

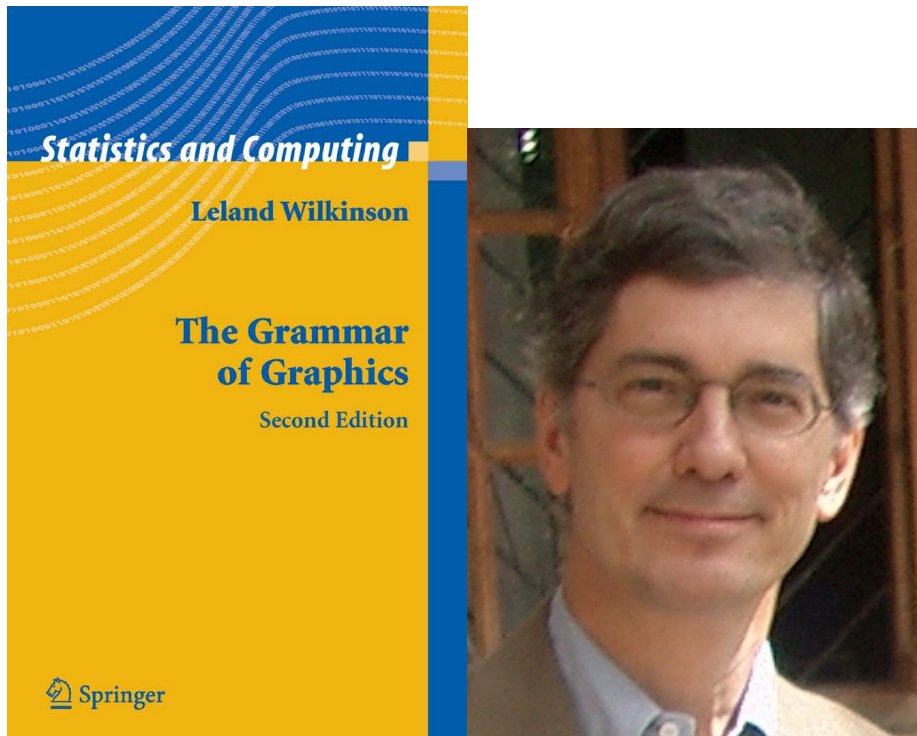
# R graphics with ggplot2

Keon-Woong Moon

2015/10/14 (updated: 2018-12-25)

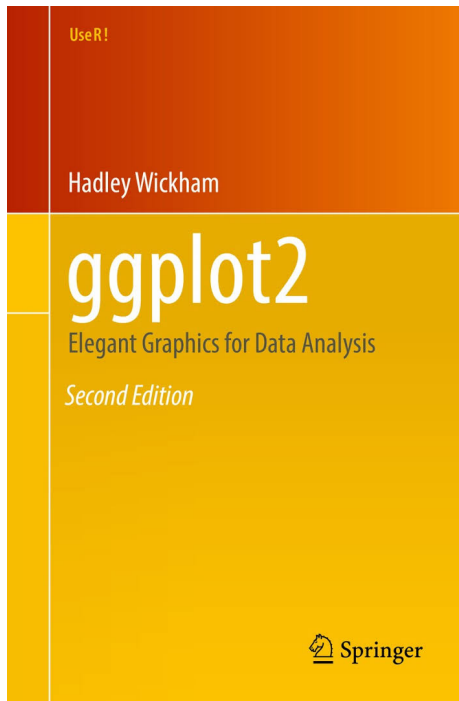
# The Grammar of Graphics

- 그래픽스 문법
- Leland Wilkinson(2005)
- Adjunctive Professor of Computer Science, University of Illinois at Chicago



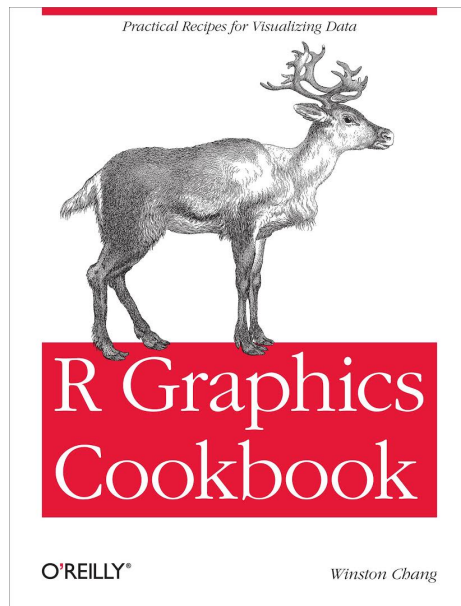
# ggplot2

- 그래픽스문법에 따라 `plot`을 그리는 R 패키지
- Hadley Wickham(2009)
- Adjunctive Professor of Statistics at Rice University



# R Graphics Cookbook

- ggplot2 예제 및 설명서
- Winston Chang(2012)
- software engineer at RStudio



# Learn ggplot2 Using Shiny App

- Shiny app으로 ggplot2를 만들수 있는 책
- Keon-Woong Moon(2017)
- Professor of Cardiology at Catholic University of Korea



# 필요한 패키지들

```
install.packages(c("ggplot2", "car", "gcookbook", "lattice"))
```

# 예제1



# 그래프의 구성

1. 데이터(data):
2. 좌표계(coordinate system):
3. 형태(geoms):
4. 미적 특징(aesthetics) :
5. 척도(scale):
6. 통계(stats):
7. 분할(facets)



- 데이터(data):
    - ggplot에서는 R의 데이터프레임(data.frame)만 사용 가능하다.
  - 좌표계(coordinate system):
    - 좌표계는 데이터가 투영되는 2차원 공간을 말하는 것
    - 예를 들어 Cartesian 좌표계(디폴트), polar 좌표계, map projection 등이 있다.
  - 형태(geoms):
    - data를 나타내는 기하학적인 형태
    - 예를 들어 점(point), 선(line), 면(area), 다각형(polygon) 등이 있다.
  - 미적 특징(aesthetics) :
    - 데이터의 시각적 특징을 나타내는 것
    - 예를 들어 위치, 크기, 색, 투명도 등이 있다.
-

- 척도(scale):
  - 데이터의 미적 특징을 수치화하는 척도
  - 예를 들어 로그척도, 색척도, 크기척도 등이 있다.
- 통계(stats):
  - 데이터의 요약에 사용되는 통계학적인 변형
  - 예를 들어 개수, 평균, 중앙값, 회귀선 등이 있다.
- 분할(facets)
  - 데이터를 여러 개의 부분집합으로 나누고 작은 여러 개의 그래프로 분할하여 그리는 것

# 그래프를 그리는 순서

## 1. 데이터 할당(data):

- 데이터 프레임만 가능, 예: `data=Salaries`

## 2. 변수 할당 또는 설정(aes)

- x축 변수: 반드시 필요하다.
- y축 변수: 경우에 따라 필요하다(히스토그램, 밀도 곡선 등은 x축 변수만 지정하여 그릴 수 있으며 산점도 등에는 x축 변수와 함께 y축 변수가 필요하다).
- `colour`, `fill`, `size` 등에도 변수를 할당(예: `colour = sex`)하거나 설정(예: `colour = "black"`)할 수도 있다.
- 영국식 철자 사용

## 3. 형태 설정(geom):

- 점(`point`), 선(`line`), 면(`area`), 다각형(`polygon`) 등
- 여러 형태를 `layer by layer`로 선택할 수 있다.

## 4. 기타 :

- 좌표계와 척도 등은 기본값이 있으므로 따로 변경할 필요가 있는 경우를 제외하고는 설정해주지 않아도 그래프를 그릴 수 있다.
- 필요에 따라 좌표계/척도의 설정을 변경, 통계 추가, 면 분할 등을 추가

# 첫번째 예

## Salaries 데이터

```
require(ggplot2)      # ggplot()을 사용하기 위해  
require(car)          # Salaries 데이터를 사용하기 위해  
  
str(Salaries)         # Salaries 데이터의 구조는?
```

```
'data.frame':      397 obs. of  6 variables:  
 $ rank           : Factor w/ 3 levels "AsstProf","AssocProf",...: 3 3 1 3 3 2 3  
 $ discipline      : Factor w/ 2 levels "A","B": 2 2 2 2 2 2 2 2 2 2 ...  
 $ yrs.since.phd   : int   19 20 4 45 40 6 30 45 21 18 ...  
 $ yrs.service     : int   18 16 3 39 41 6 23 45 20 18 ...  
 $ sex             : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 1 ...  
 $ salary          : int  139750 173200 79750 115000 141500 97000 175000 147765
```

```
?Salaries           # 도움말 보기
```

