ggplot2によるグラフ化

紀ノ定保礼

#HijiyamaR
2015/5/23

本日の発表の目的

- 可視化ツールとしてのggplot2の紹介
  - ①すでに要約されたデータをグラフ化する
    - 例：A群は平均3点，B群は平均５点（$t(19)=3.25, p<.05$）
      - 慣習的に行われる，文字 → 図　の変換を補助する役割
  - ②探索的に，変数（間の関係）を理解する
    - ごちゃごちゃしたデータ（Row data）を用いることも多い
      - 変数を入れ替えやすいこと
      - 複数の変数を同時に表示しやすいこと

      } が重要
  - ③データを視覚的に要約する

本日の発表の目的

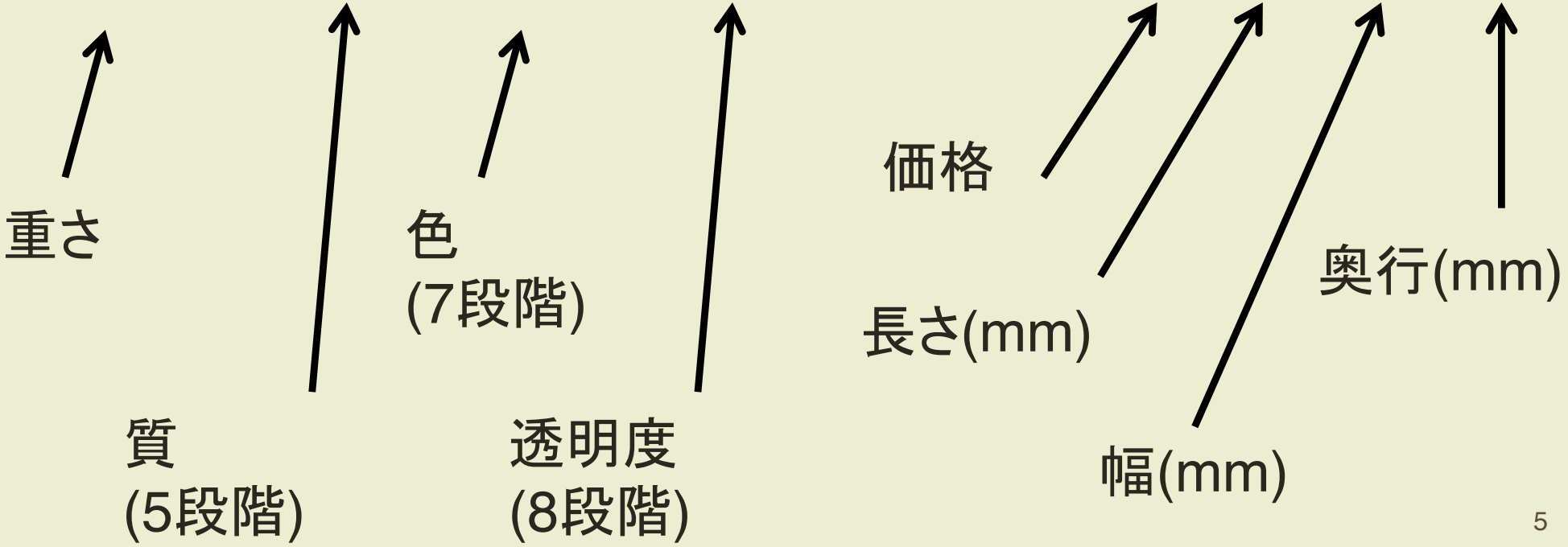
- 可視化ツールとしてのggplot2の紹介
  - ①すでに要約されたデータをグラフ化する
    - 例：A群は平均3点，B群は平均5点（$t(19)=3.25, p<.05$）
      - 慣習的に行われる，文字 → 図　の変換を補助する役割
  - ②探索的に，変数（間の関係）を理解する
    - ごちゃごちゃしたデータ（Row data）を用いることも多い
      - 変数を入れ替えやすいこと
      - 複数の変数を同時に表示しやすいこと

    } が重要
  - ③データを視覚的に要約する

サンプルデータ：diamonds

ggplot2パッケージ内に入っています（53940行×10列）

```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```



データが多いもので…

- いったん500行に減らしておきましょう

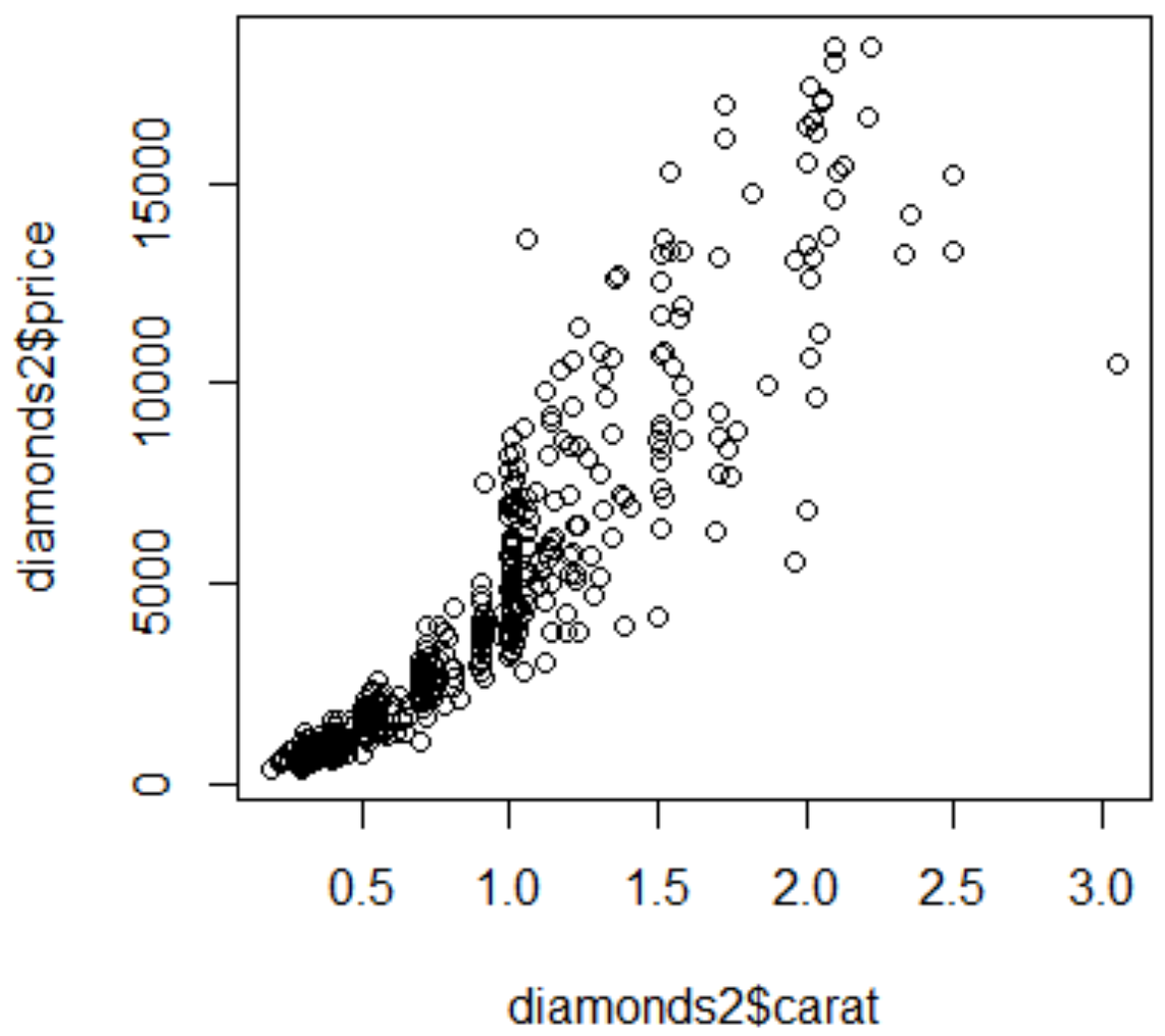
```
library(ggplot2)
diamonds2 <- diamonds[sample(nrow(diamonds),500),]
```

※注）ggplot2の前にdplyr及びplyrパッケージをロードする
場合は、必ずplyr → dplyrの順に

重さと価格の関係をみる

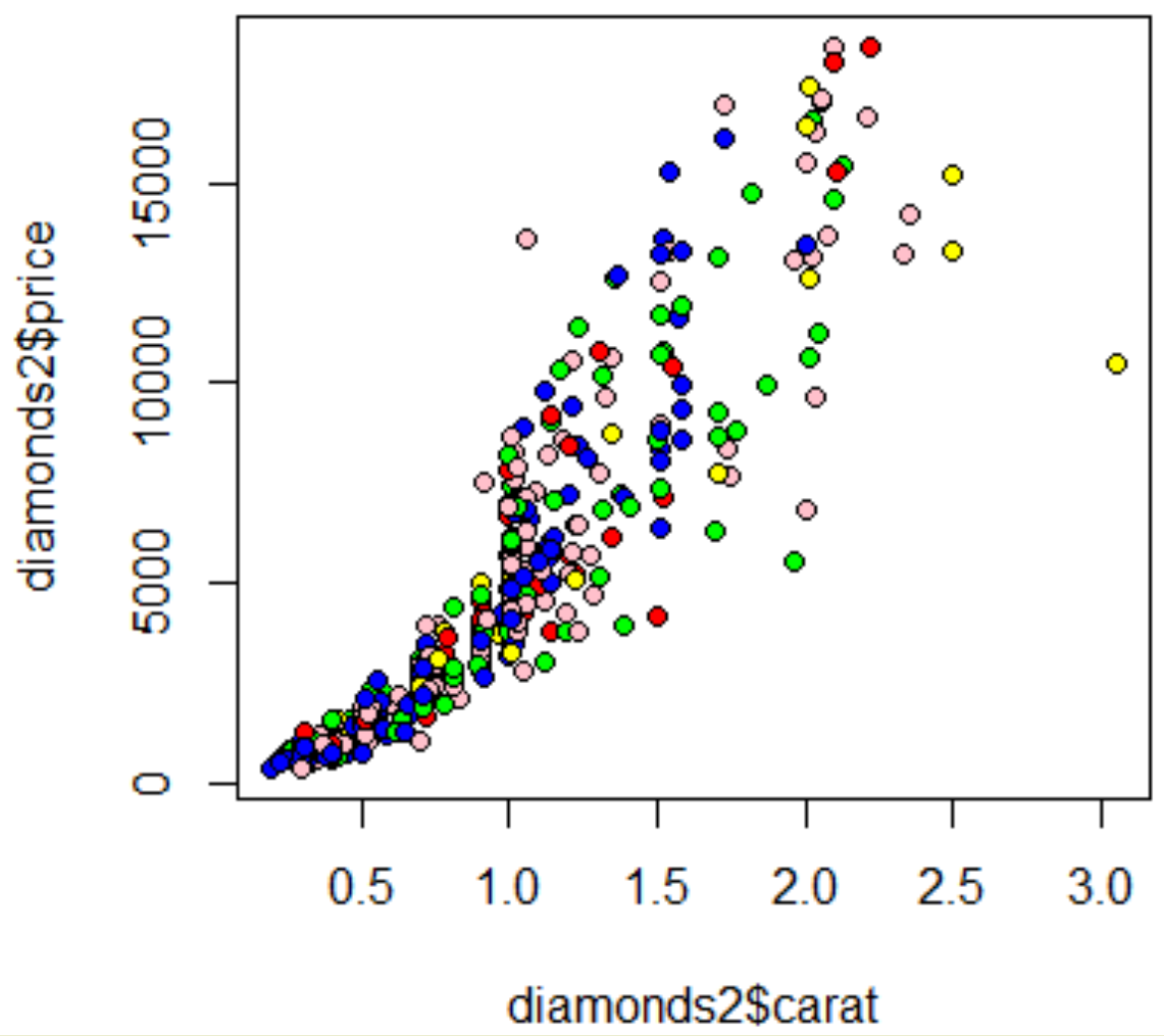
- 従来のbaseパッケージで描画する場合

