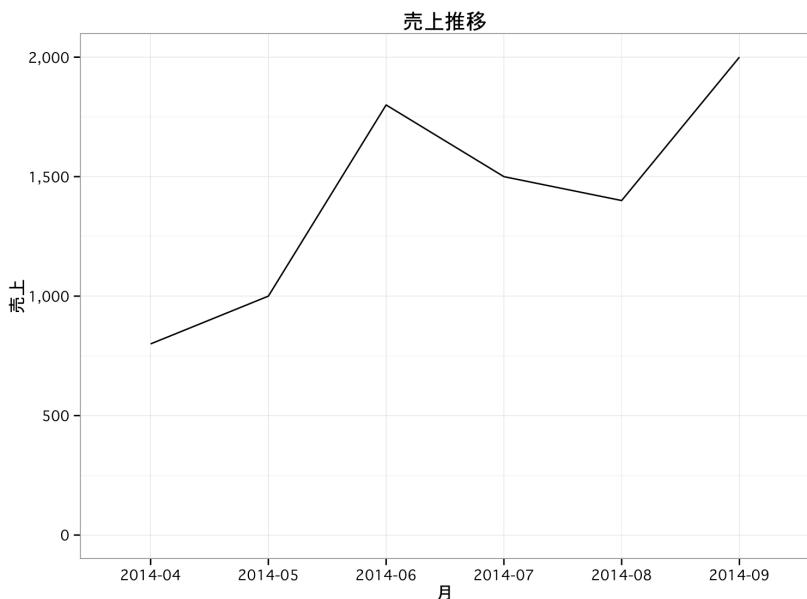


geom_point関数を追加する

```
# 点を追加
ggplot(sales.data1, aes(x=month, y=sales,group=1)) +
  geom_line() +
  geom_point() + geom_point() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma,limits=ylim) +
  theme_bw(base_family="Osaka")
```

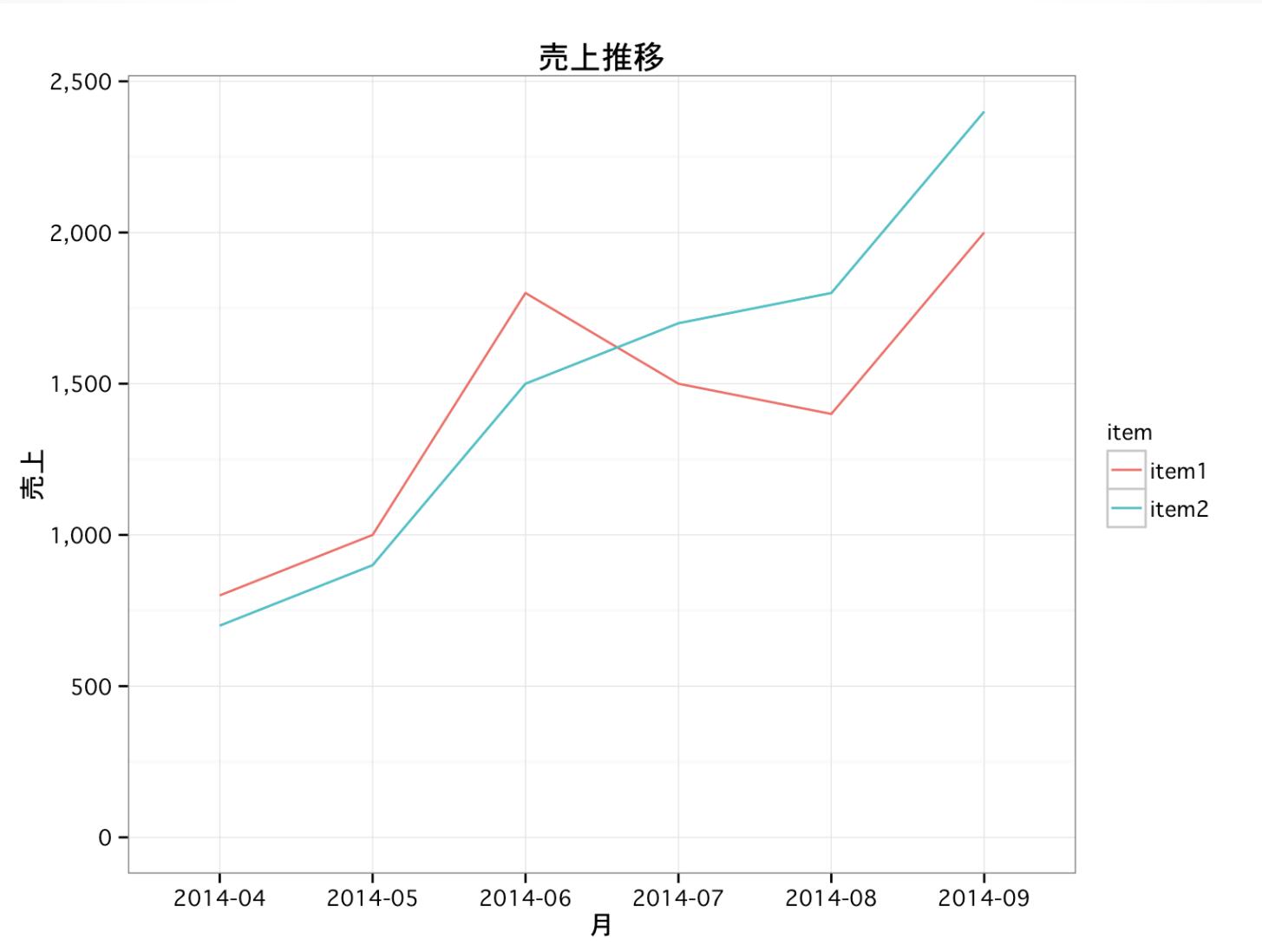
点の追加（結果）

```
# 点を追加
ggplot(sales.data1, aes(x=month, y=sales,group=1)) +
  geom_line() +
  geom_point() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma,limits=ylim) +
  theme_bw(base_family="Osaka")
```