# 데이터 할당

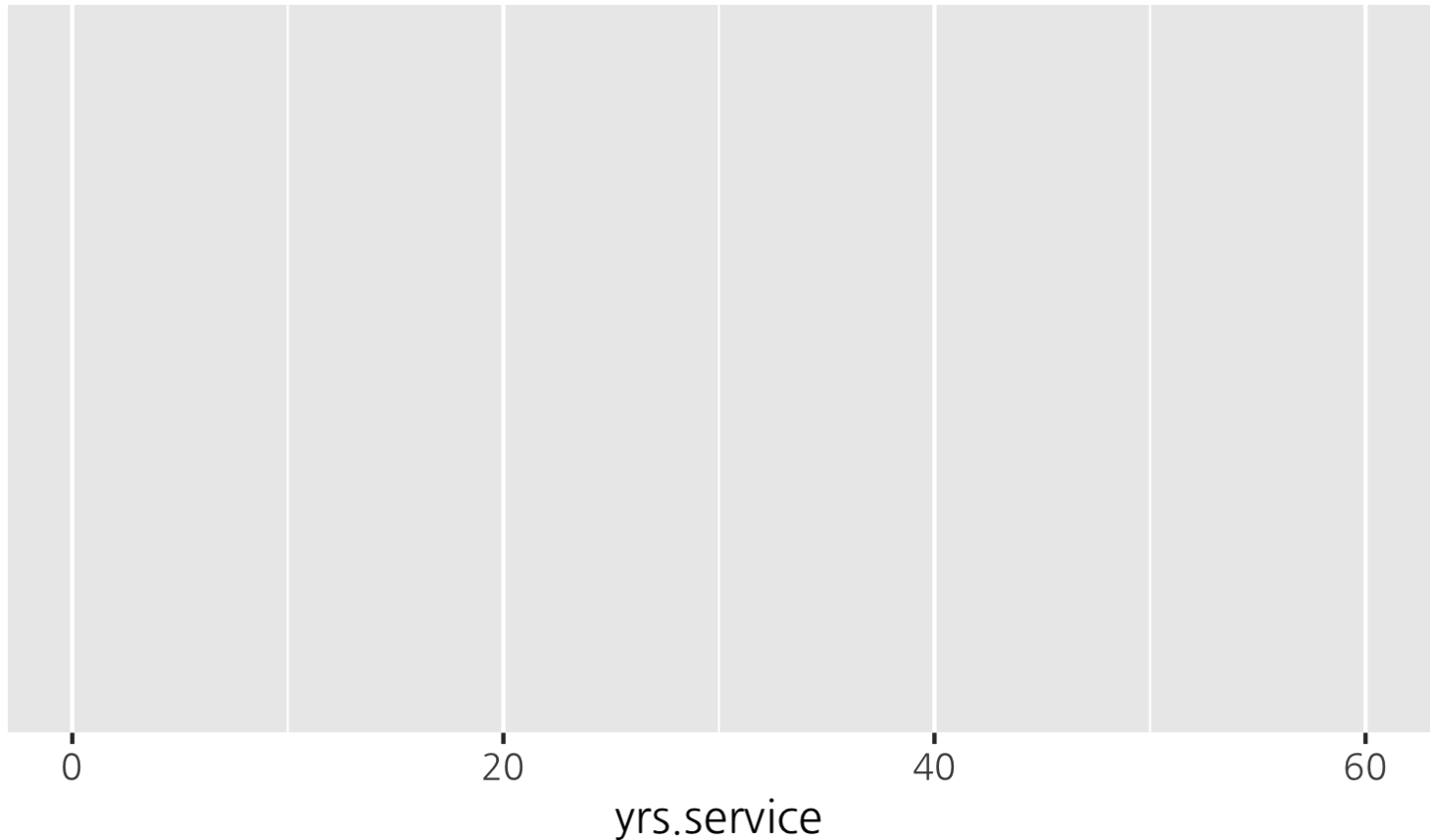
```
ggplot(data=Salaries)
```

```
# 데이터 할당
```