```
plot(diamonds2$carat, diamonds2$price)
```



せめて，質（5段階）別に関係を理解したい

```
plot(diamonds2$carat,diamonds2$price,
    pch=21,bg=c("yellow","red","green","blue","pink")[diamonds$cut])
```



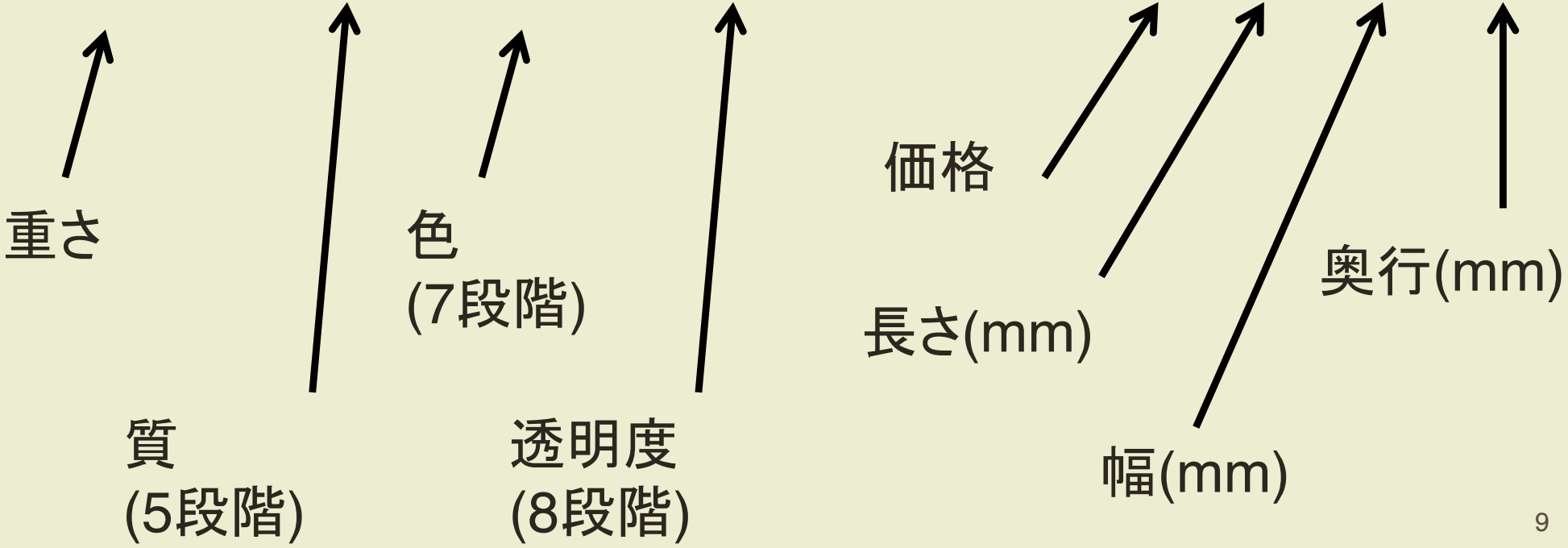
出来ますが…

自分で各水準を定義する
必要あり

変数の交換

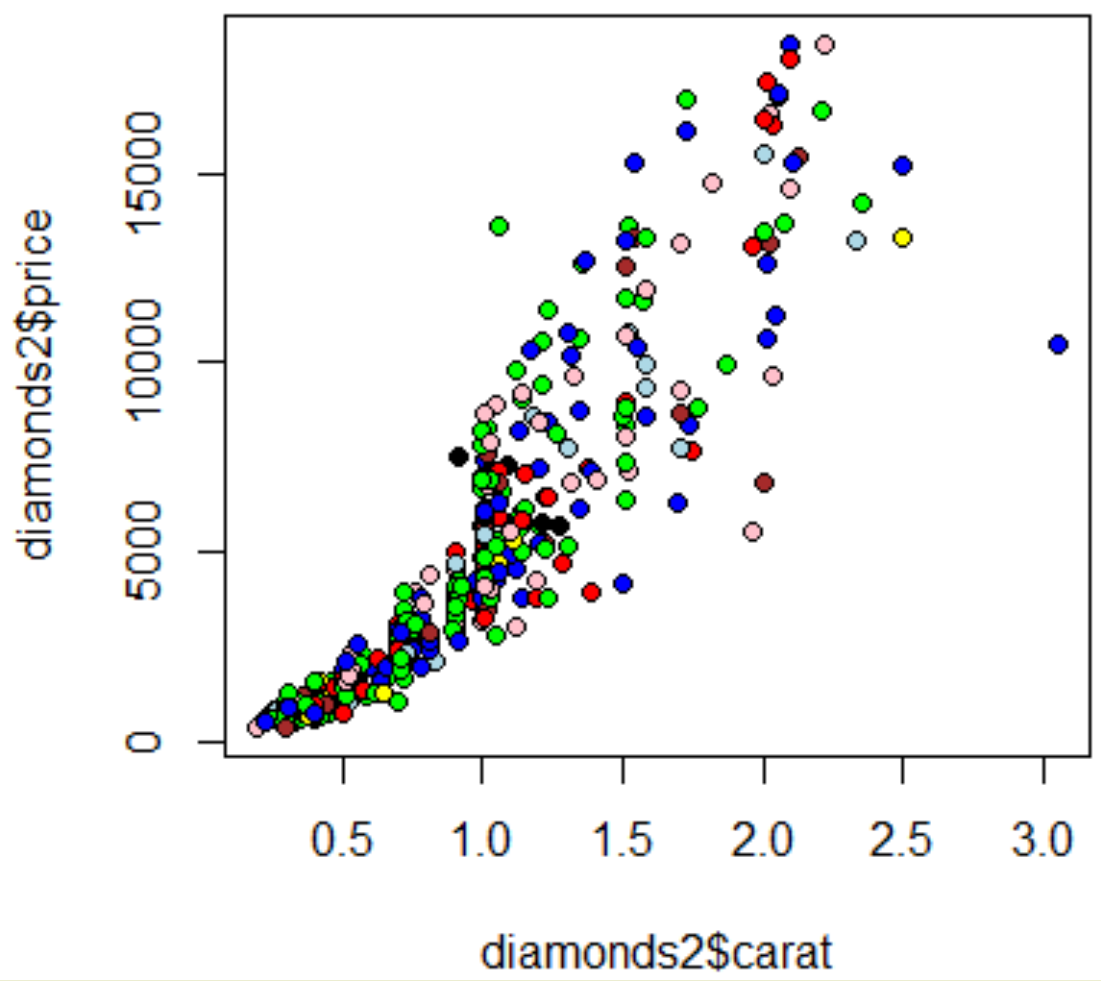
- 要請①：質ではなく，透明度で区別したい

```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```



出来ますが…

```
plot(diamonds2$carat,diamonds2$price,
    pch=21,bg=c("yellow","red","green","blue","pink",
    "brown","lightblue","black")[diamonds$clarity])
```