折れ線グラフのグループ化

グループごとの折れ線グラフを作りたい



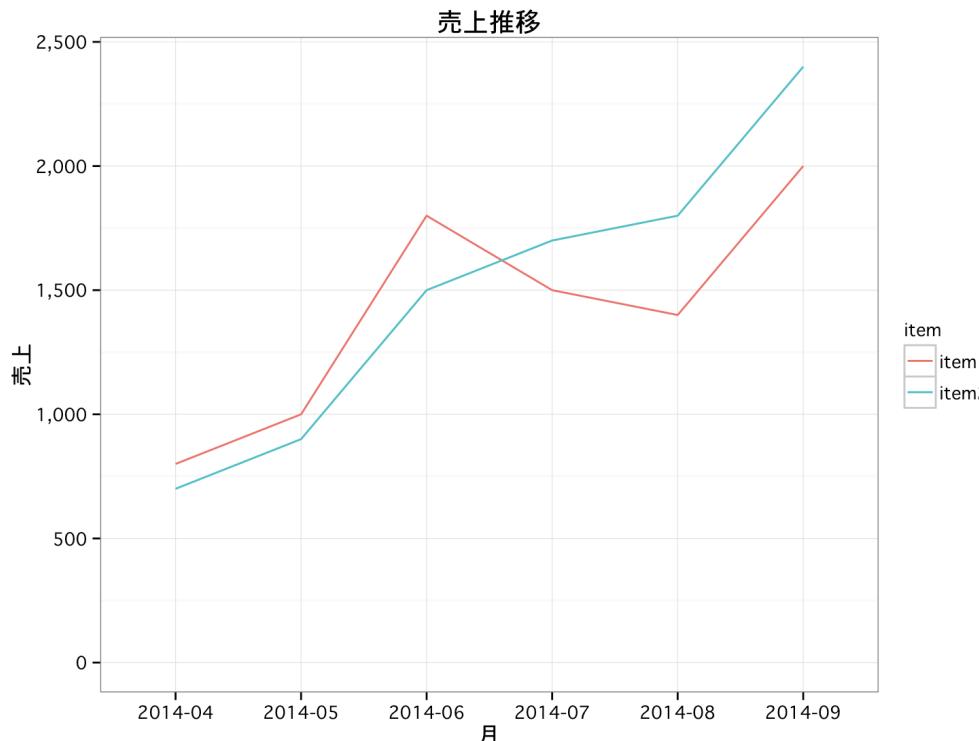
折れ線グラフのグループ化（Rコード）

groupとcolオプションに
グループとなる変数を指定する

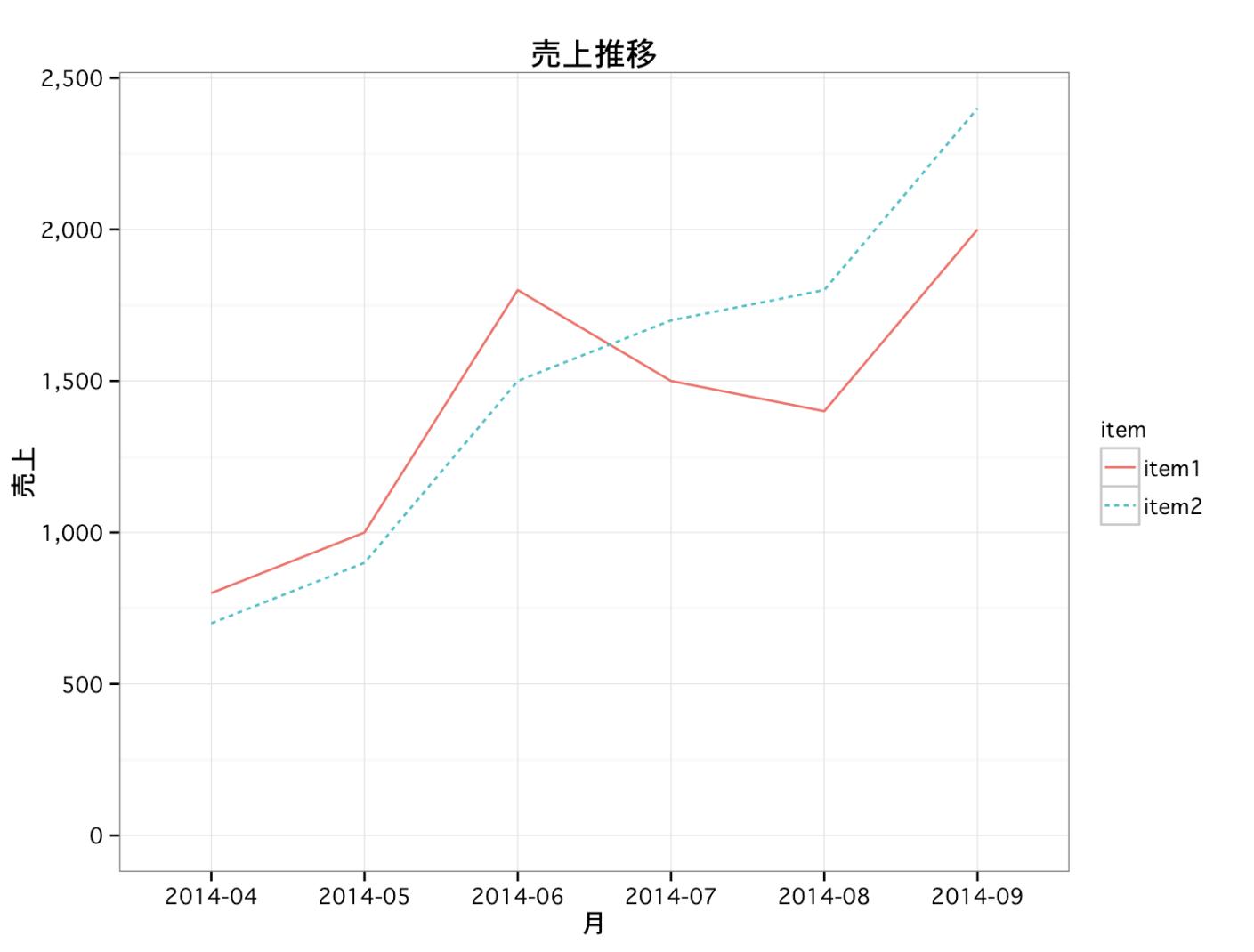
```
# グループごとに折れ線グラフ
ylimit <- c(0,max(sales.data12$sales))
ggplot(sales.data12, aes(x=month, y=sales, group=item,col=item)) +
  geom_line() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma,limits=ylimit) +
  theme_bw(base_family="Osaka")
```

折れ線グラフのグループ化（結果）

```
# グループごとに折れ線グラフ
ylimit <- c(0,max(sales.data12$sales))
ggplot(sales.data12, aes(x=month, y=sales, group=item, col=item)) +
  geom_line() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma,limits=ylimit) +
  theme_bw(base_family="Osaka")
```



グループごとに線の種類を変えたい



線の種類の変更 (Rコード)

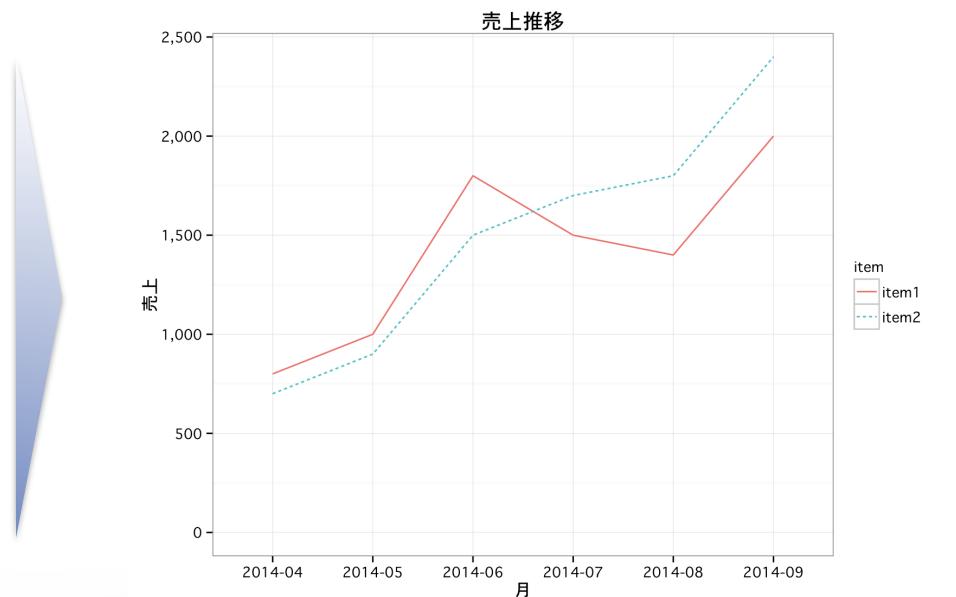
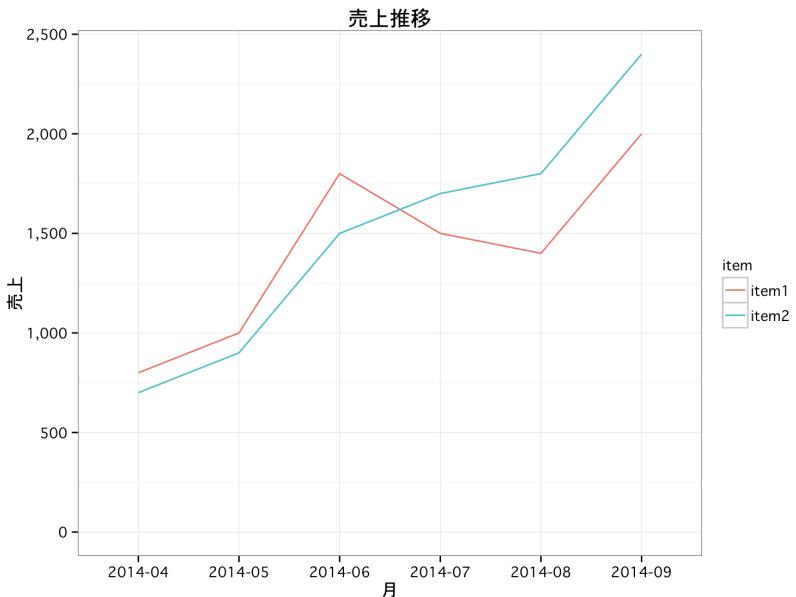
ltyオプションにグループとなる変数を指定

```
# グループごとに線の種類を変える
ggplot(sales.data12, aes(x=month, y=sales, group=item, col=item, lty=item)) +
  geom_line() +
  xlab("月") +
  ylab("売上") +
  ggtitle("売上推移") +
  scale_y_continuous(label=comma, limits=ylimit) +
  theme_bw(base_family="Osaka")
```

線の種類の変更（結果）

#グループごとに線の種類を変える

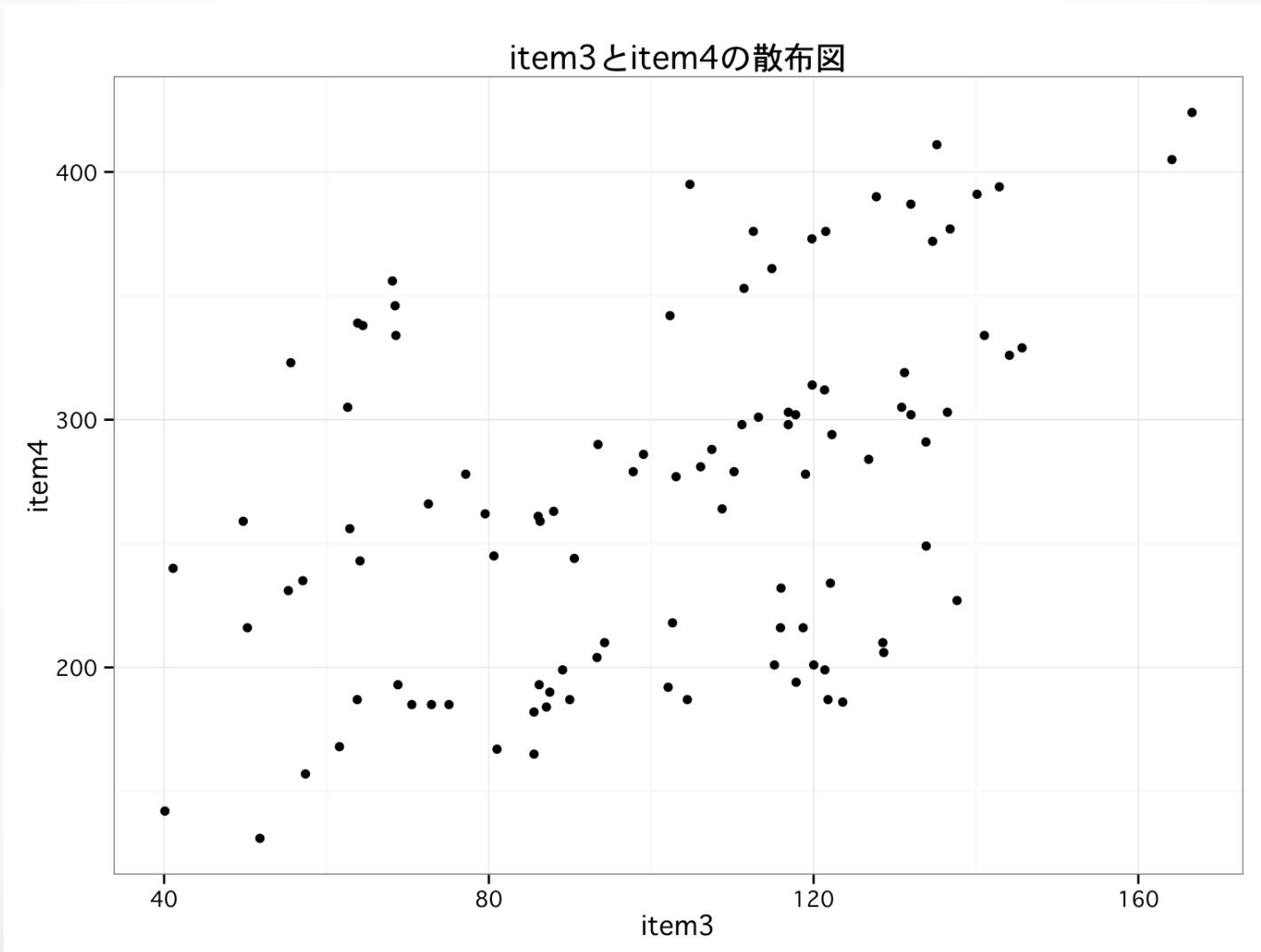
```
ggplot(sales.data12, aes(x=month, y=sales, group=item, col=item, lty=item)) +  
  geom_line() +  
  xlab("月") +  
  ylab("売上") +  
  ggtitle("売上推移") +  
  scale_y_continuous(label=comma, limits=ylimit) +  
  theme_bw(base_family="Osaka")
```