# 데이터 및 x축 변수 할당

```
ggplot(data=Salaries,  
       aes(x=yrs.service))
```

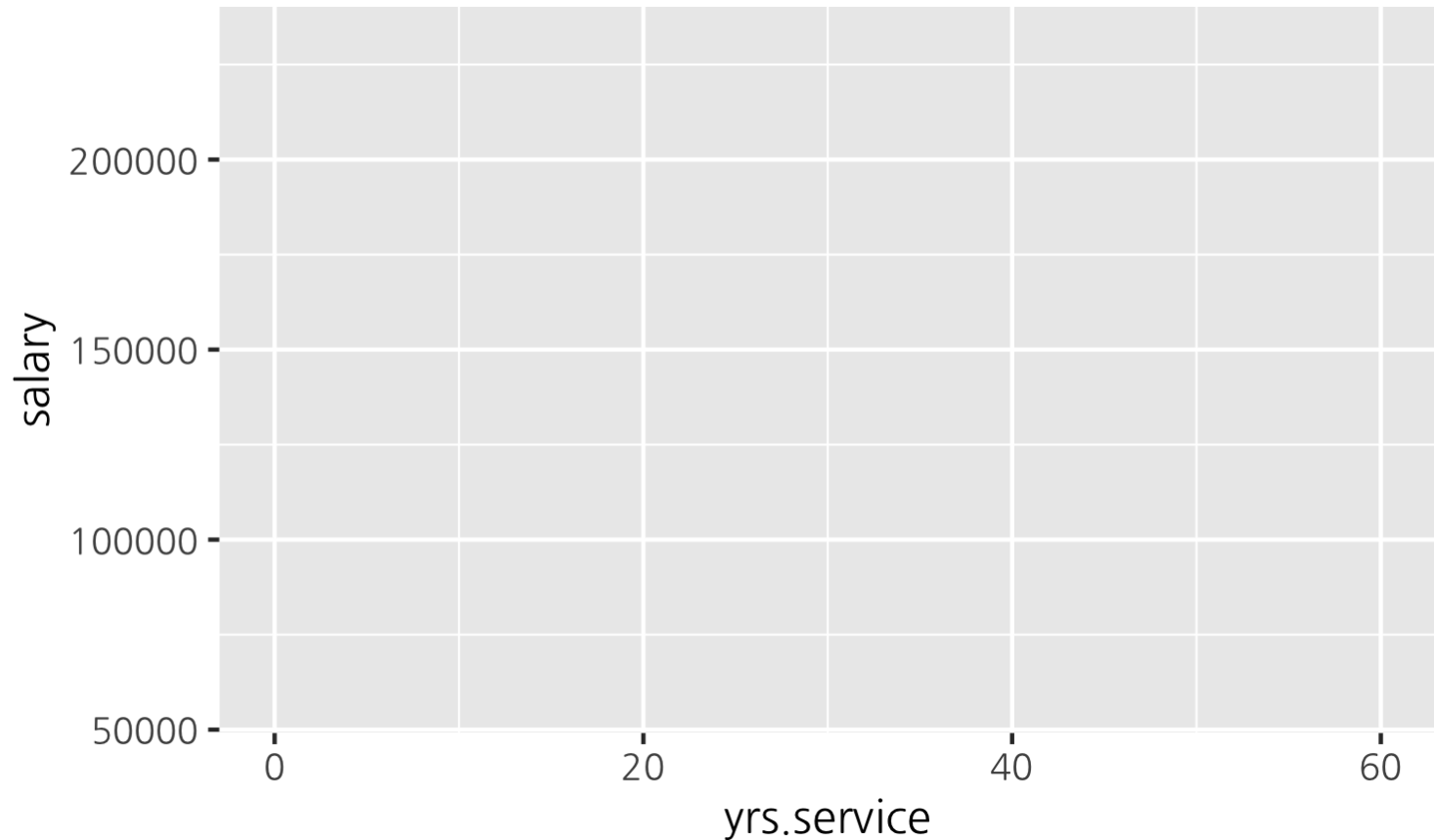
# 데이터 할당  
# 변수 할당



## 데이터 및 x축, y축변수 할당

```
ggplot(data=Salaries,  
       aes(x=yrs.service,y=salary))
```

# 데이터 할당  
# 변수 할당



## 점그래프 추가

```
p <- ggplot(data=Salaries,  
            aes(x=yrs.service,y=salary))  
p + geom_point()
```

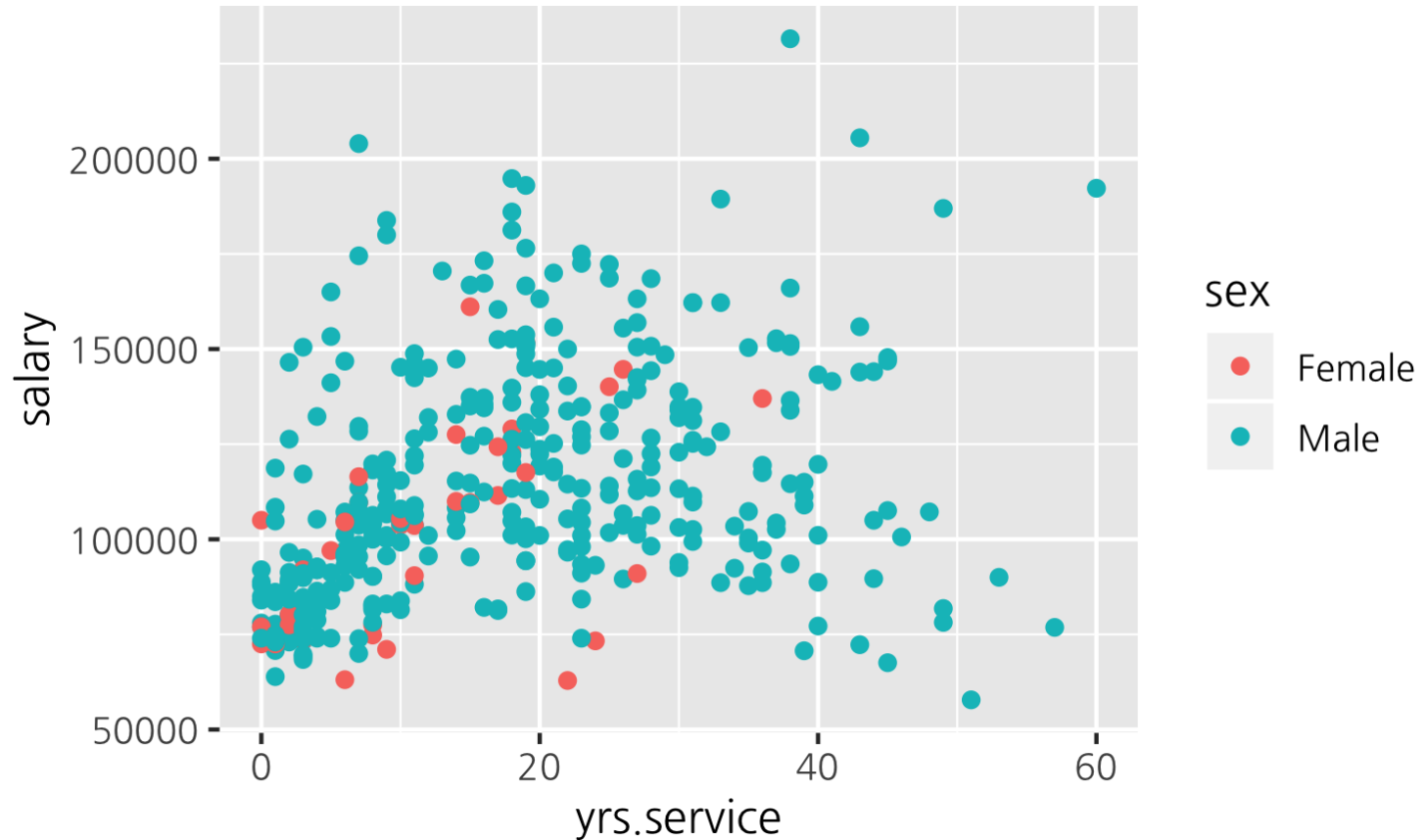
# 데이터 할당  
# 변수 할당



# 성별에 따른 색깔구분

```
p + geom_point(aes(colour=sex))
```

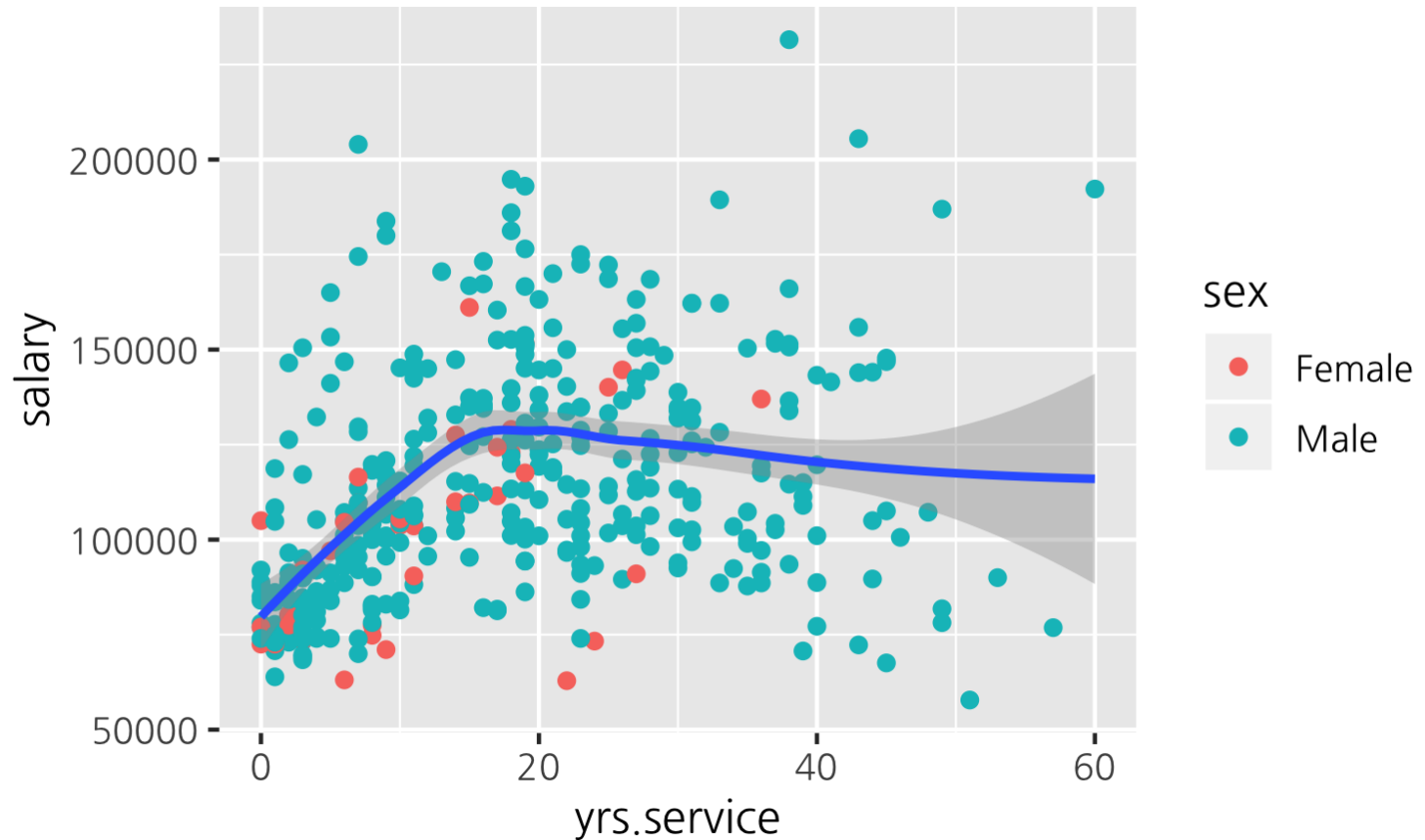
# 점그래프의 색에 성별을 할당



## 통계추가: loess회귀선 추가

```
p + geom_point(aes(colour=sex)) +  
  geom_smooth()
```

# 점그래프의 색에 성별을 할당  
# 회귀선 추가



## 면분할: 성별로 면분할

```
p + geom_point(aes(colour=sex)) +  
  geom_smooth() +  
  facet_grid(~sex)
```

```
# 점그래프의 색에 성별을 할당  
# 회귀선 추가  
# 면을 수직으로 분할
```

```
p <- ggplot(data=Salaries,
            aes(x=yrs.service,y=salary,fill=sex))
p + geom_point(pch=21) +
  geom_smooth(method="lm",formula=y~poly(x,2)) +
  facet_grid(~sex)
```



























# 데이터 할당  
# 변수할당  
# 점그래프  
# 회귀선 추가  
# 면을 수직으로 분할



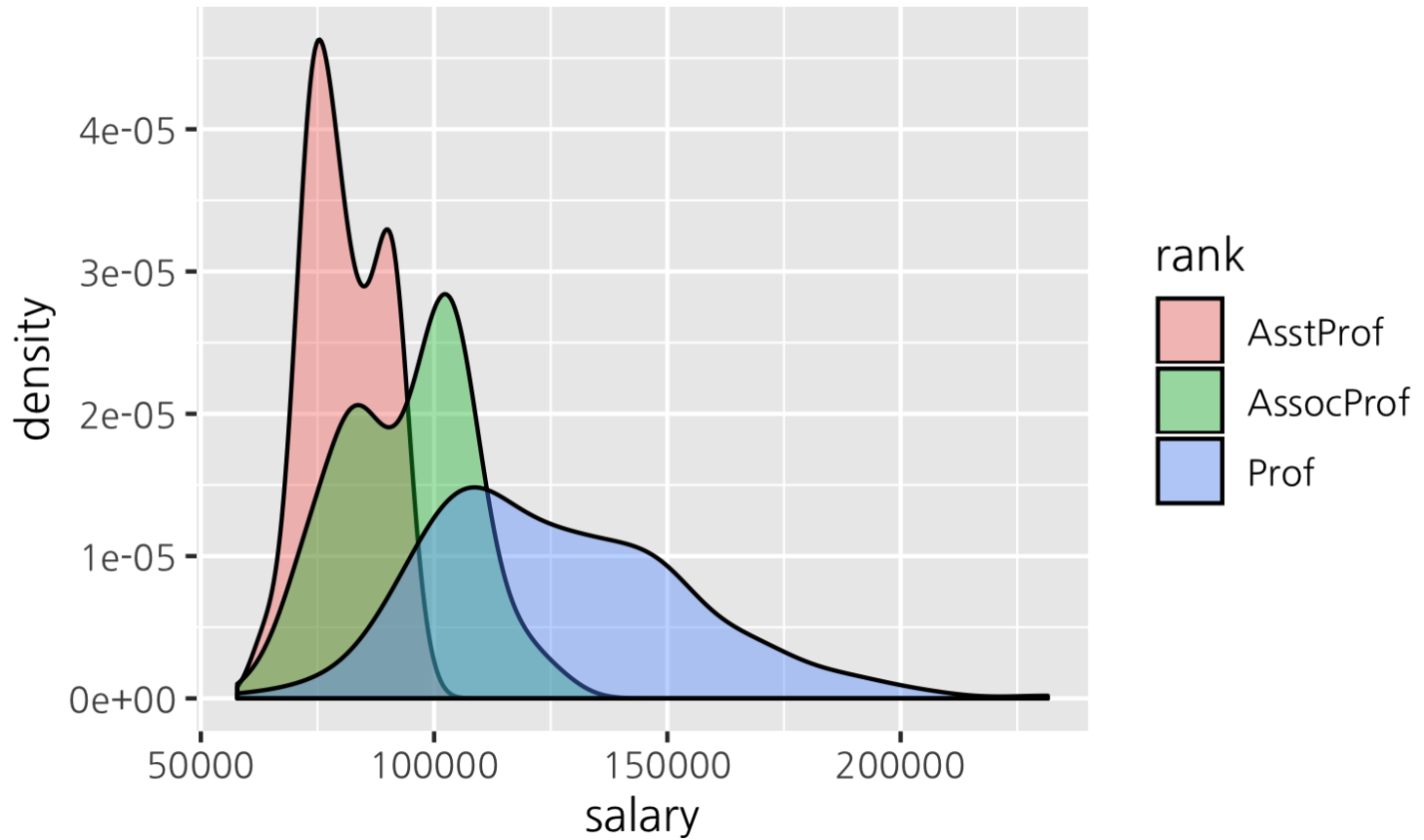
# 점의 형태

```
x=0:48
pch=c(0:25,65:73,91:104)
shape=factor(pch)
mypoints=data.frame(x=x,pch=pch,shape=shape)
ggplot(data=mypoints,aes(x=floor(x/7),y=x%%7))+
  geom_point(aes(shape=shape),size=4,color="red")+
  scale_shape_manual(values=pch)+
  geom_text(label=pch,vjust=-1.1,size=4)+
  theme(legend.position="none")+
  labs(title="Demonstration of point shape",x="",y="")+
  scale_y_reverse()+
  expand_limits(y=-0.2,ymax=6.2)
```