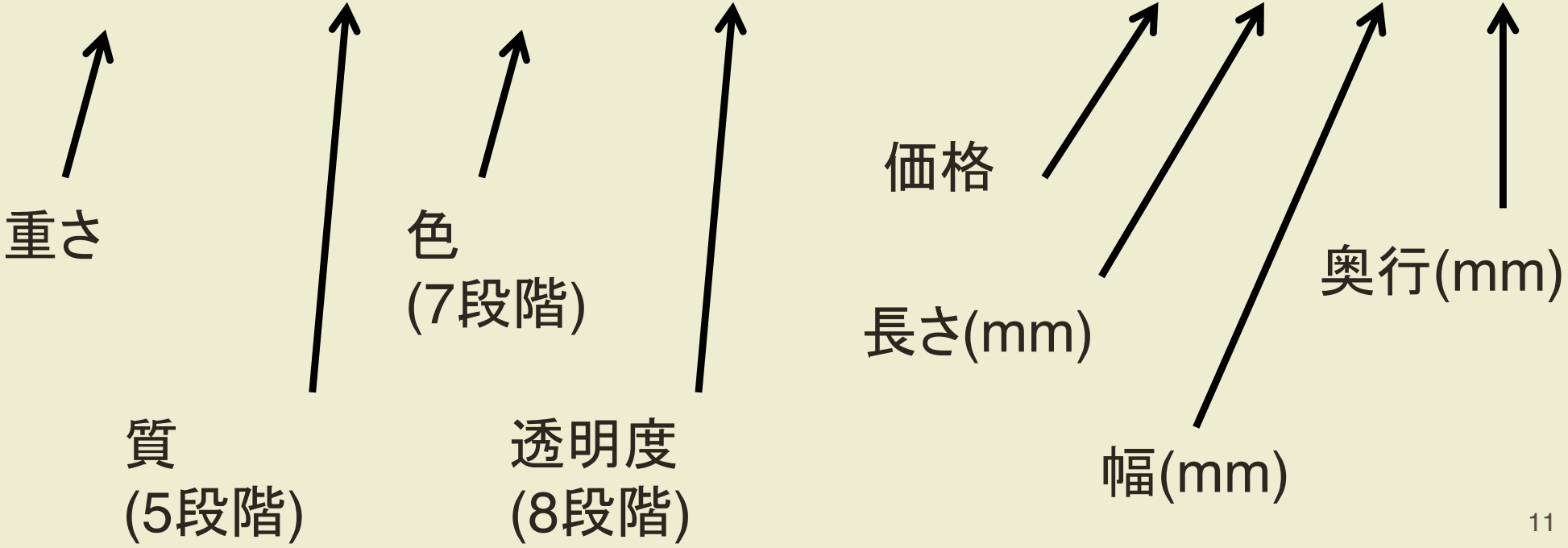


水準数が増えた分，
塗り分ける色を
改めて定義する必要

変数の追加

- 要請②：質と色と透明度で区別したい

```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```



想像がつくと思いますが…

- 基準（要因）の数
- 水準（条件）の数

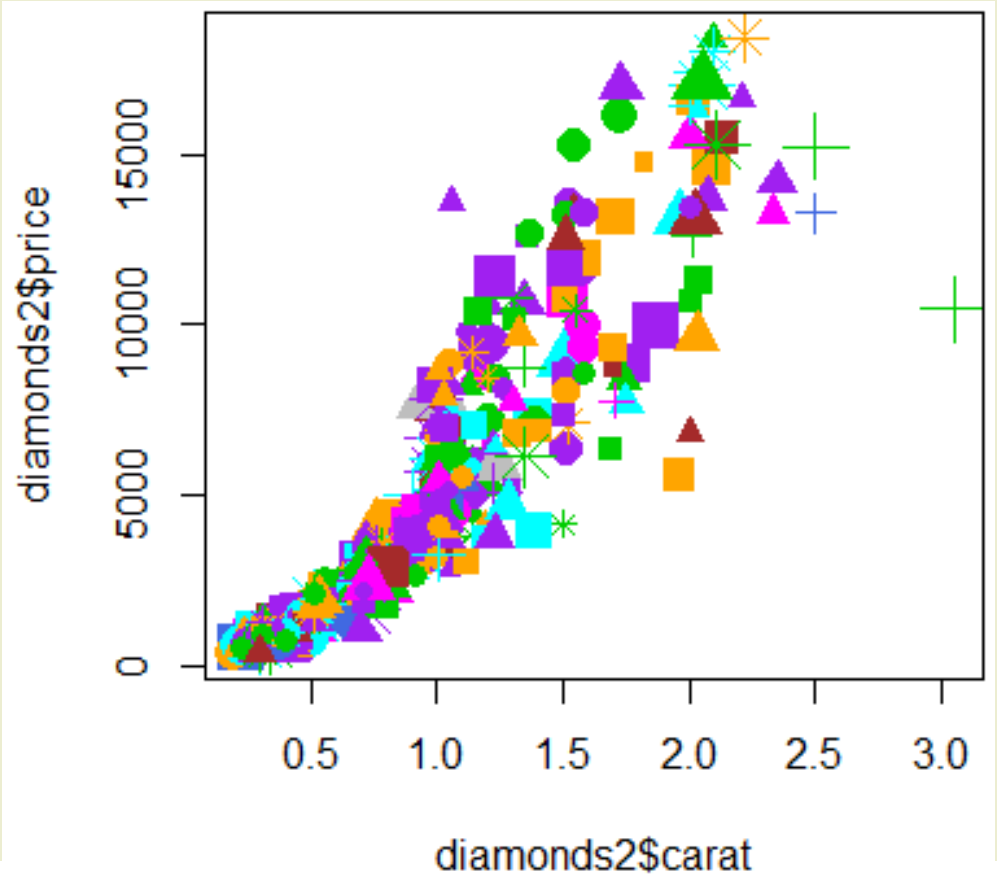
・・・が増えるにしたがい，
「区別の仕方」も定義する必要

- 例：質(5段階)と色(7段階)と透明度(8段階)で 区別したい
  - 5 + 7 + 8 = 20通りの指定

変数が増えるほど大変…

記号 →
サイズ →
色 →

```
plot(diamonds2$carat,diamonds2$price,
     pch=c(3,8,15,16,17)[diamonds$cut],
     cex=c(1,1.25,1.5,1.75,2,2.25,2.5)[diamonds$color],
     col=c("royalblue","cyan","purple","green3","orange",
           "brown","magenta","gray")[diamonds$clarity])
```



そこでggplot2 packageの出番

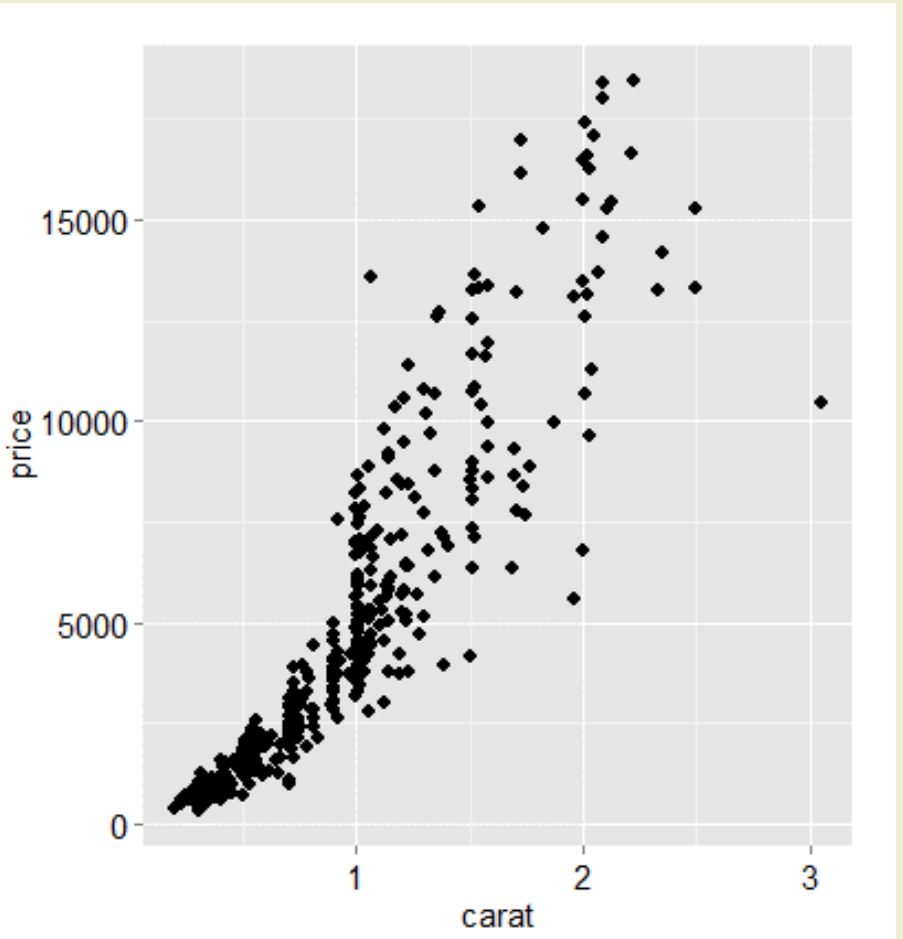
重さと価格の関係をみる

データフレーム

aes()内で,
各軸の変数や
データの弁別基準を
指定

```
ggplot(diamonds2,aes(y=price,x=carat))+
    geom_point()
```

どのような形式で描画するか
（pointなら点）



質（5段階）別に，重さと価格の関係をみる