AGENDA

- 自己紹介
 - ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図
-
-

散布図を作りたい

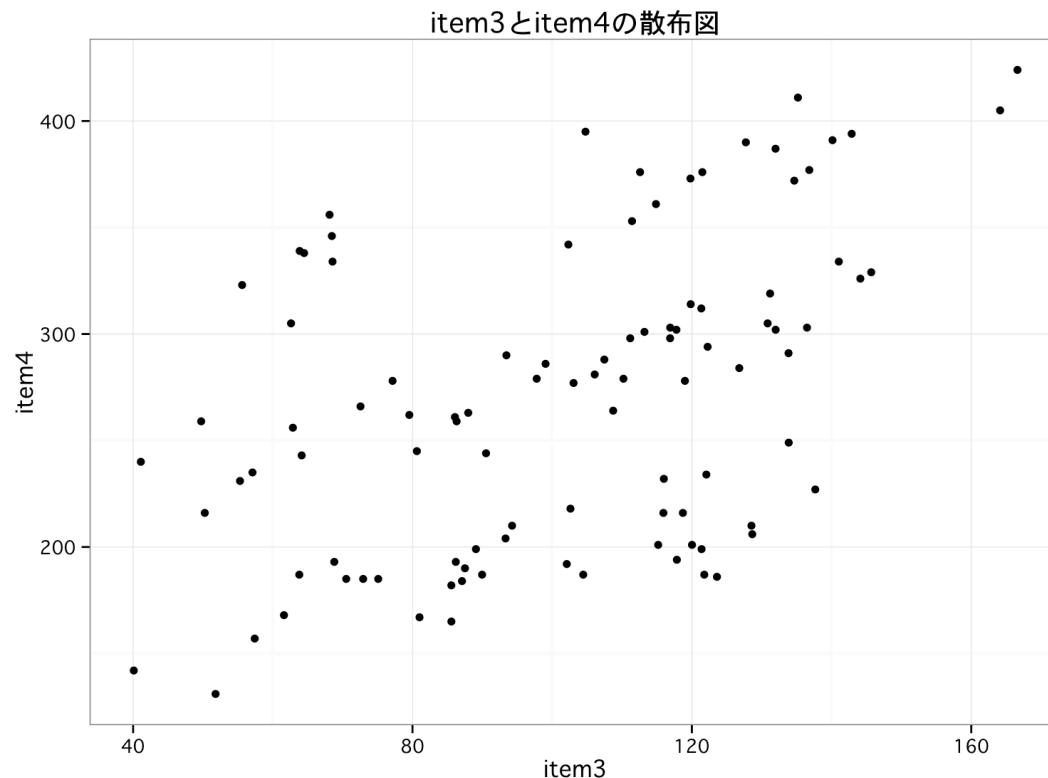


geom_point関数を使う

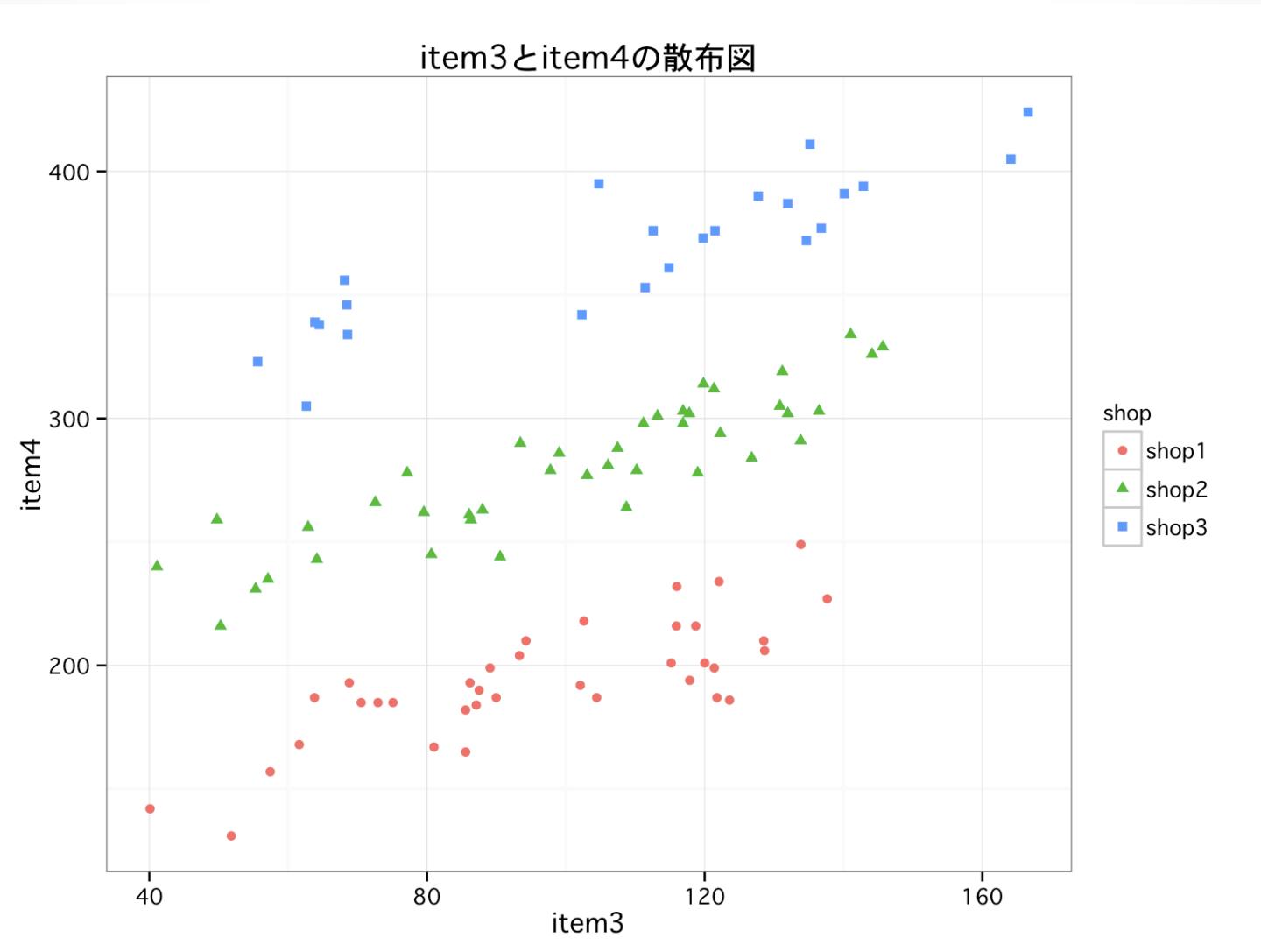
```
# 基本的な散布図
ggplot(sales.data34, aes(x=item3, y=item4)) +
  geom_point() +  
  xlab("item3") +  
  ylab("item4") +  
  ggtitle("item3とitem4の散布図") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```

散布図の作成（結果）

```
# 基本的な散布図  
ggplot(sales.data34, aes(x=item3, y=item4)) +  
  geom_point() +  
  xlab("item3") +  
  ylab("item4") +  
  ggtitle("item3とitem4の散布図") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