# Demonstration of point shape

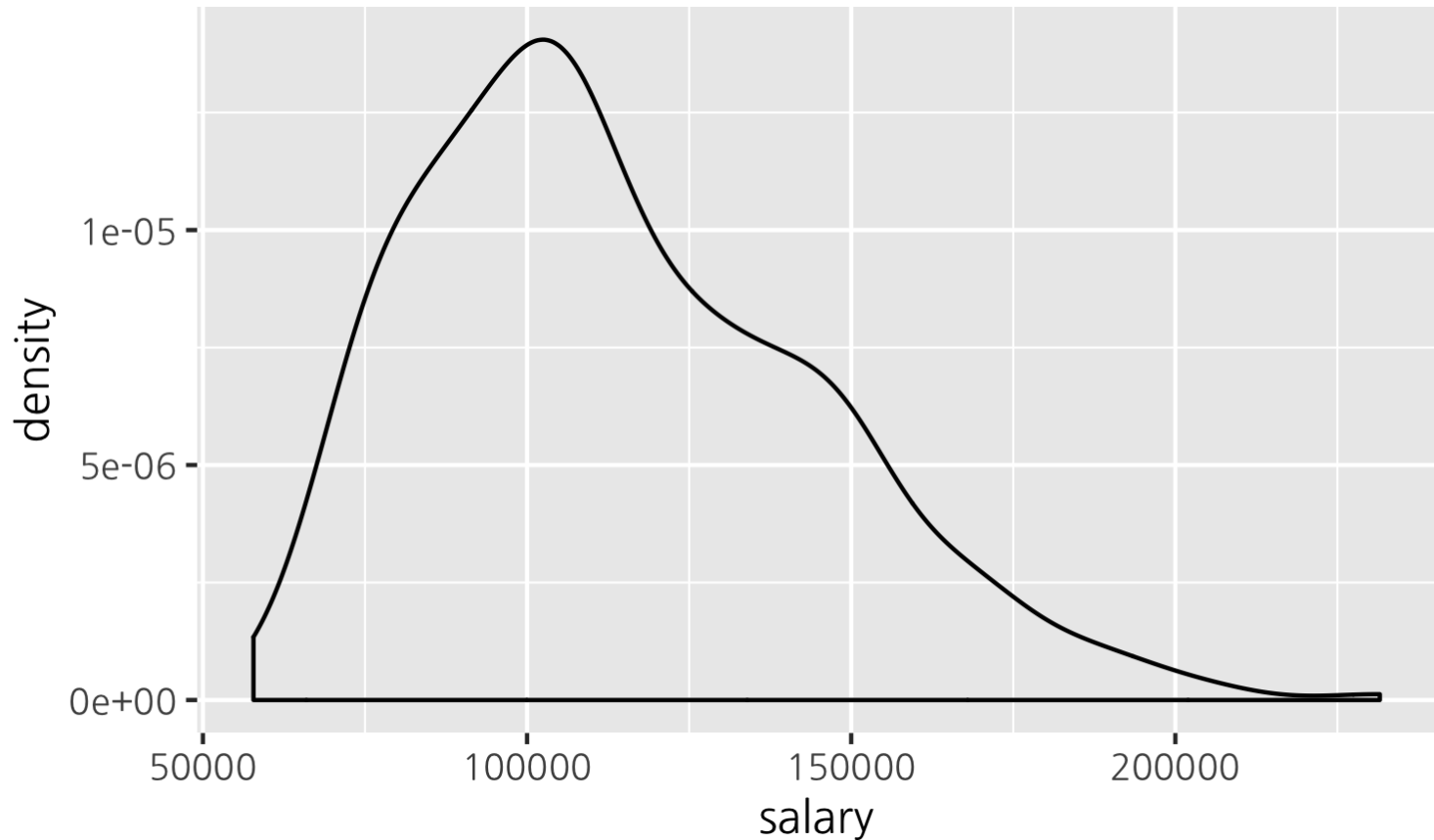
|   |   |   |   |   |    |    |     |
|---|---|---|---|---|----|----|-----|
|   | 0   | 7   | 14  | 21  | 67 | 91 | 98  |
| 0 |    |    |    |  | C  | [  | b   |
|   | 1   | 8   | 15  | 22  | 68 | 92 | 99  |
|   |    |    |    |  | D  | \  | c   |
|   | 2   | 9   | 16  | 23  | 69 | 93 | 100 |
| 2 |    |    |    |  | E  | ]  | d   |
|   | 3   | 10  | 17  | 24  | 70 | 94 | 101 |
|   |    |    |    |  | F  | ^  | e   |
|   | 4   | 11  | 18  | 25  | 71 | 95 | 102 |
| 4 |    |    |    |  | G  | —  | f   |
|   | 5   | 12  | 19  | 65  | 72 | 96 | 103 |
|   |    |    |    | A   | H  | `  | g   |
|   | 6   | 13  | 20  | 66  | 73 | 97 | 104 |
| 6 |  |  |  | B   | I  | a  | h   |
|   | 0   |   | 2   |   | 4  |    | 6   |

## 예제 2: 대학교수 직급별 연봉분포



# 대학교수의 연봉분포

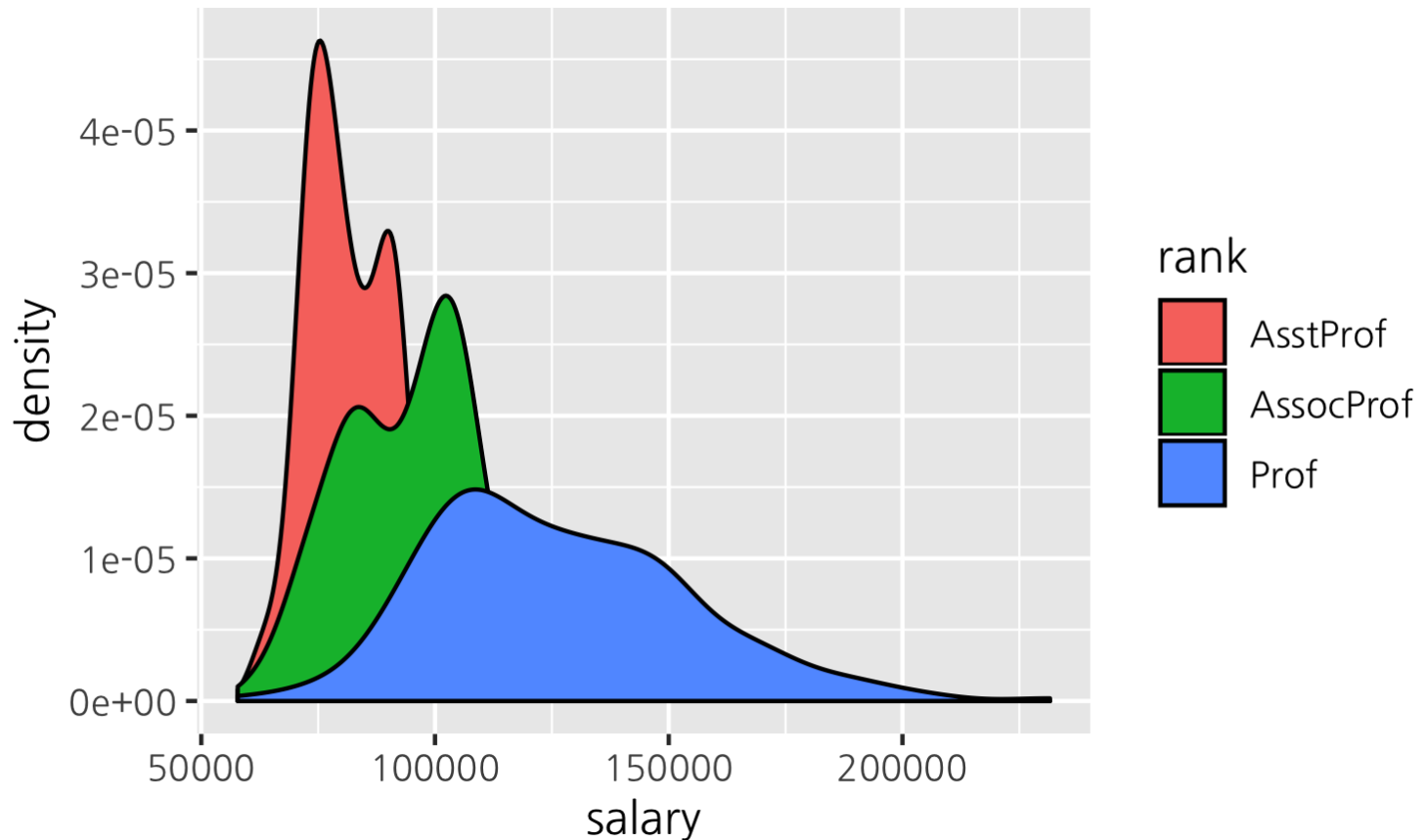
```
p<-ggplot(data=Salaries,aes(x=salary)) # 데이터, 변수 할당  
p+ geom_density() # density 추가, 투명도 조절
```