```
ggplot(diamonds2,aes(y=price,x=carat,color=cut))+
    geom_point()
```

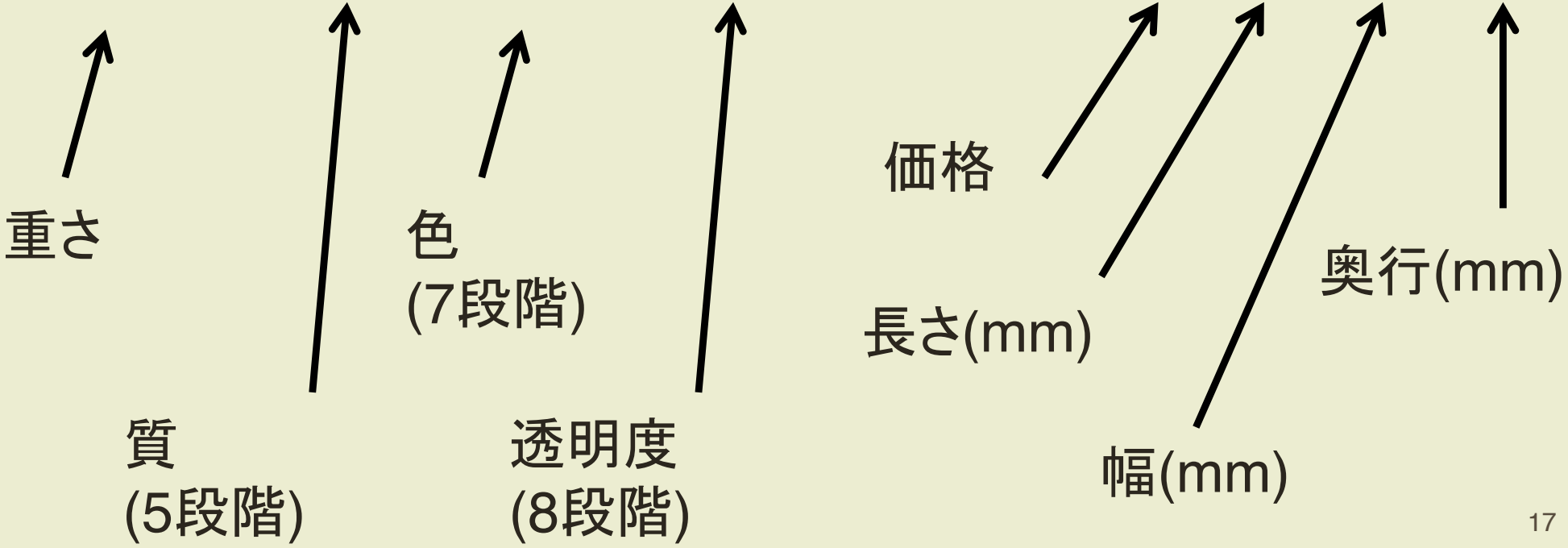
弁別基準はaes()内に

配色もさることながら，
コードがすっきり！

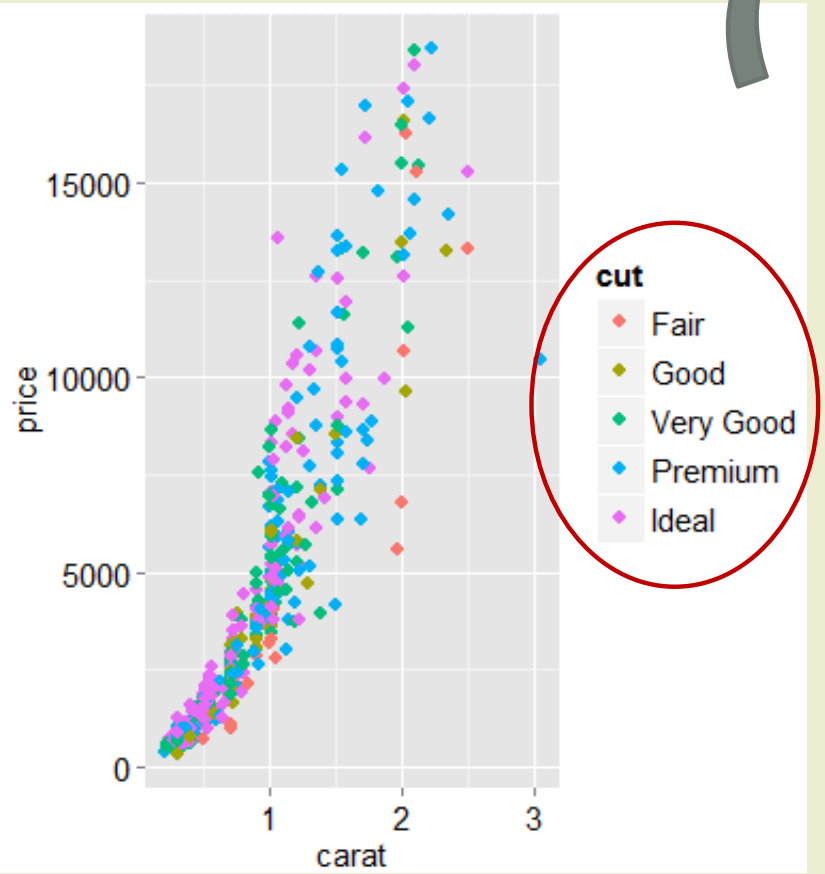
変数の交換

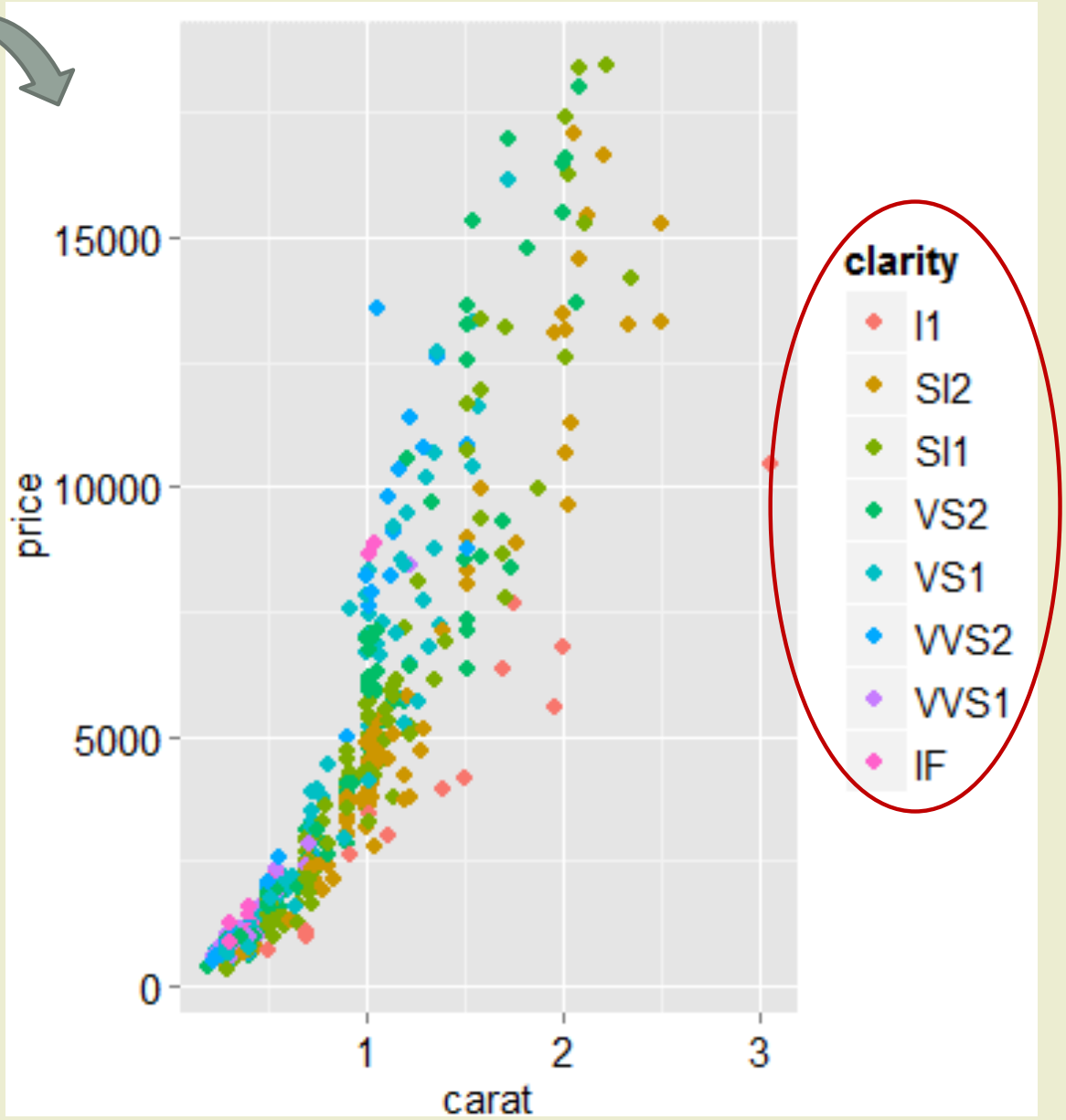
- 要請①：質ではなく，透明度で区別したい

```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```



```
ggplot(diamonds2,aes(y=price,x=carat,color=clarity))+
    geom_point()
```