グループごとに色と形を変えたい



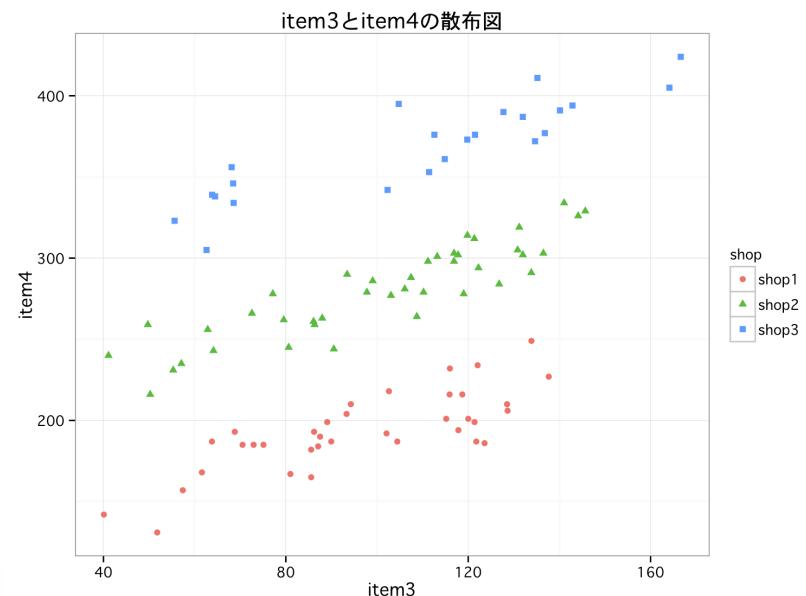
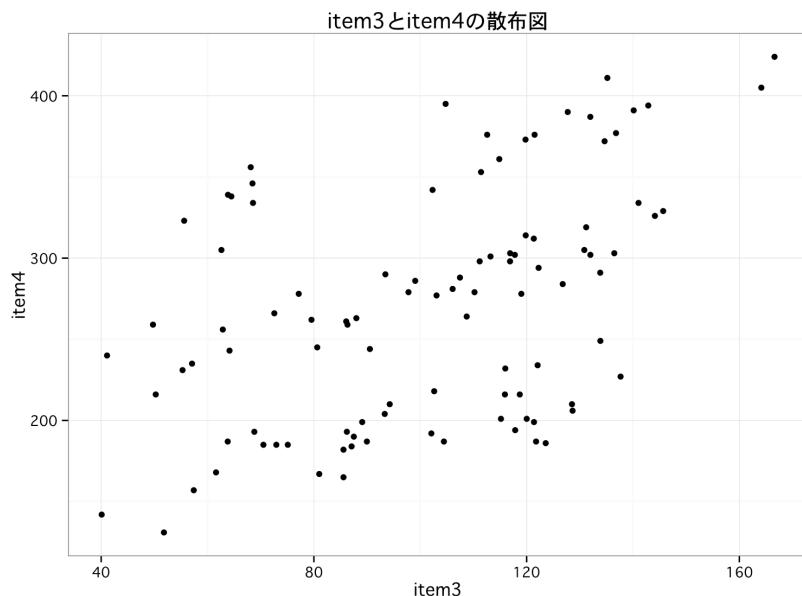
色と形の変更 (Rコード)

colとshapeオプションに
グループとなる変数を指定する

```
# グループごとに色と形を変える
ggplot(sales.data34, aes(x=item3, y=item4, col=shop, shape=shop)) +
  geom_point() +
  xlab("item3") +
  ylab("item4") +
  ggtitle("item3とitem4の散布図") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

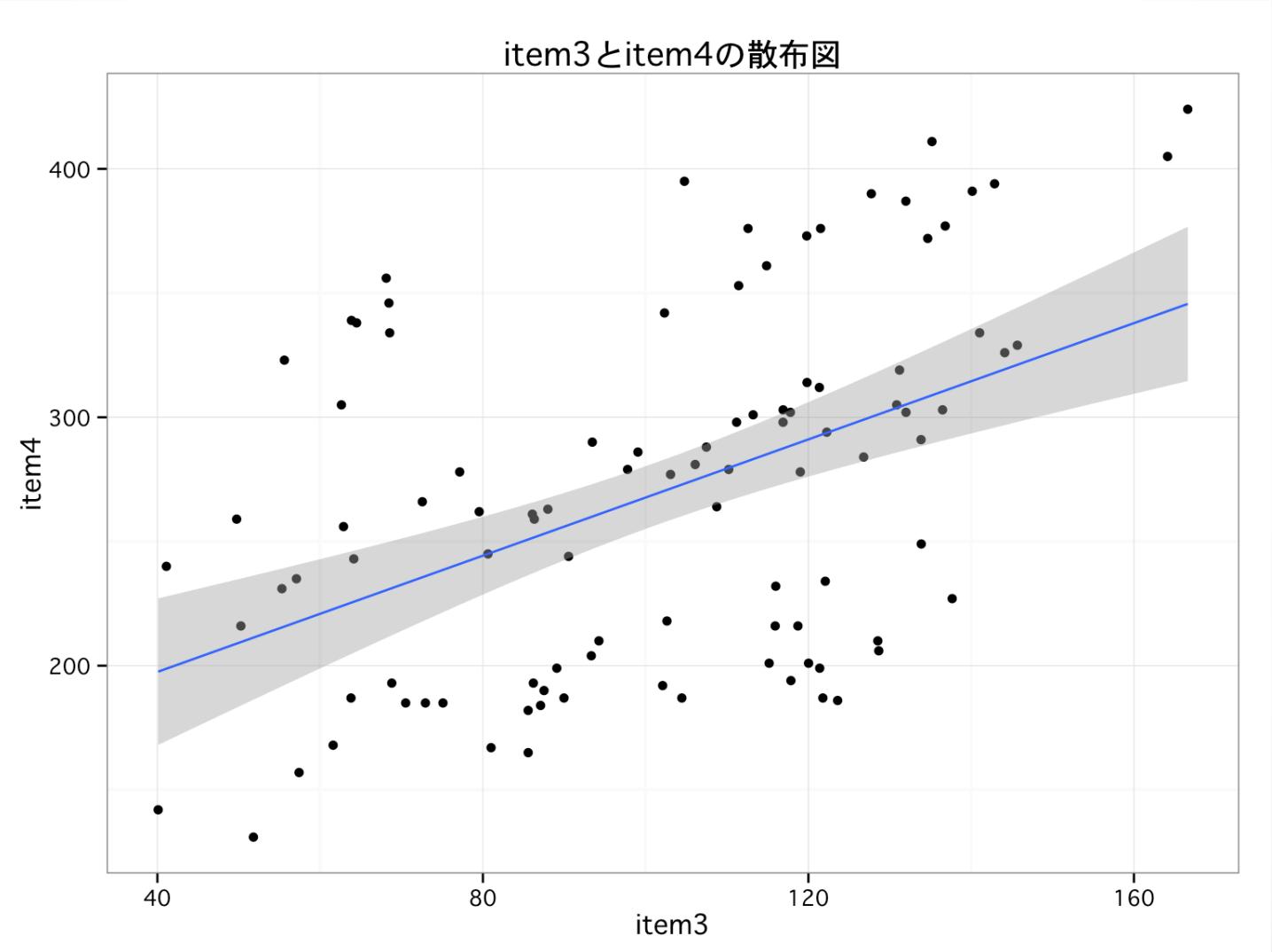
色と形の変更（結果）

```
# グループごとに色と形を変える
ggplot(sales.data34, aes(x=item3, y=item4, col=shop, shape=shop)) +
  geom_point() +
  xlab("item3") +
  ylab("item4") +
  ggtitle("item3とitem4の散布図") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```



回帰直線の追加

回帰直線を追加したい



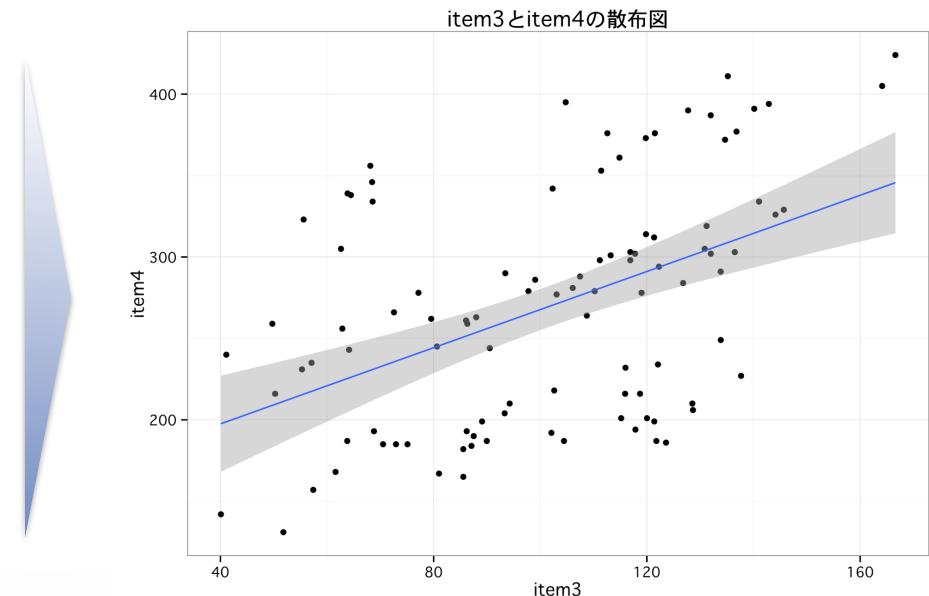
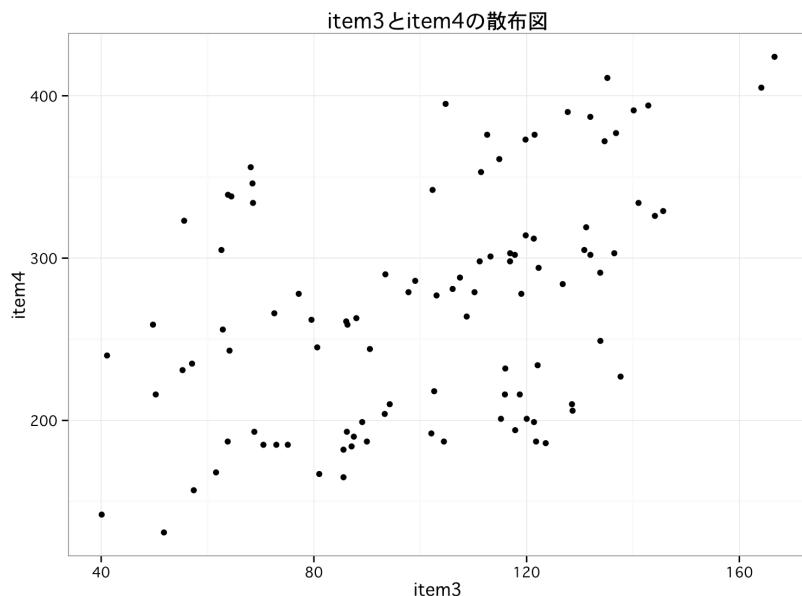
回帰直線の追加（Rコード）