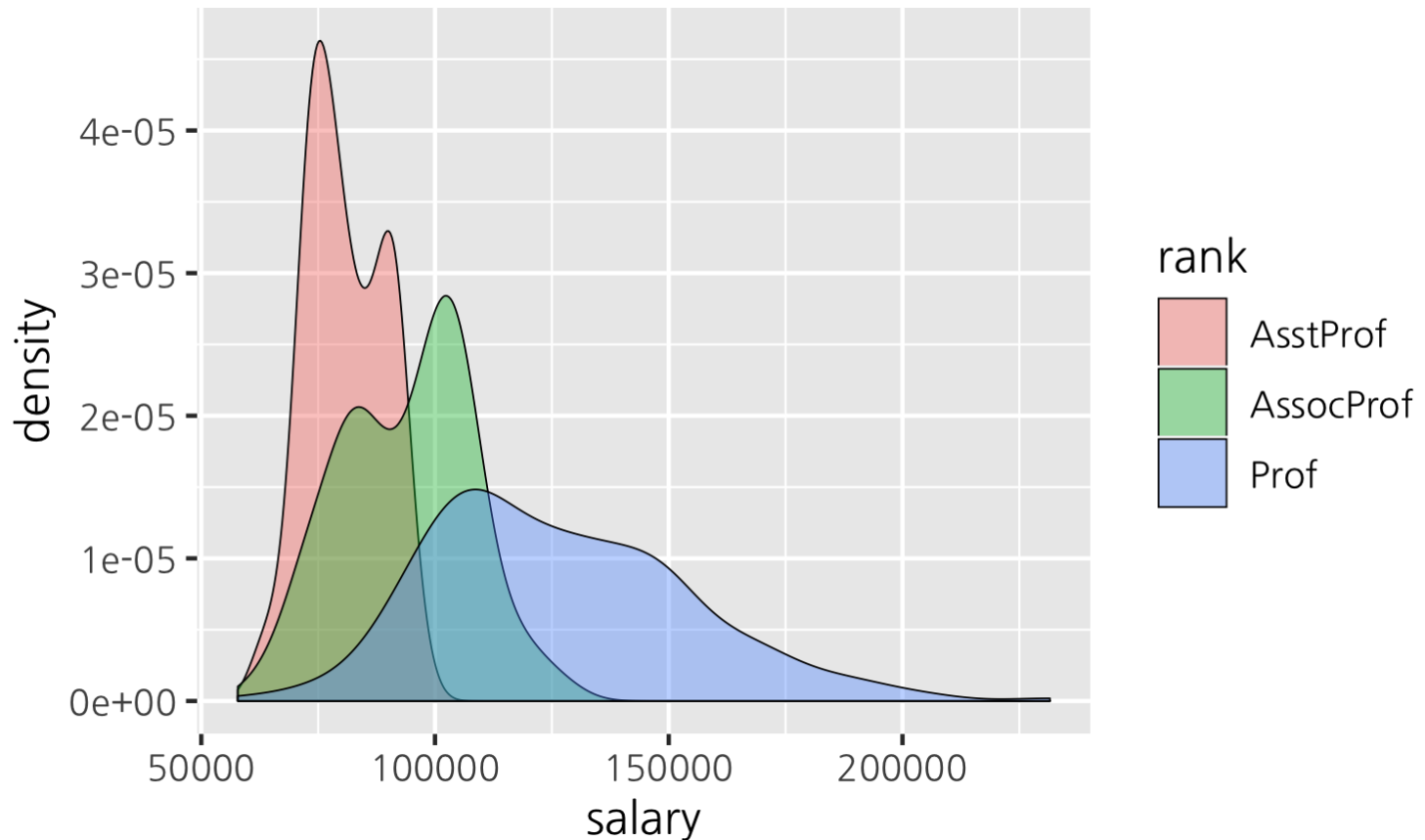
# 대학교수의 직급별 연봉분포

```
p<-ggplot(data=Salaries,aes(x=salary,fill=rank)) # 데이터, 변수 할당  
p+ geom_density() # density 추가, 투명도 조절
```



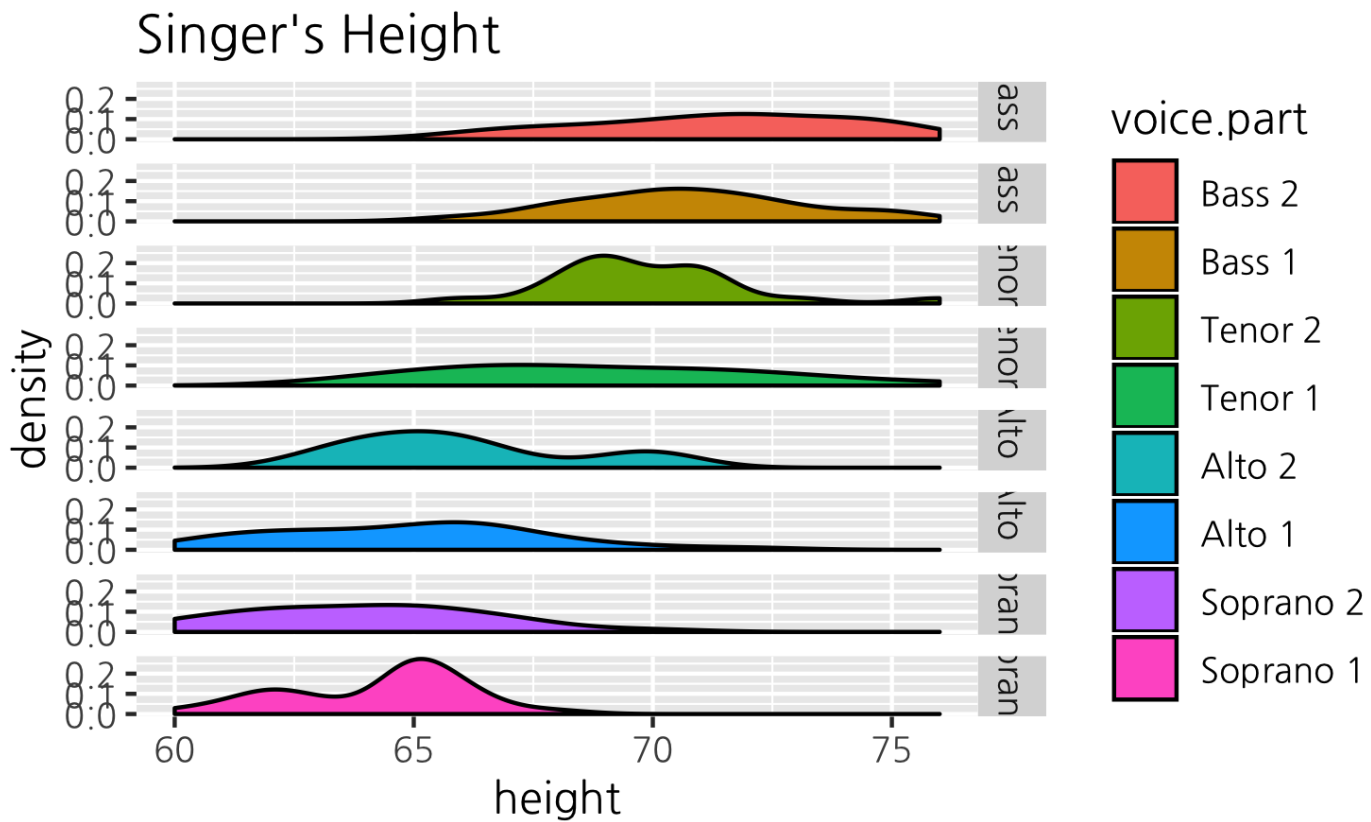
## 투명도, 선굵기 조절

```
ggplot(data=Salaries,aes(x=salary,fill=rank)) + # 데이터, 변수 할당  
geom_density(alpha=0.4,size=0.2)             # density 추가, 투명도 조절
```



# 가수의 키 분포

- 사용데이터 `lattice::singer`



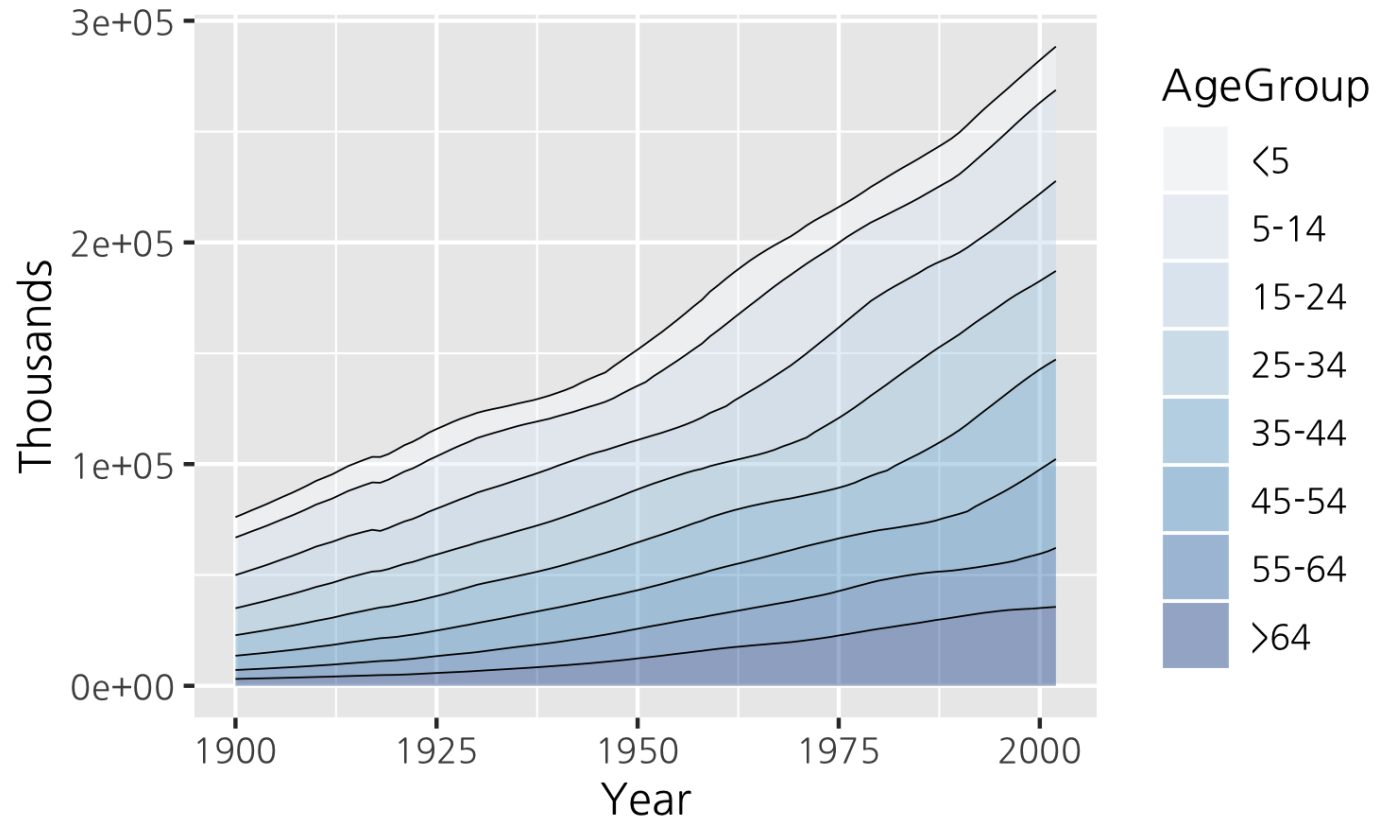
```
str(lattice::singer)
```

```
'data.frame':   235 obs. of  2 variables:  
 $ height      : num  64 62 66 65 60 61 65 66 65 63 ...  
 $ voice.part: Factor w/ 8 levels "Bass 2","Bass 1",...: 8 8 8 8 8 8 8 8 8 8 .
```

```
ggplot(data=lattice::singer,aes(x=height,fill=voice.part))+  
  geom_density()+  
  facet_grid(voice.part ~ .)+  
  labs(title="Singer's Height")
```