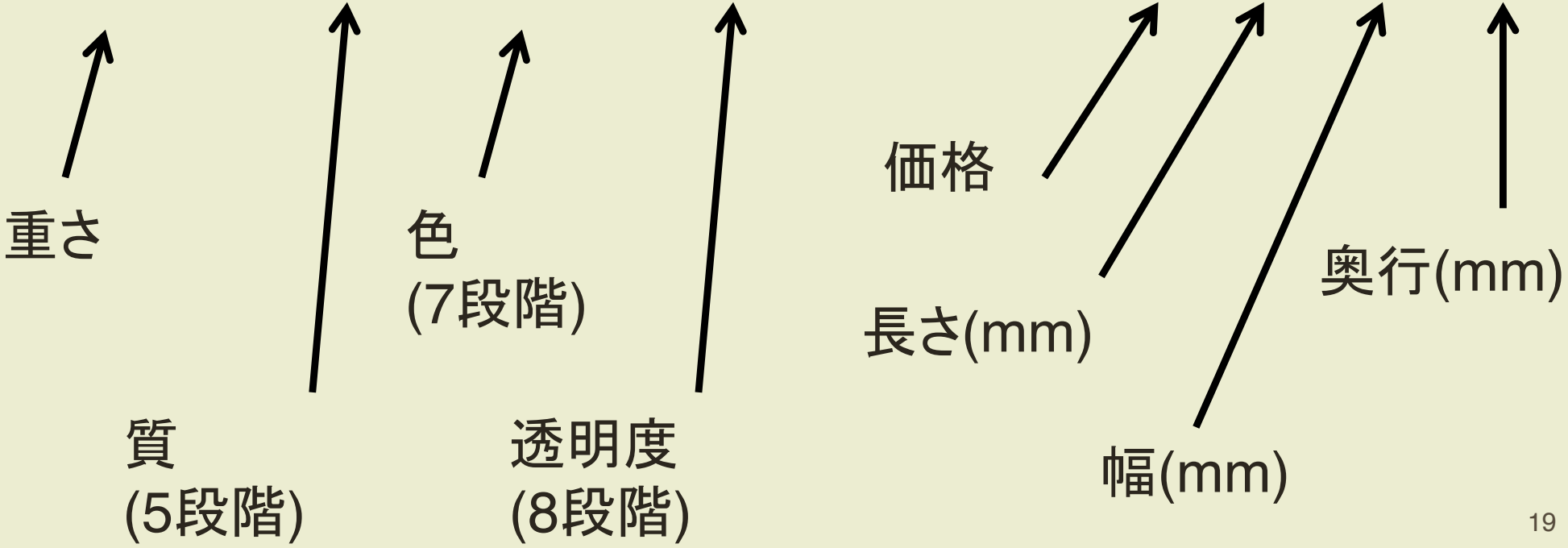


確かに，塗分け基準が
変わっている

変数の追加

- 要請②：質と色と透明度で区別したい

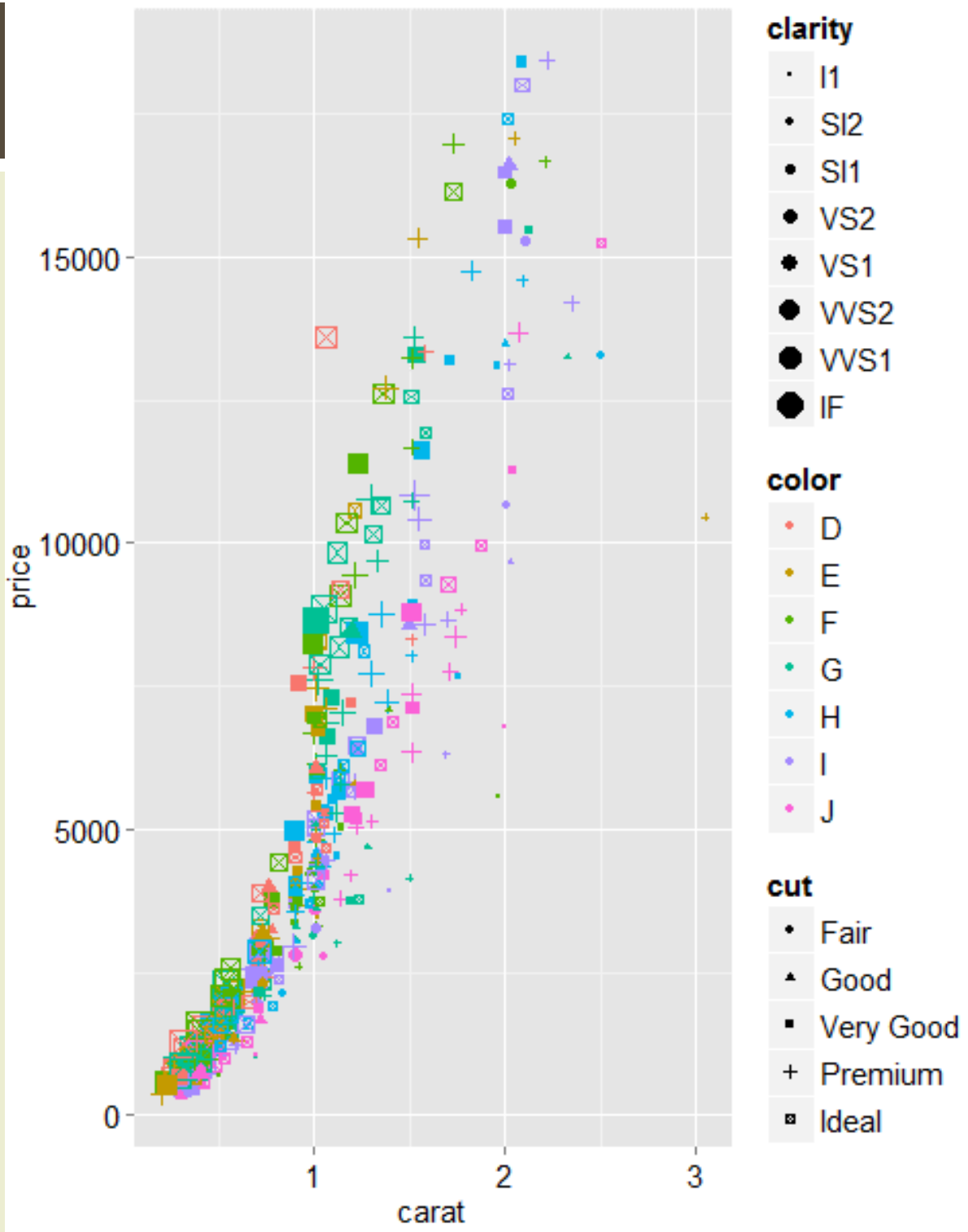
```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```



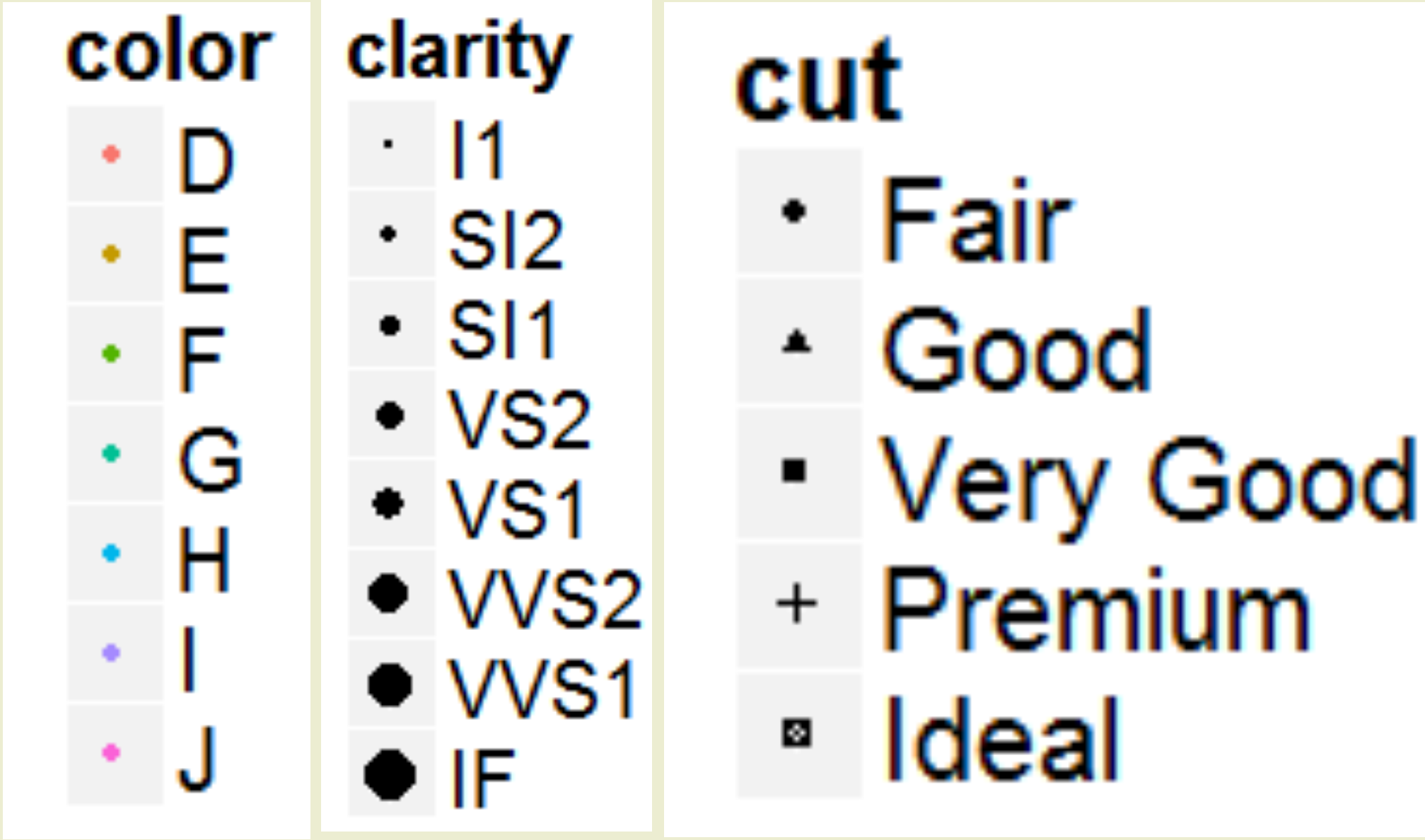
方法その1

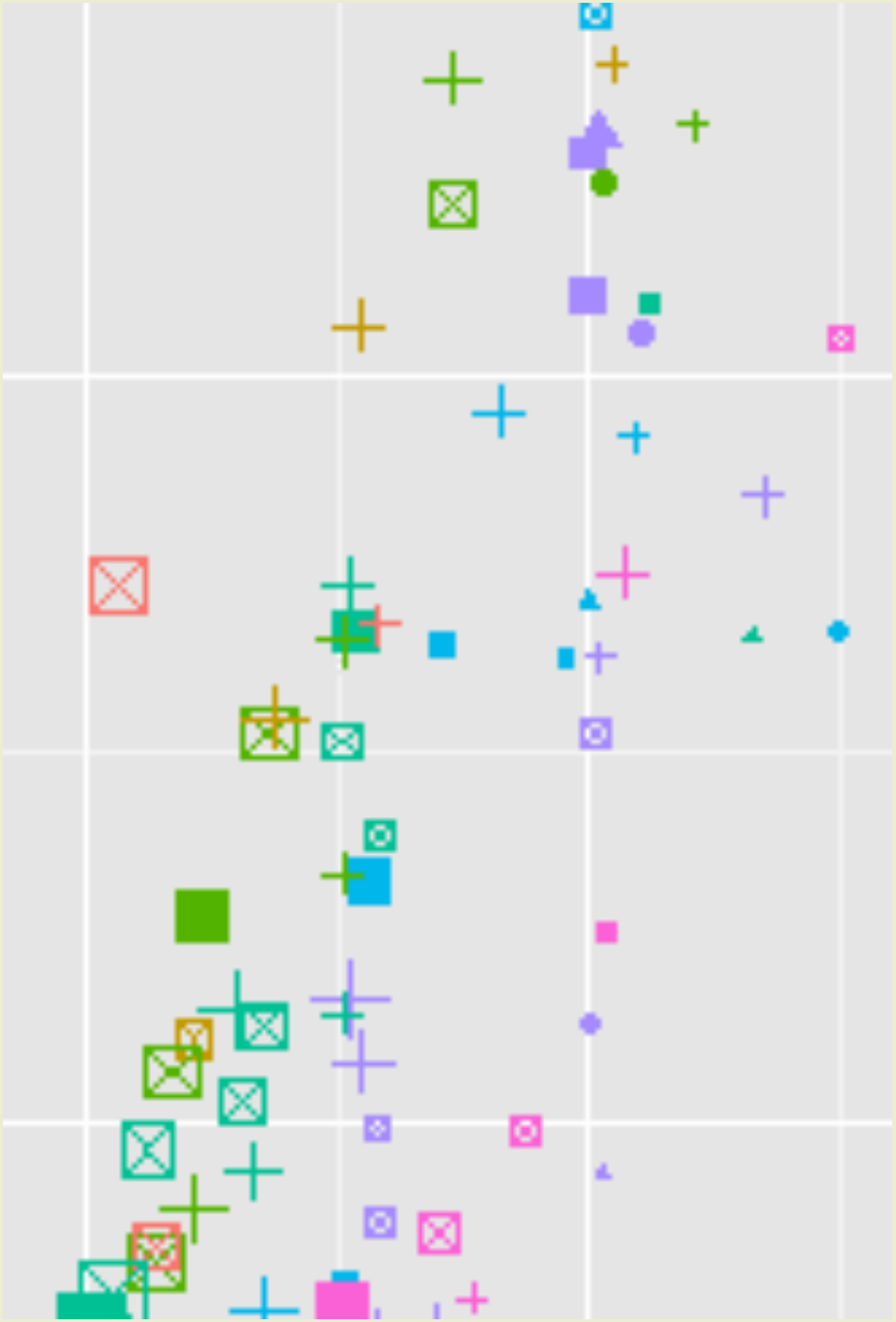
```
ggplot(diamonds2,
    aes(y=price,x=carat,
    color=color,
    size=clarity,
    shape=cut))+
    geom_point()
```

確かに，弁別基準が
増えている



小さくて分かりにくいと思うので，拡大すると…





小さくて
分かりにくいと思うので，
一部を拡大すると…

描画の仕組み

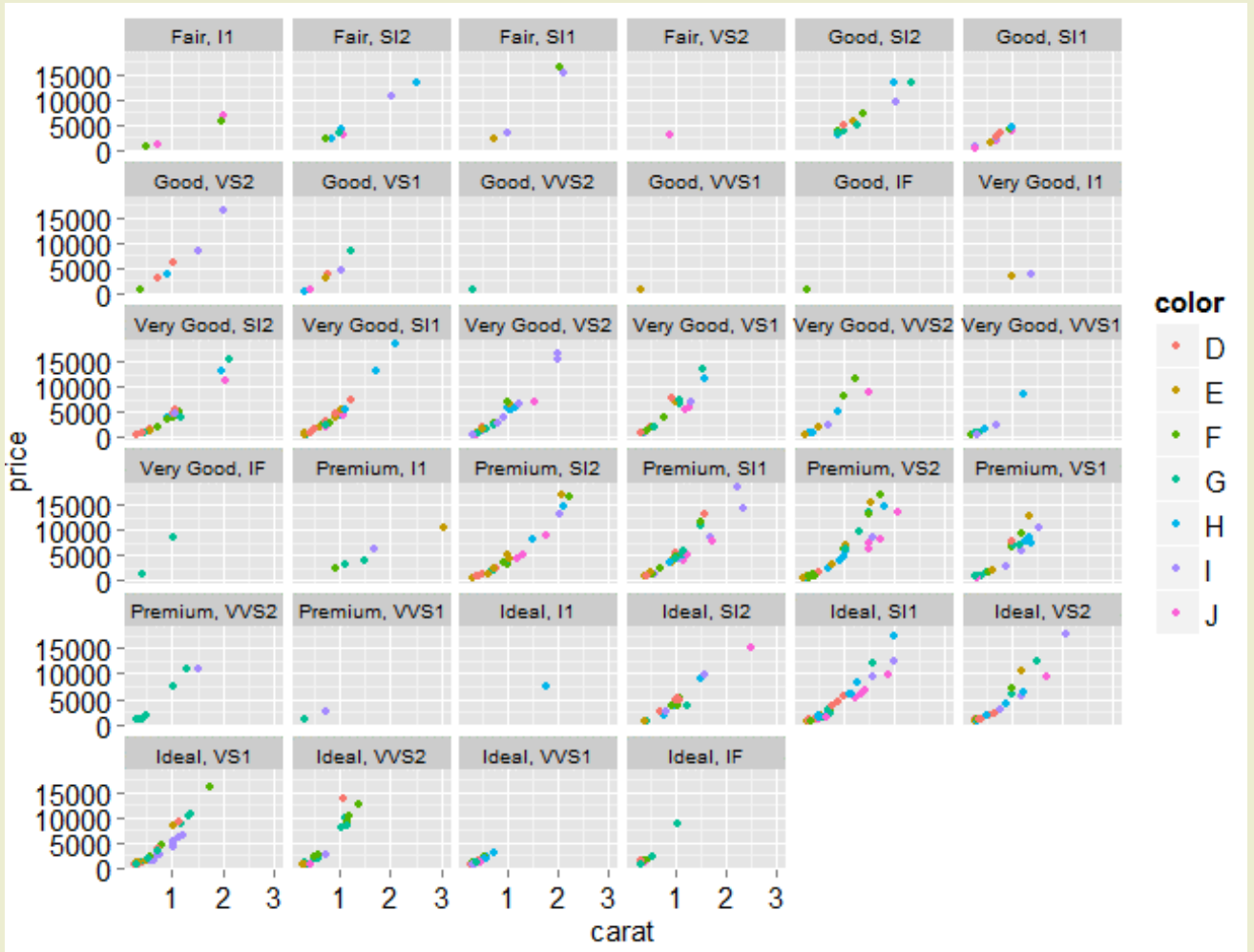
- レイヤーを重ねて図を調整
- 例（もちろん他にも指定可能）

基本レイヤー（データ指定）

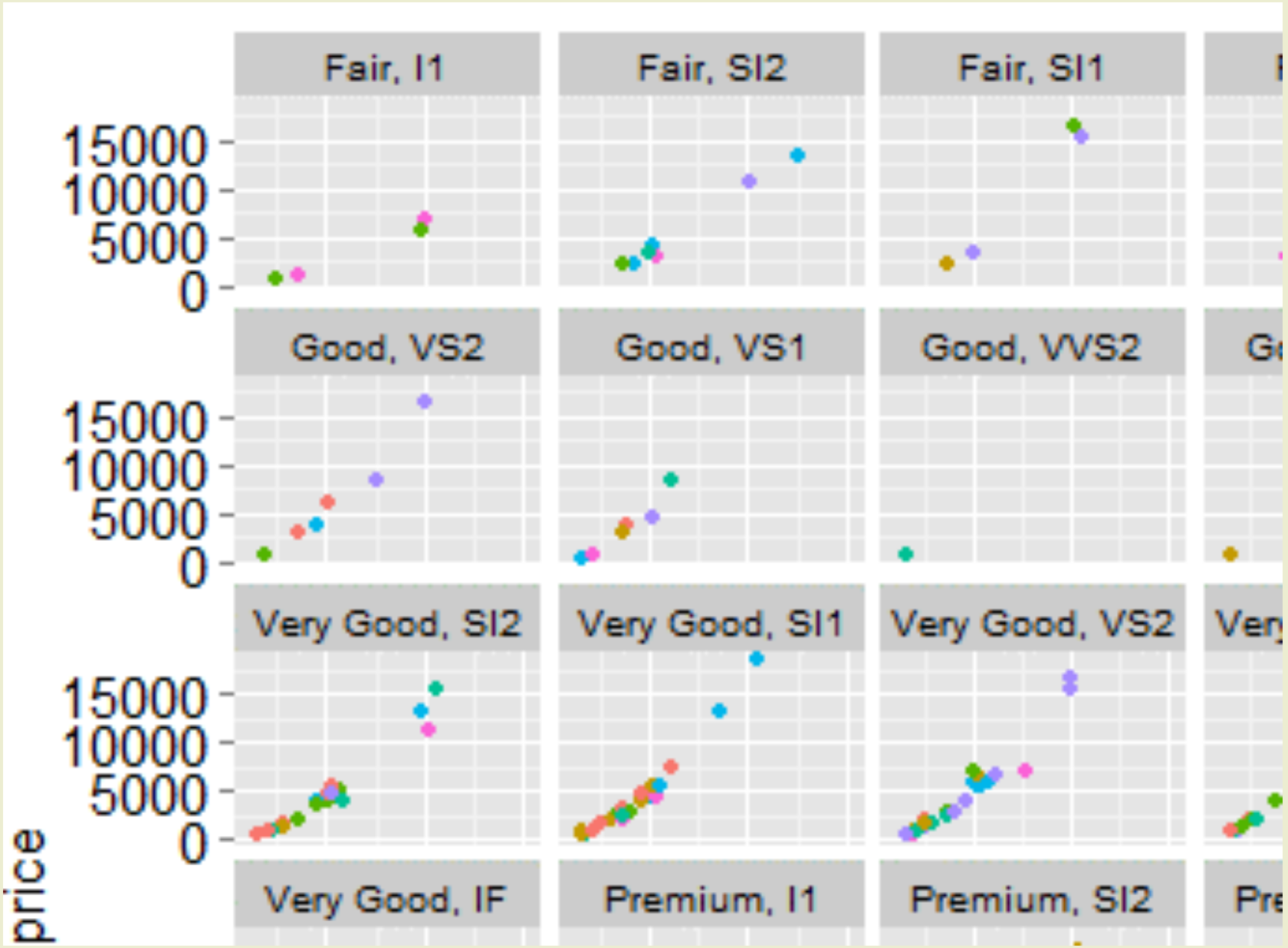
```
ggplot(iris,aes(y=Petal.Length,x=Sepal.Length,color=Species))+
    geom_point()+          プロットの仕方の指定
    theme_bw()+            背景の指定
    ylim(0,10)+            軸目盛の指定
    theme(axis.text.x=element_text(size=15),
          axis.text.y=element_text(size=15))   軸の書式の指定
```

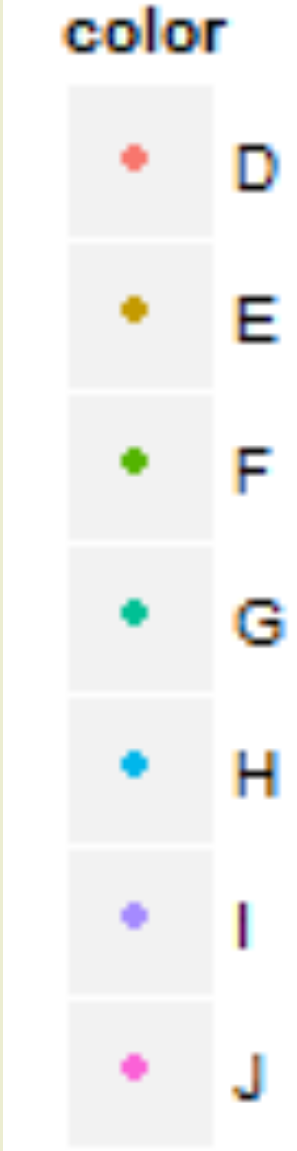
方法その2（FACETを用いる）