geom_smooth関数を使う

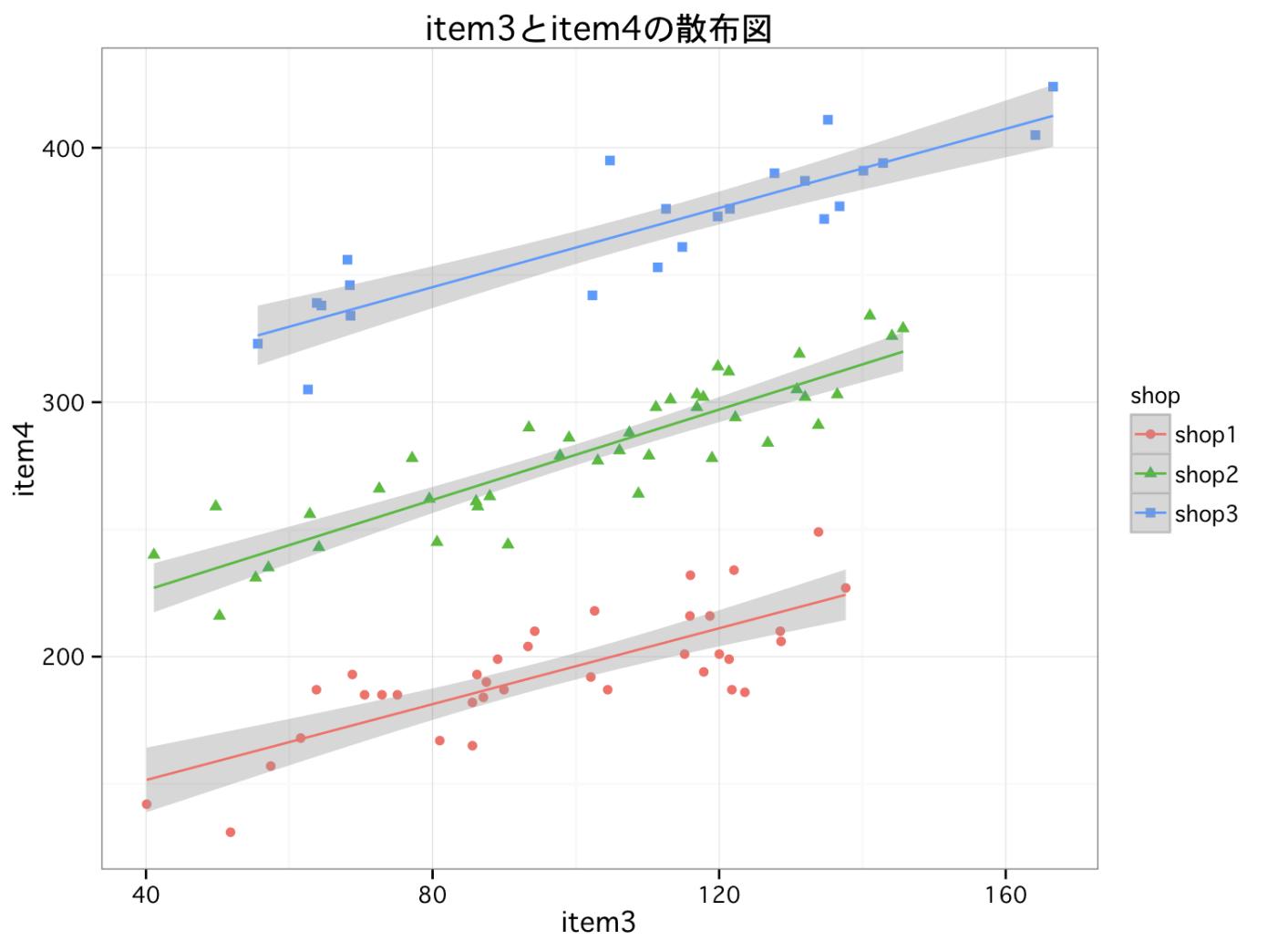
```
# 回帰直線の追加
ggplot(sales.data34, aes(x=item3, y=item4)) +
  geom_point() +
  geom_smooth(method="lm") +
  xlab("item3") +
  ylab("item4") +
  ggtitle("item3とitem4の散布図") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

回帰直線の追加（結果）

```
# 回帰直線の追加  
ggplot(sales.data34, aes(x=item3, y=item4)) +  
  geom_point() +  
  geom_smooth(method="lm") +  
  xlab("item3") +  
  ylab("item4") +  
  ggtitle("item3とitem4の散布図") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



グループごとに回帰直線を引きたい



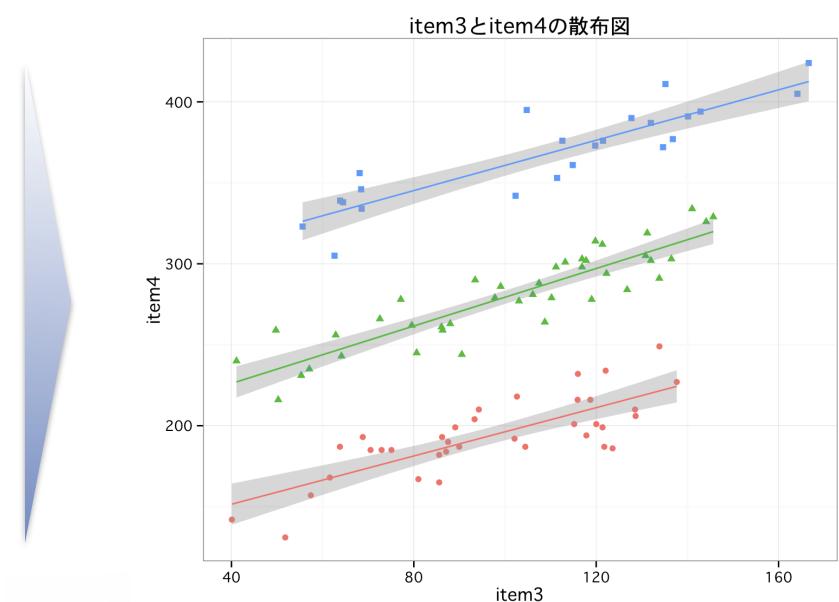
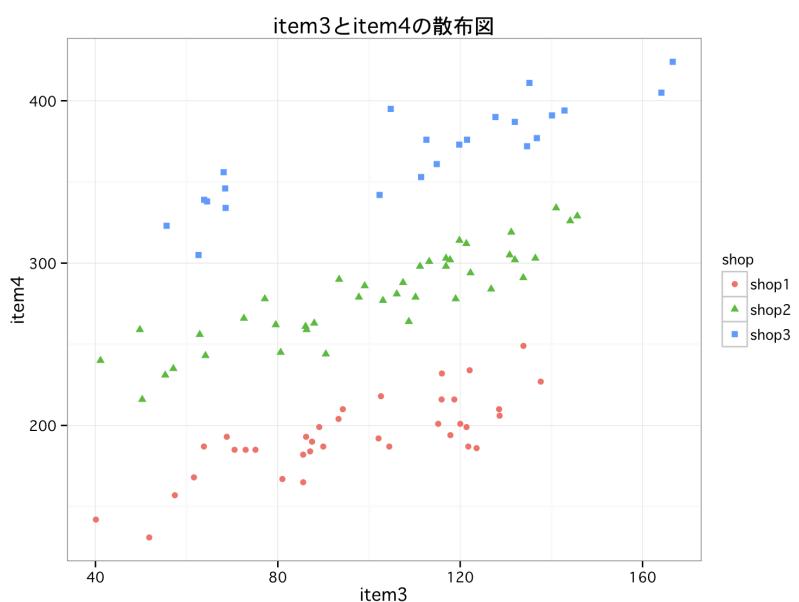
グループ別回帰直線 (Rコード)