# Area Plot

- 사용데이터 `gcookbook::uspopage`

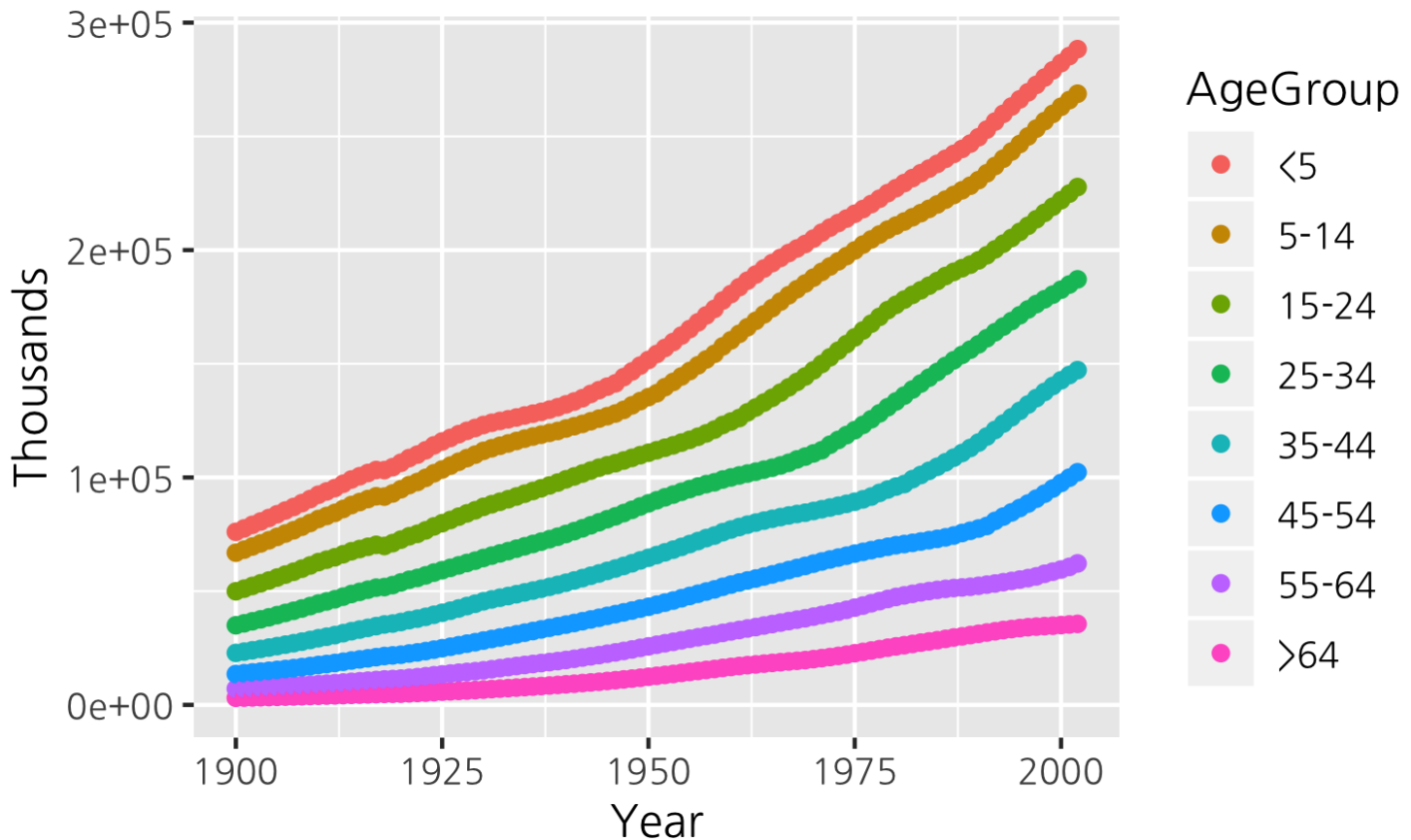


## 3-1 점그래프

```
require(gcookbook)
ggplot(uspopage,aes(x=Year,y=Thousands,color=AgeGroup))+
  geom_point()
```

## 3-2 점그래프 위치변경

```
ggplot(uspopage, aes(x=Year, y=Thousands, color=AgeGroup)) +  
  geom_point(position=position_stack())
```



### 3-3 점그래프 + 선그래프

```
ggplot(uspopage, aes(x=Year, y=Thousands, color=AgeGroup)) +  
  geom_point(position=position_stack()) +  
  geom_line()
```

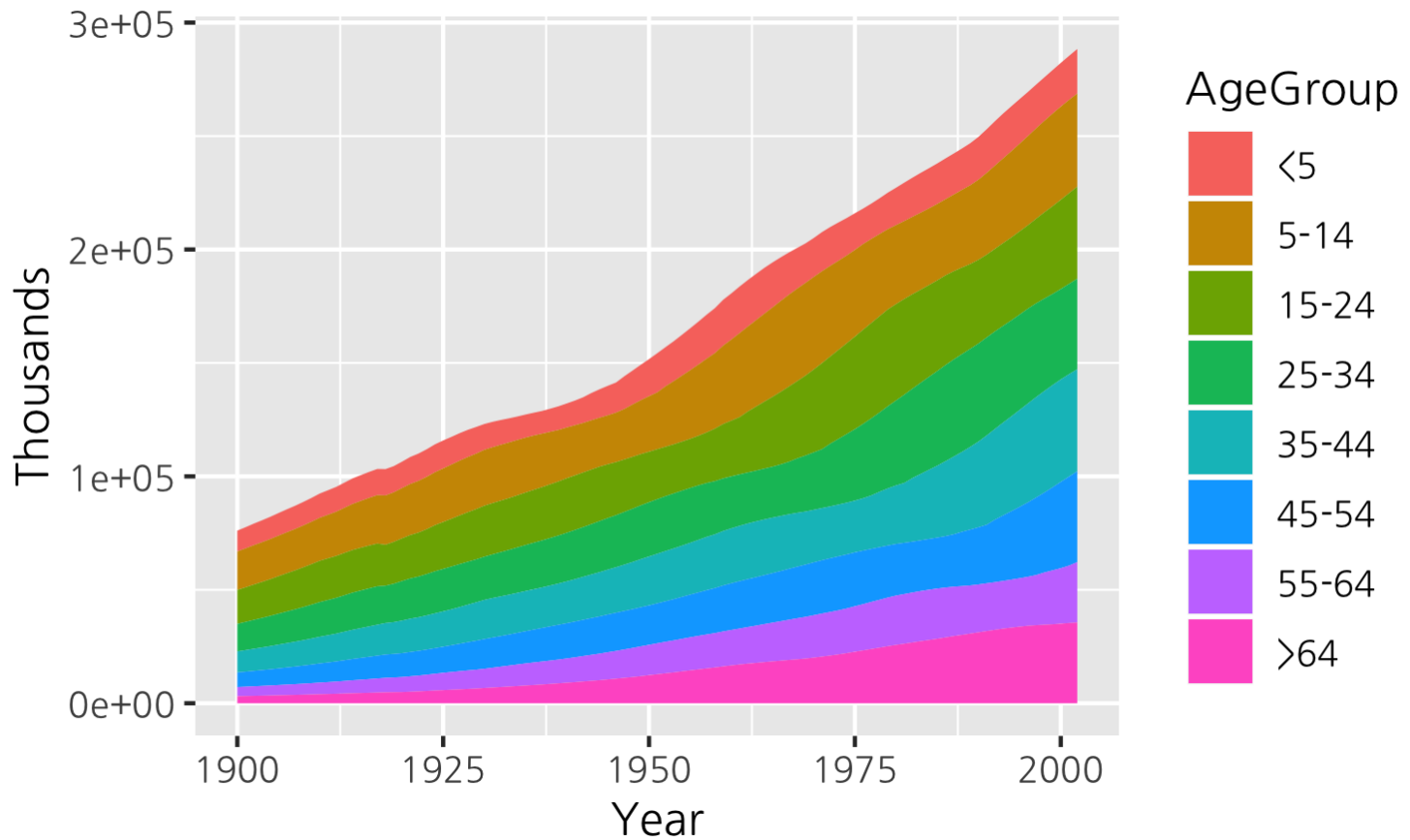


## 3-4 점그래프 + 선그래프 위치변경

```
ggplot(uspopage,aes(x=Year,y=Thousands,color=AgeGroup))+  
  geom_point(position=position_stack())+  
  geom_line(position=position_stack())
```