```
ggplot(diamonds2,aes(y=price,x=carat,color=color))+
    geom_point()+
    facet_wrap(cut~clarity)
```



小さくて見えないと思うので拡大





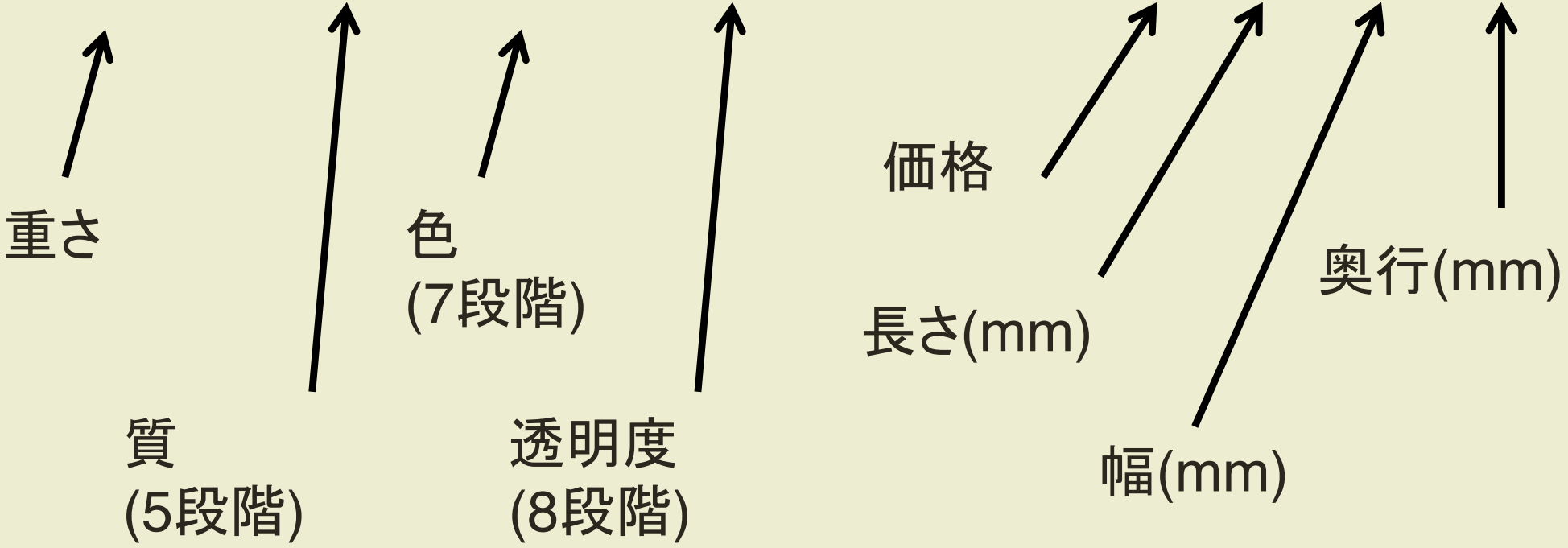
本日の発表の目的

- 可視化ツールとしてのggplot2の紹介
  - ①すでに要約されたデータをグラフ化する
    - 例：A群は平均3点，B群は平均5点（$t(19)=3.25, p<.05$）
      - 慣習的に行われる，文字 → 図　の変換を補助する役割
  - ②探索的に，変数（間の関係）を理解する
    - ごちゃごちゃしたデータ（Row data）を用いることも多い
      - 変数を入れ替えやすいこと
      - 複数の変数を同時に表示しやすいこと

      } が重要
  - ③データを視覚的に要約する

たとえデータが53940行もあったって