グループ化してgeom_smooth関数を使う

```
# グループごとに回帰直線を引く
ggplot(sales.data34, aes(x=item3, y=item4, col=shop, shape=shop, group=shop)) +
  geom_point() +
  geom_smooth(method="lm") +  
  xlab("item3") +
  ylab("item4") +
  ggtitle("item3とitem4の散布図") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

グループ別回帰直線（結果）

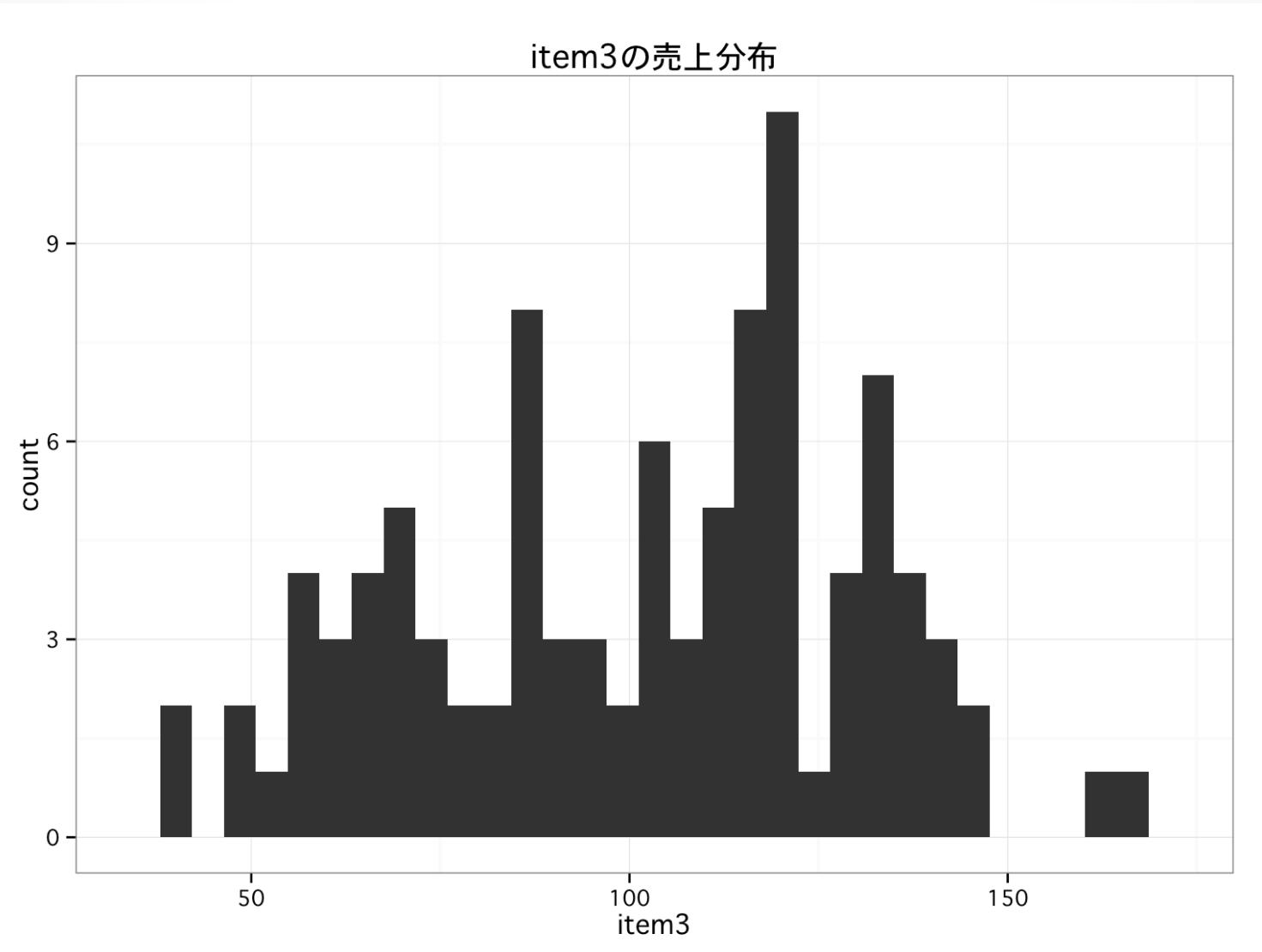
```
# グループごとに回帰直線を引く
ggplot(sales.data34, aes(x=item3, y=item4, col=shop, shape=shop, group=shop)) +
  geom_point() +
  geom_smooth(method="lm") +
  xlab("item3") +
  ylab("item4") +
  ggtitle("item3とitem4の散布図") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```



AGENDA

- 自己紹介
 - ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図
-
-

ヒストグラムを作りたい



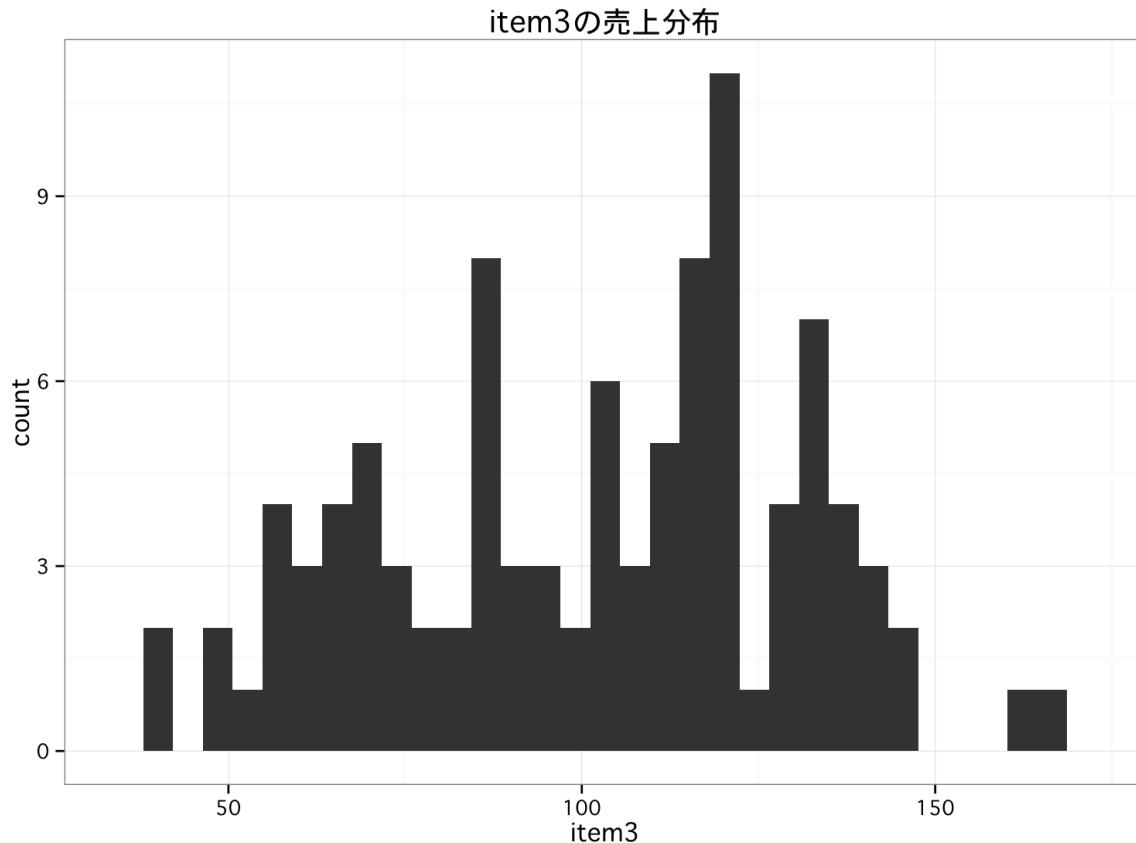
ヒストグラムの作成 (Rコード)

geom_histogram関数を使う

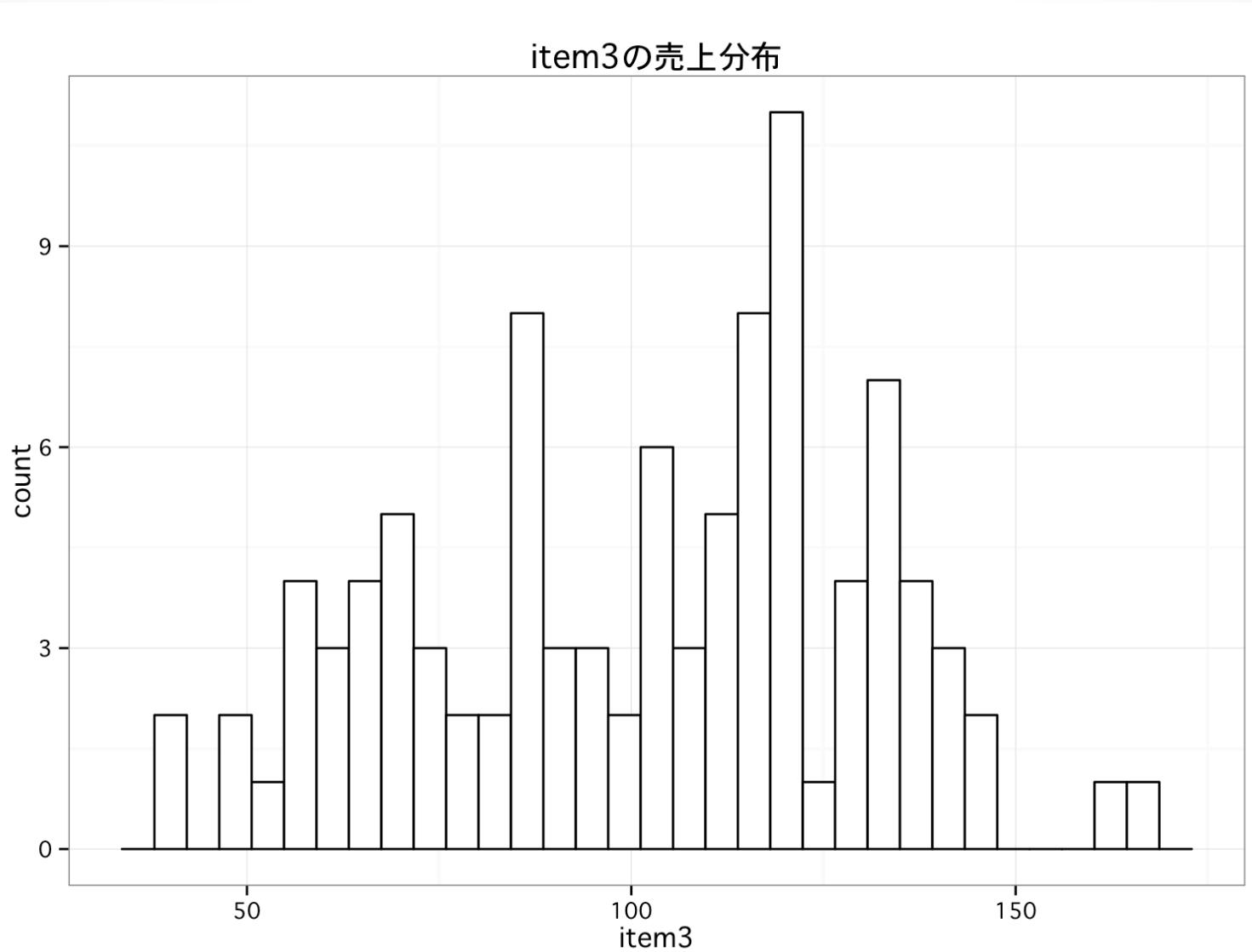
```
# 基本  
ggplot(sales.data34, aes(x=item3)) +  
  geom_histogram() +  
  ggtitle("item3の売上分布") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```

ヒストグラムの作成（結果）