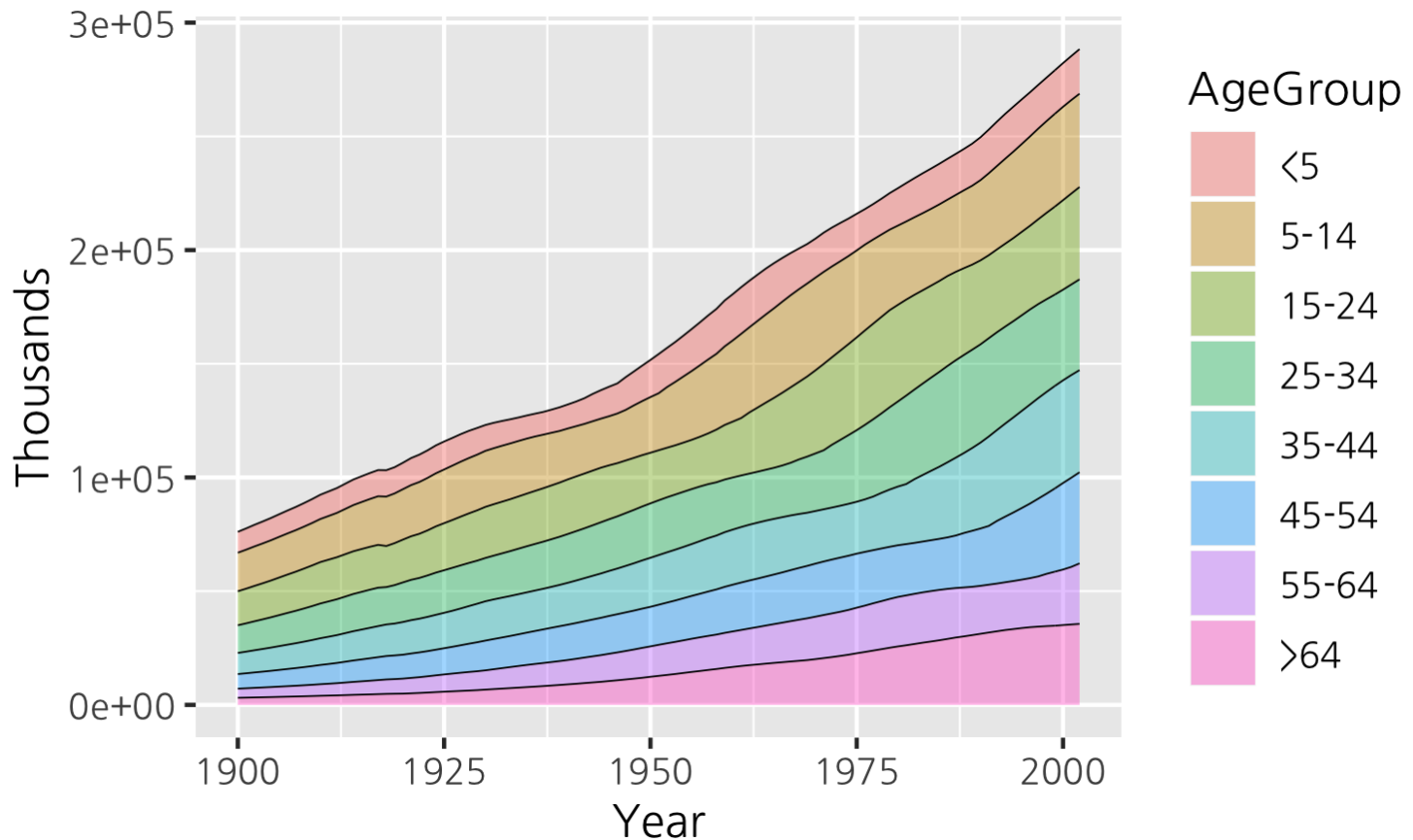
### 3-5 영역그래프

```
p <- ggplot(uspopage, aes(x=Year, y=Thousands, fill=AgeGroup))  
p + geom_area()
```



## 3-6 투명도 조절 + 선추가

```
p + geom_area(alpha=0.4) +  
  geom_line(position='stack',size=0.2)
```

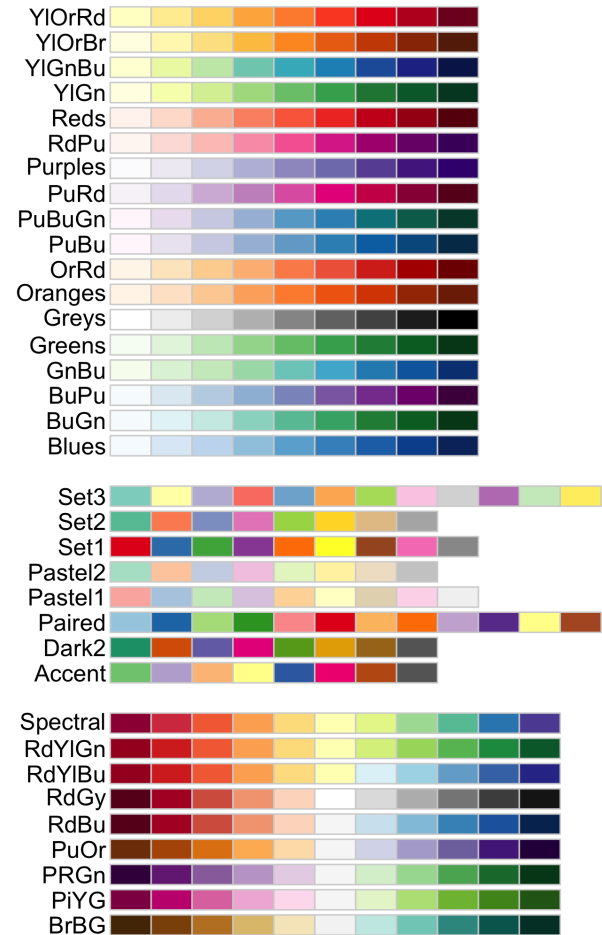


## 3-7 팔레트적용

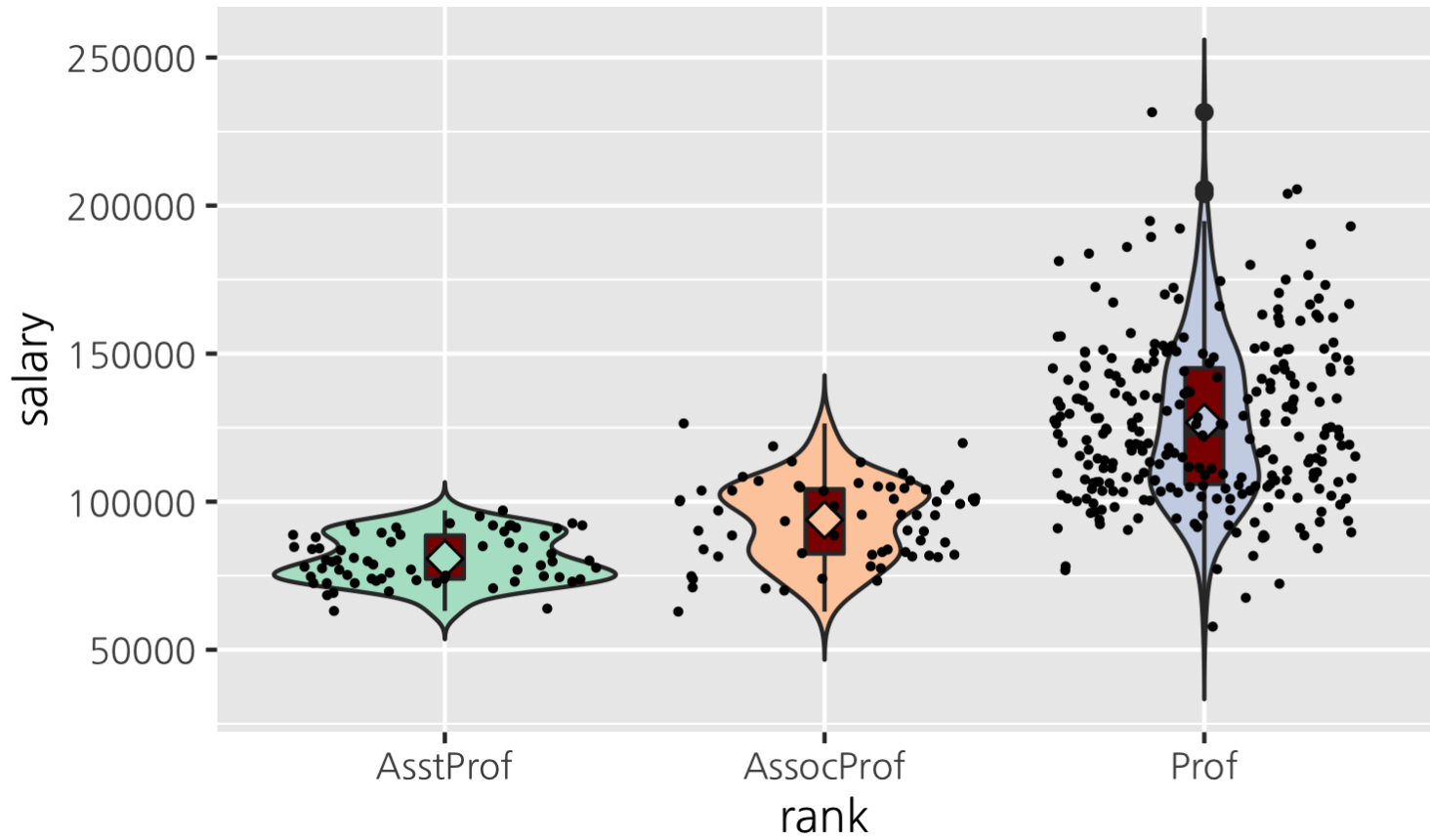
```
p + geom_area(alpha=0.4) +  
    geom_line(position='stack',size=0.2) +  
    scale_fill_brewer(palette="Blues")
```