```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```



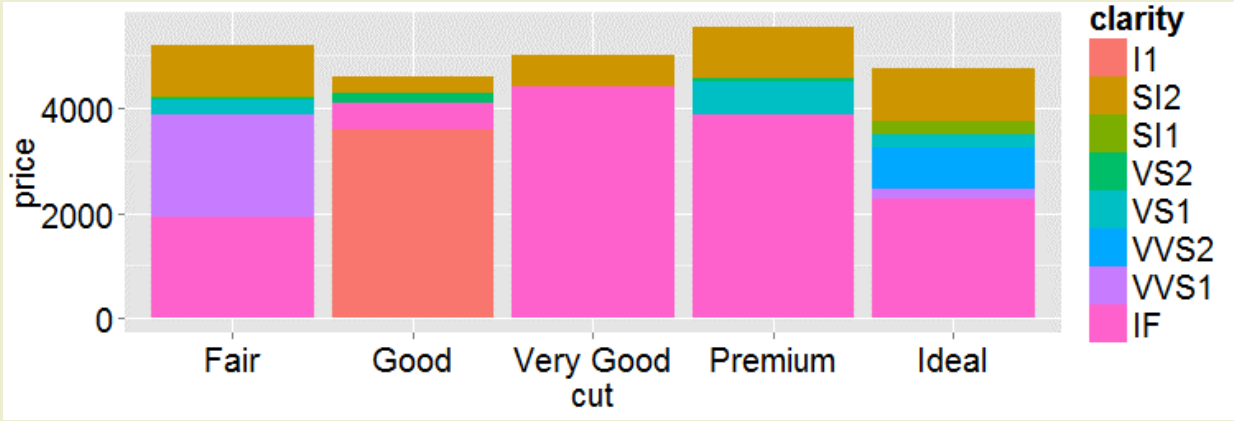
一気に視覚的に要約

- 例えば，質（5段階）別に価格の平均をみたい

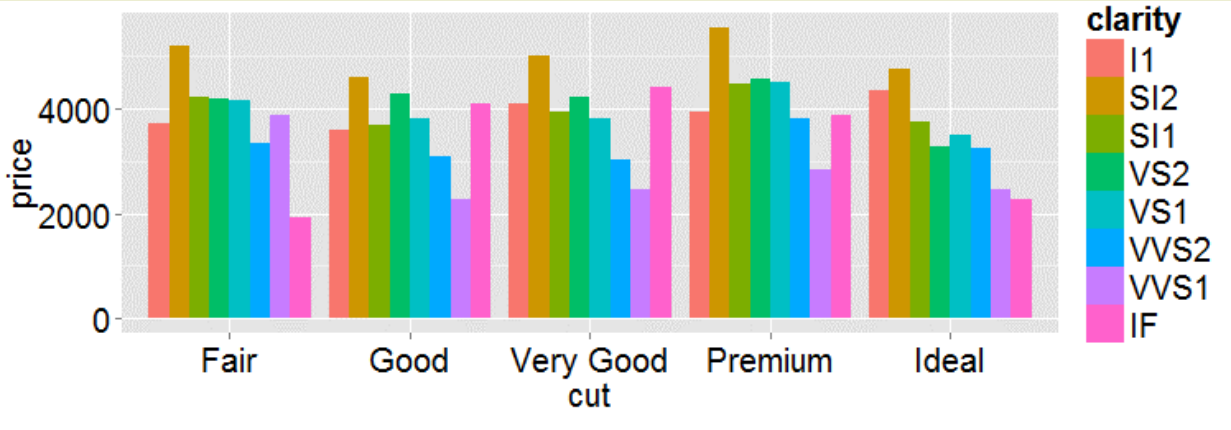
```
ggplot(diamonds,aes(y=price, x=cut))+
    stat_summary(fun.y=mean, geom="bar")
```

- 例えば，質と透明度別に価格の平均をみたい

```
ggplot(diamonds,aes(y=price,x=cut,fill=clarity))+
    stat_summary(fun.y=mean,geom="bar")
```

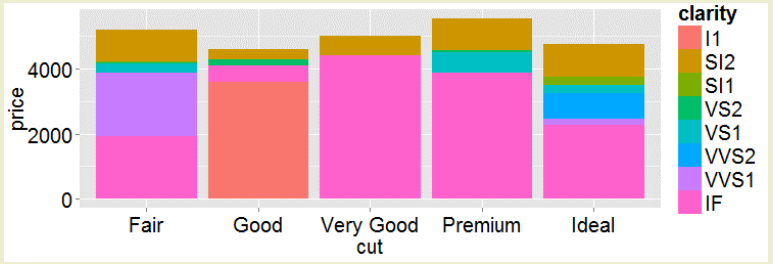


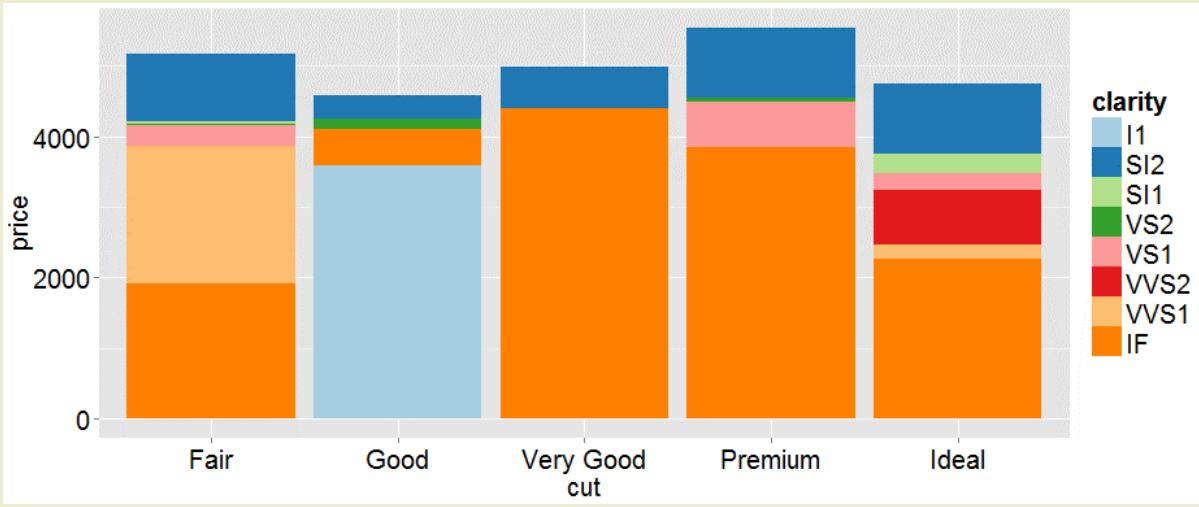

```
ggplot(diamonds,aes(y=price,x=cut,fill=clarity))+
    stat_summary(fun.y=mean,geom="bar",position= "dodge")
```

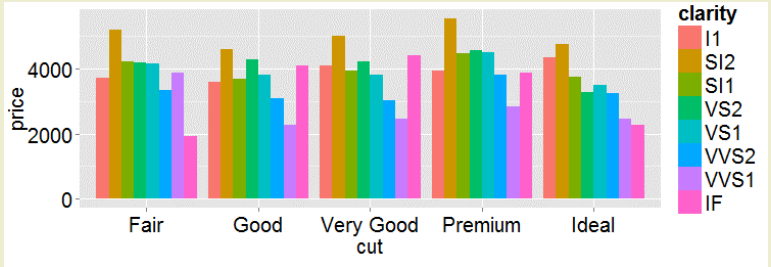


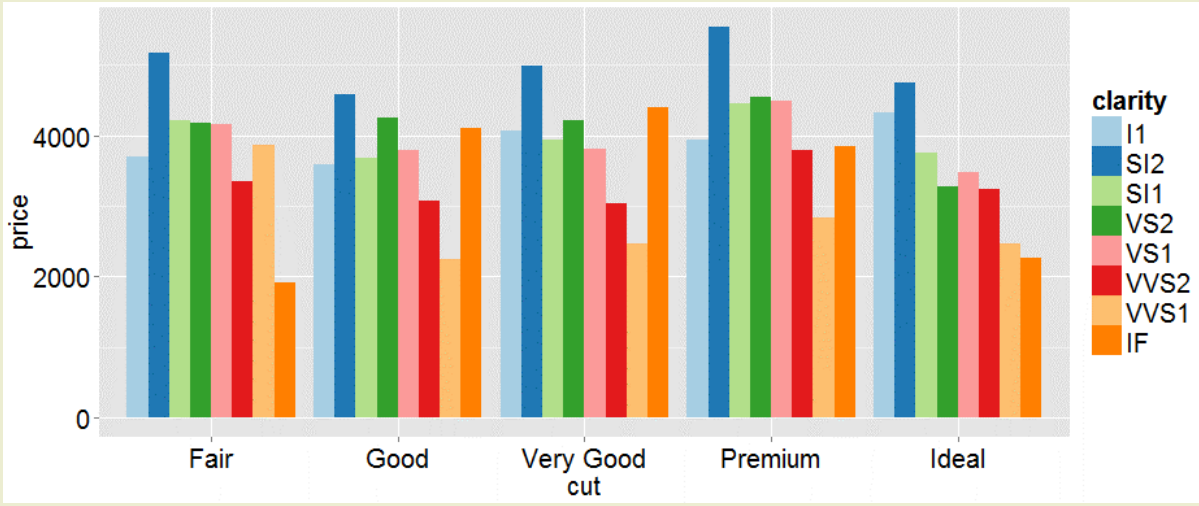
色セットを変えたくなったら

- あっという間



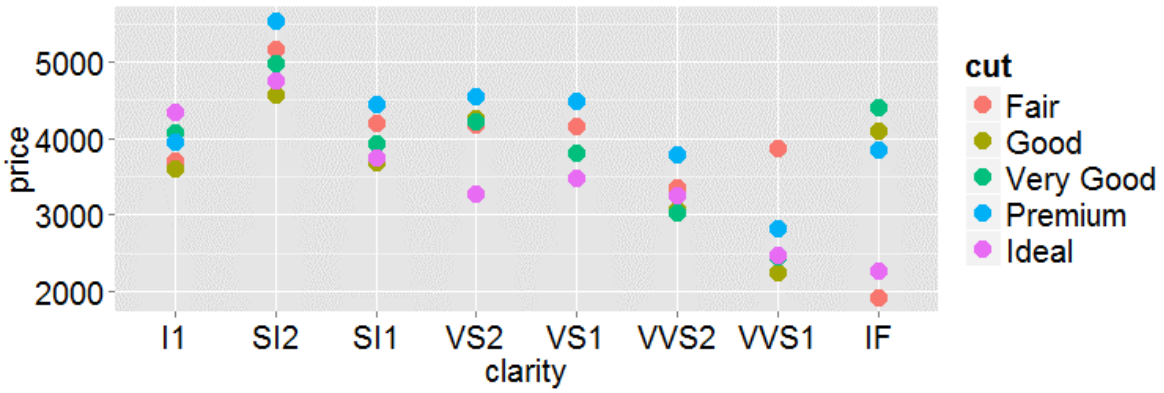




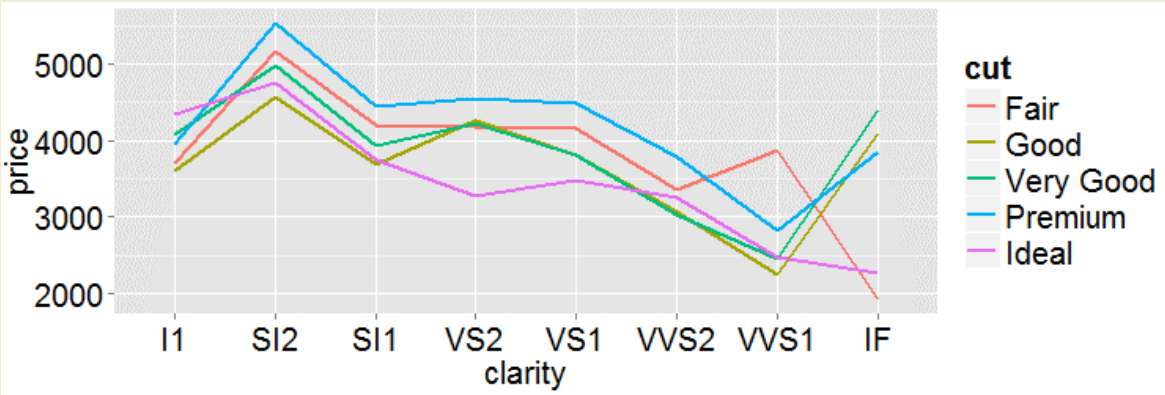


レイヤーを重ね描き出来るということは…

```
ggplot(diamonds,aes(y=price,x=clarity,group=cut,color=cut))+
  stat_summary(fun.y=mean,geom="point")
```