```
# 基本  
ggplot(sales.data34, aes(x=item3)) +  
  geom_histogram() +  
  ggtitle("item3の売上分布") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



棒を白抜きにしたい



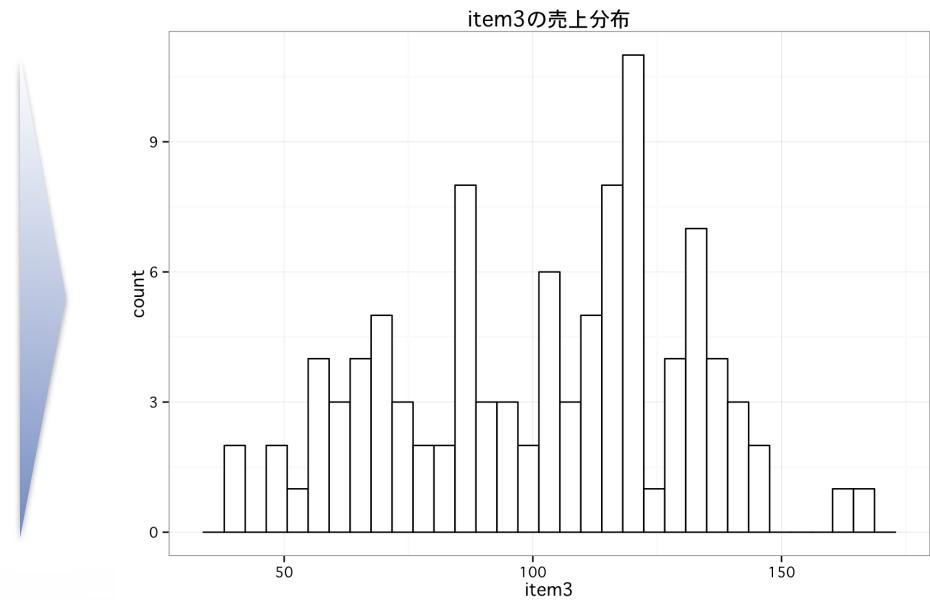
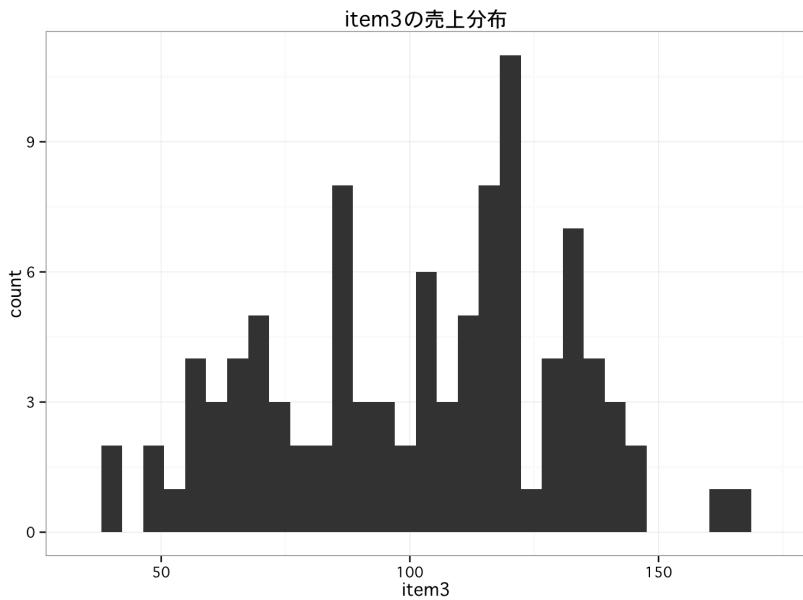
色の変更 (Rコード)

colオプションとfillオプションで
線と棒の色を指定する

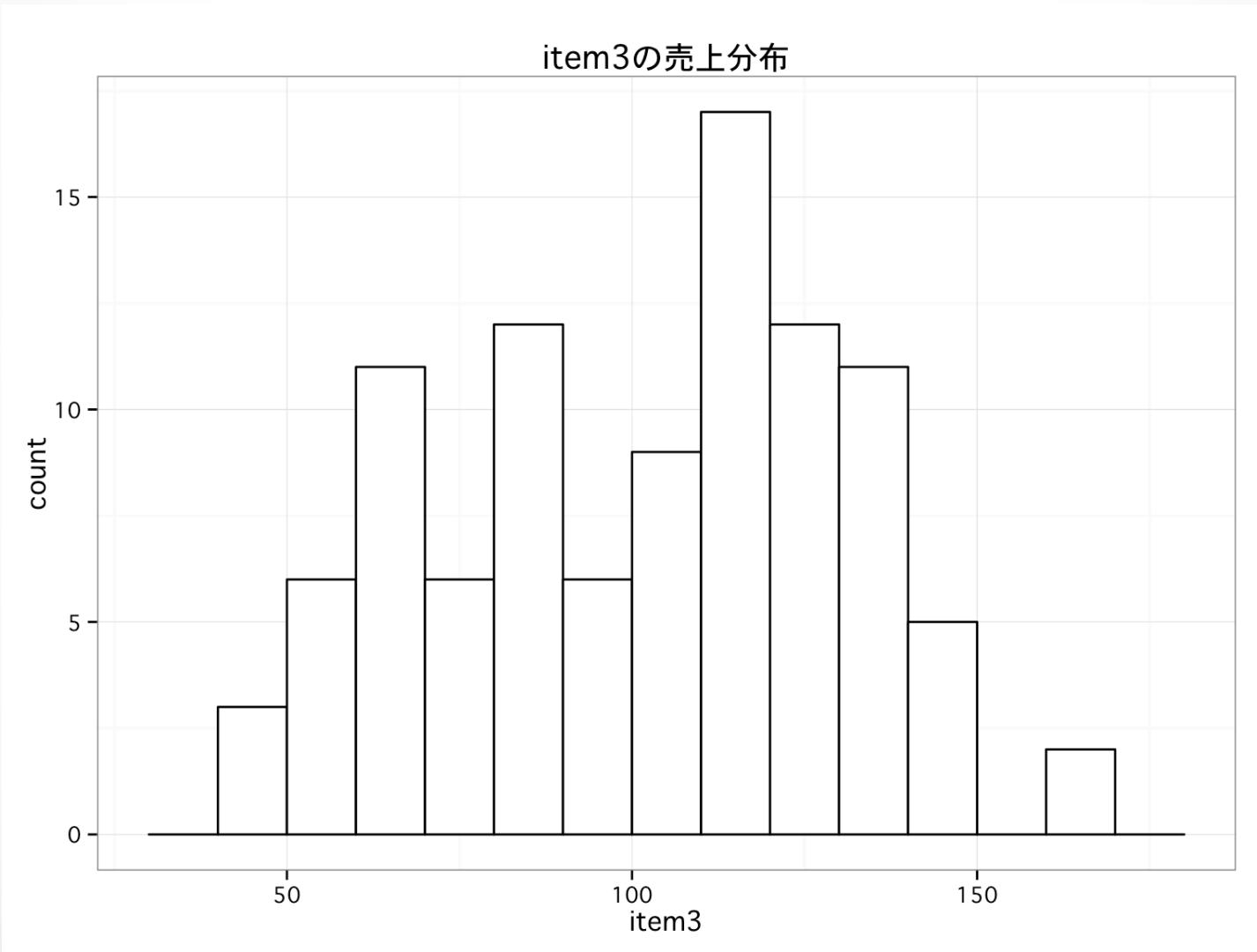
```
ggplot(sales.data34, aes(x=item3)) +  
  geom_histogram(fill="white", col="black") +  
  ggtitle("item3の売上分布") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```

色の変更 (Rコード)

```
ggplot(sales.data34, aes(x=item3)) +  
  geom_histogram(fill="white",col="black") +  
  ggtitle("item3の売上分布") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



棒の幅を変更したい

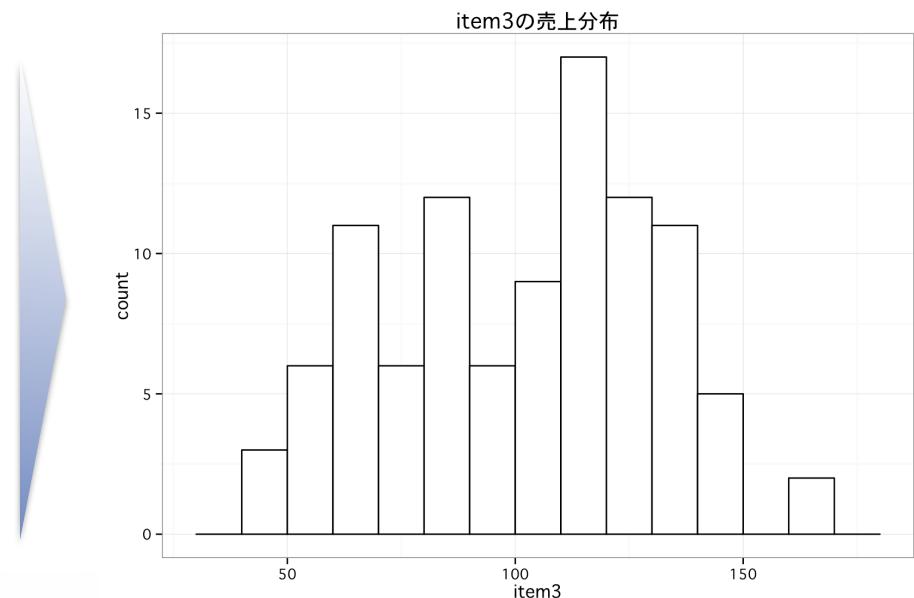
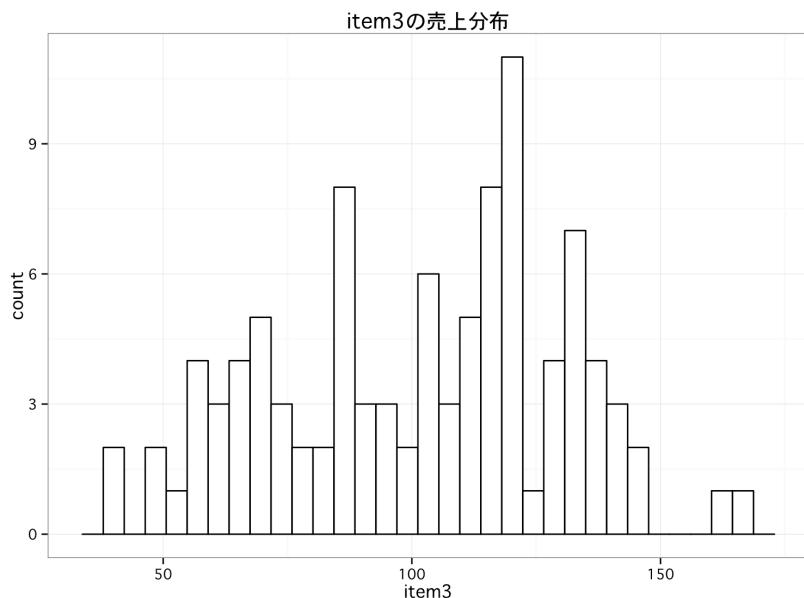