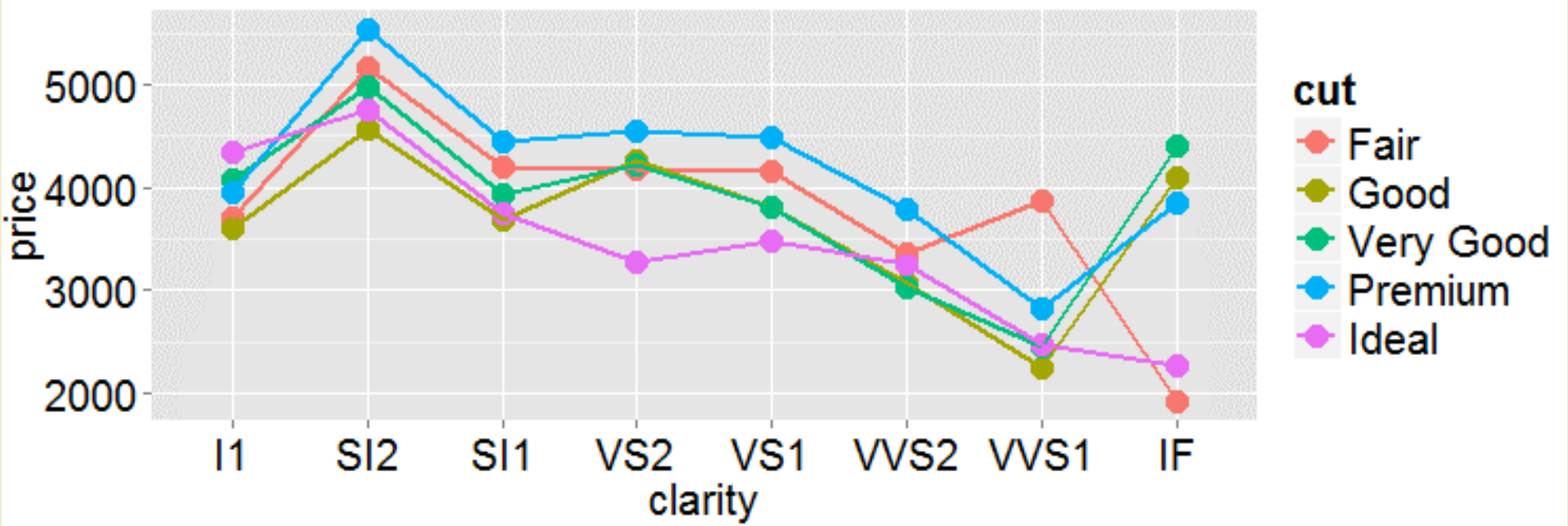



```
ggplot(diamonds,aes(y=price,x=clarity,group=cut,color=cut))+
  stat_summary(fun.y=mean,geom="line")
```

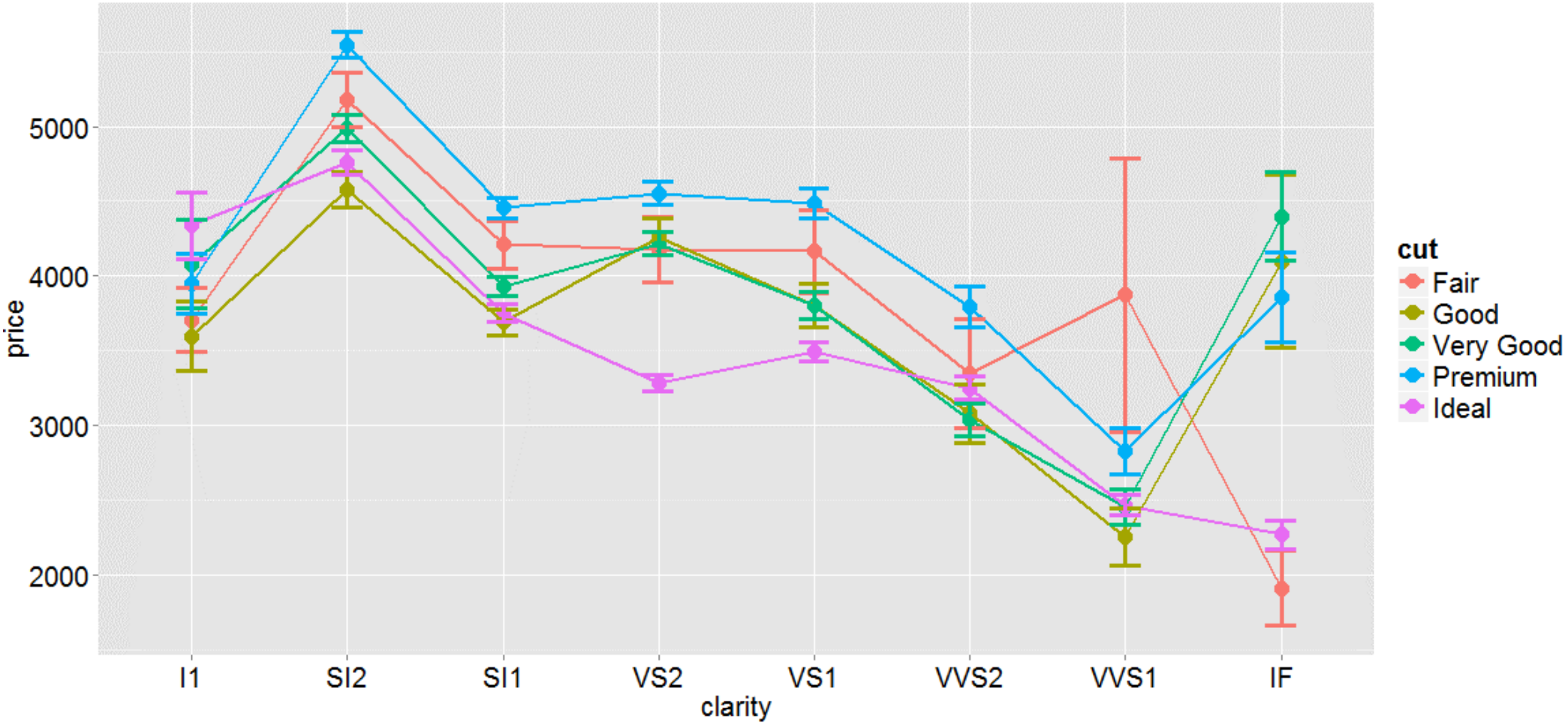


レイヤーの重ね描きの成せるわざ

```
ggplot(diamonds,aes(y=price,x=clarity,group=cut,color=cut))+
    stat_summary(fun.y=mean,geom="point")+
    stat_summary(fun.y=mean,geom="line")
```



エラーバーだってあっという間

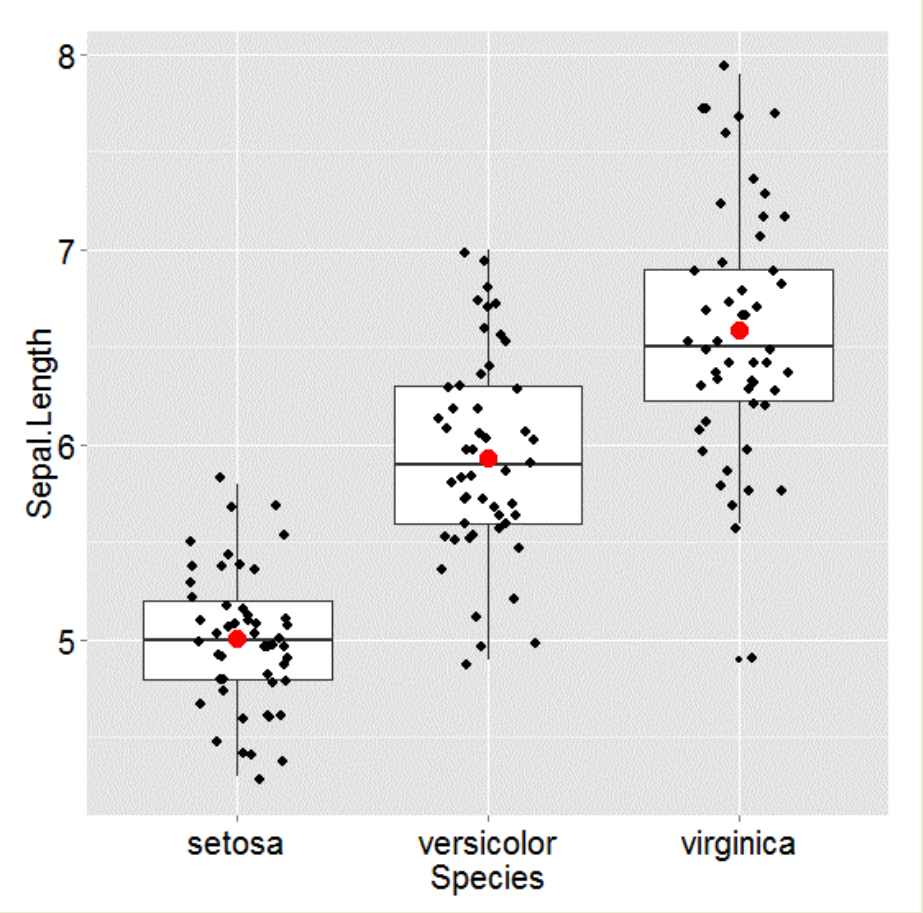


53940行×10列のデータも,
　数行でここまで仕上げることが出来る

代表値だけでなく，各計測値も重畳可能

```
ggplot(data=iris,aes(y=Sepal.Length,x=Species))+
        geom_boxplot()+
        stat_summary(fun.y=mean,geom="point",color="red")+
        geom_jitter(position=position_jitter())
```

- 箱ひげ図
  ＋
- 平均値
  ＋
- 各計測値



Let’s enjoy ggplot2 !