binwidthオプションを使う

```
# ビン幅を10に設定
ggplot(sales.data34, aes(x=item3)) +
  geom_histogram(fill="white", col="black", binwidth=10) +
  ggtitle("item3の売上分布") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

ビン幅の変更（結果）

```
# ビン幅を10に設定
ggplot(sales.data34, aes(x=item3)) +
  geom_histogram(fill="white", col="black", binwidth=10) +
  ggtitle("item3の売上分布") +
  scale_y_continuous(label=comma) +
  theme_bw(base_family="Osaka")
```

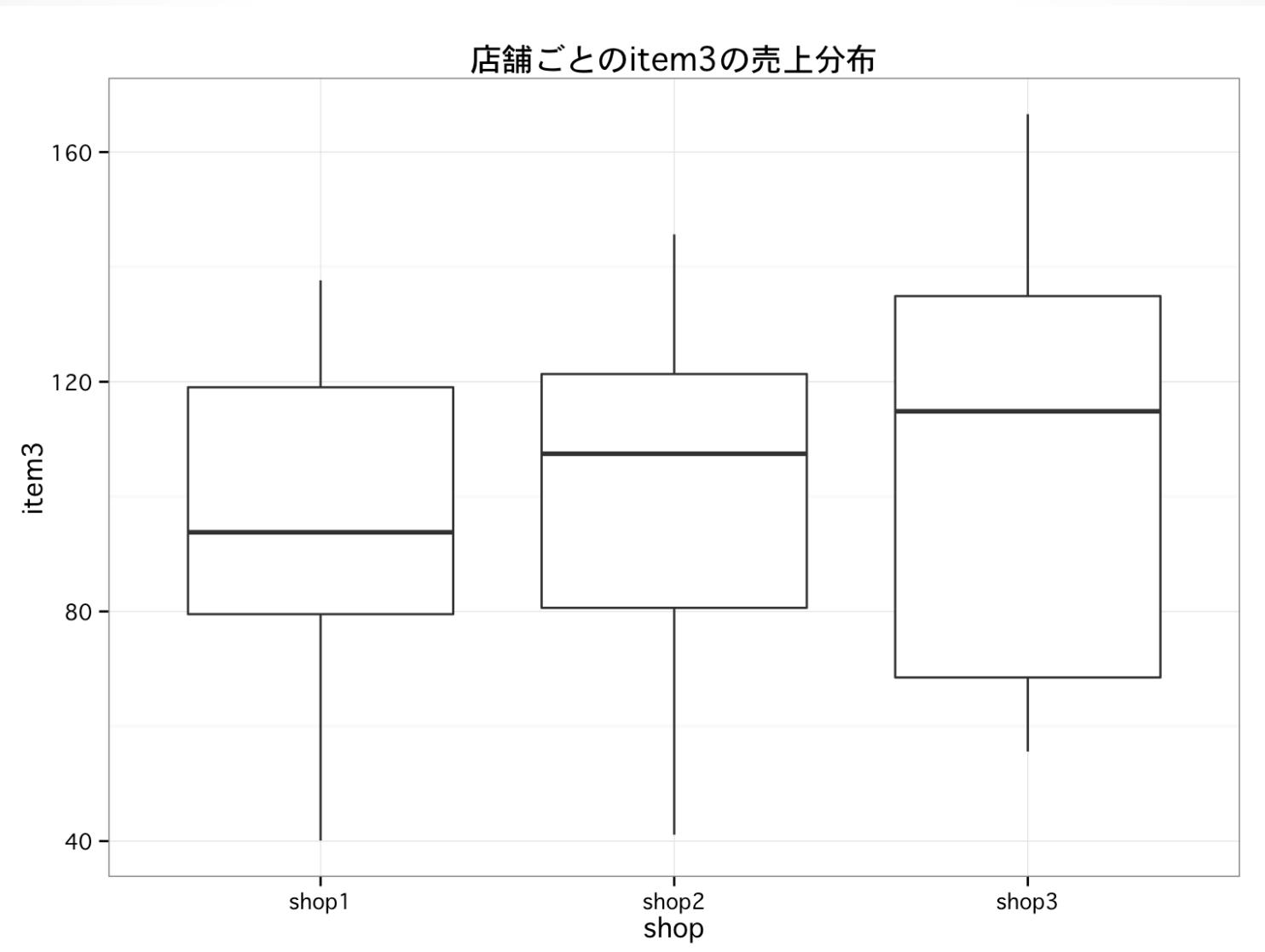


AGENDA

- 自己紹介
- ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図



箱ひげ図を作りたい

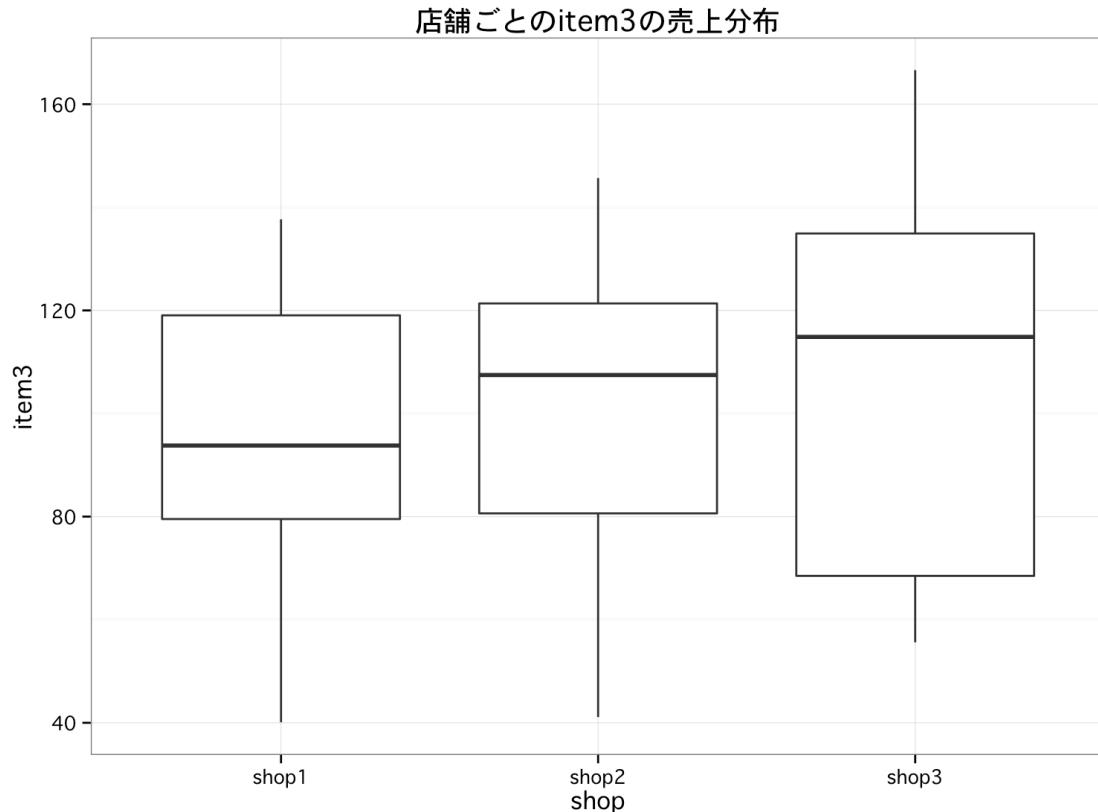


geom_boxplot関数を使う

```
ggplot(sales.data34, aes(x=shop,y=item3)) +  
  geom_boxplot() +  
  ggtitle("店舗ごとのitem3の売上分布") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```

箱ひげ図の作成（結果）

```
ggplot(sales.data34, aes(x=shop,y=item3)) +  
  geom_boxplot() +  
  ggtitle("店舗ごとのitem3の売上分布") +  
  scale_y_continuous(label=comma) +  
  theme_bw(base_family="Osaka")
```



AGENDA

- 自己紹介
 - ggplot2によるデータの可視化
 - ggplot2とは
 - 棒グラフ
 - 折れ線グラフ
 - 散布図
 - ヒストグラム
 - 箱ひげ図
-
-

Enjoy!