# R 에서 팔레트 사용

```
library(RColorBrewer)  
display.brewer.all()
```

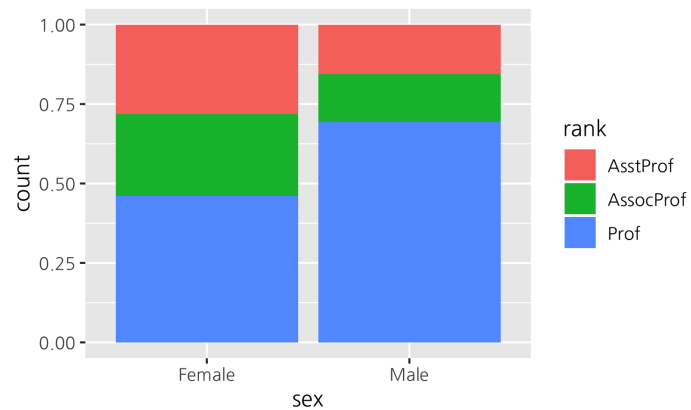
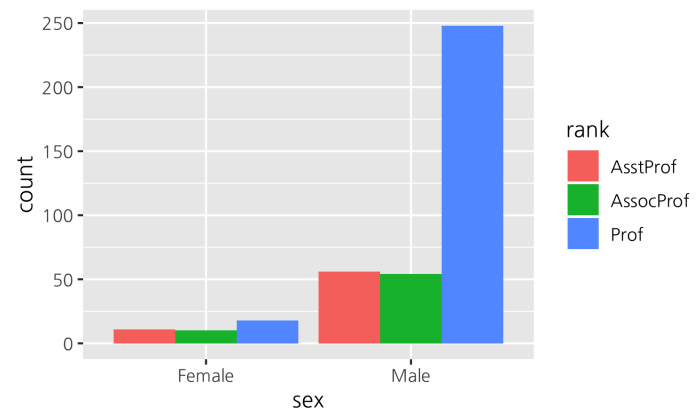
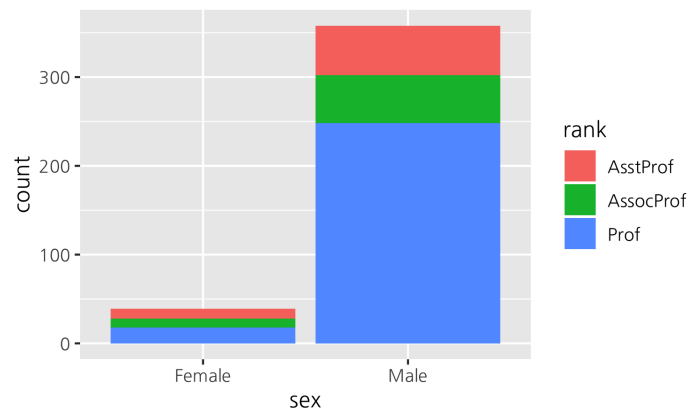


## 예제 4: 직급별 연봉



```
ggplot(data=Salaries,aes(x=rank,y=salary,fill=rank))+  
  geom_violin(trim=FALSE)+  
  geom_boxplot(fill='darkred',width=0.1)+  
  stat_summary(geom='point',fun.y=mean,shape=23,size=3)+  
  geom_point(position='jitter',size=0.5)+  
  scale_fill_brewer(palette='Pastel2')+  
  theme(legend.position='none')
```

## 예제 5. 막대그래프

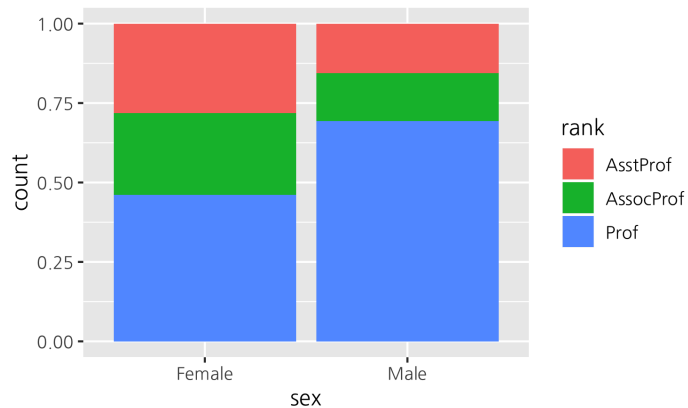




# 막대의 위치

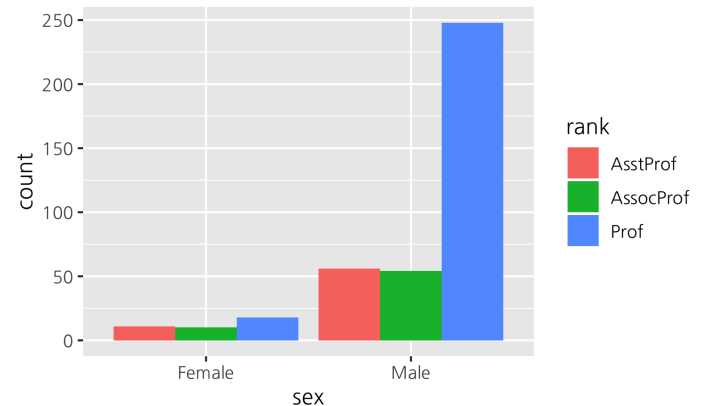
**position = "fill"**

```
ggplot(data=Salaries,
       aes(x=sex, fill=rank)) +
  geom_bar(position="fill")
```

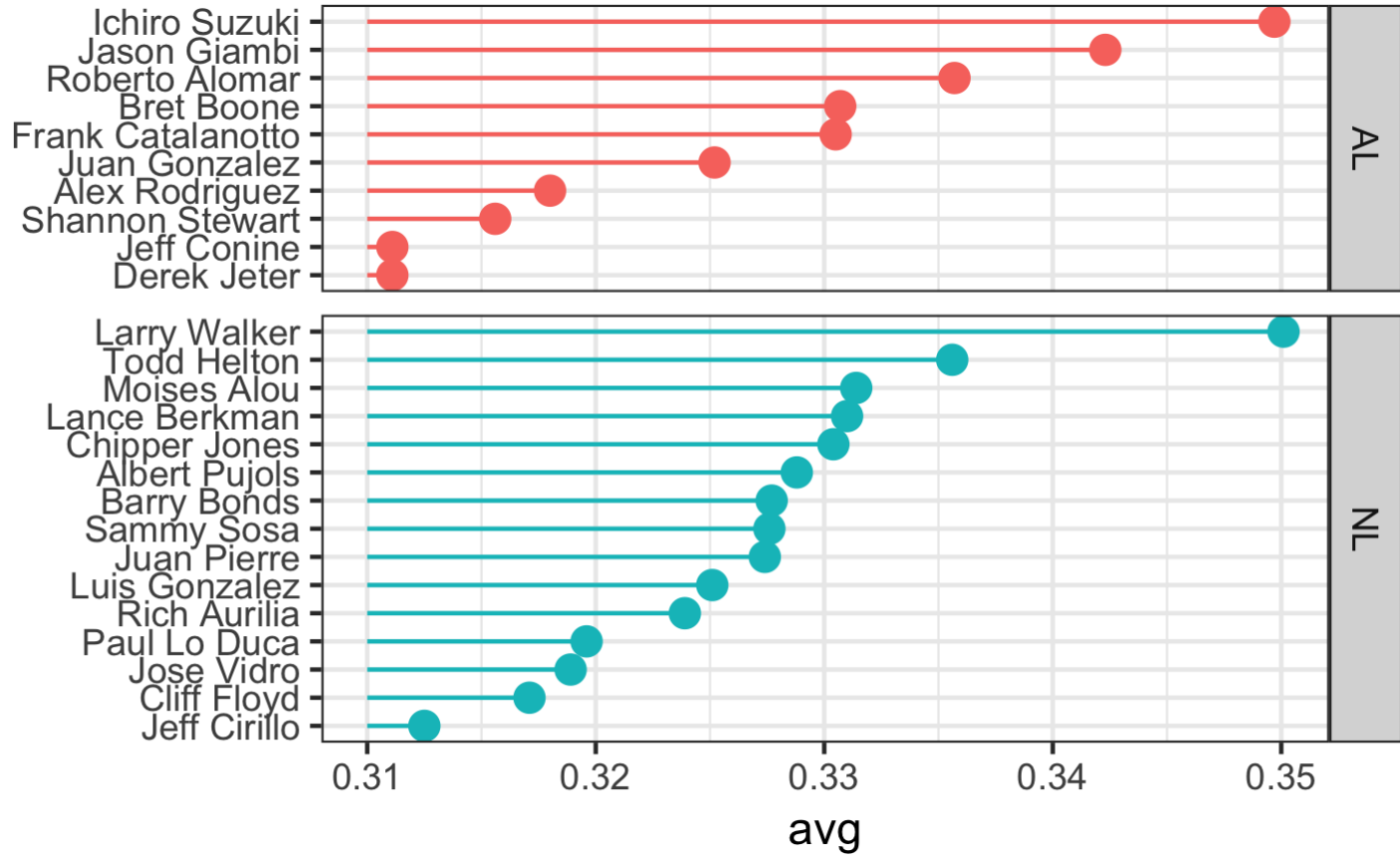


**position = "dodge"**

```
ggplot(data=Salaries,
       aes(x=sex, fill=rank)) +
  geom_bar(position="dodge")
```



## 예제 6 클리브랜드 dot plot



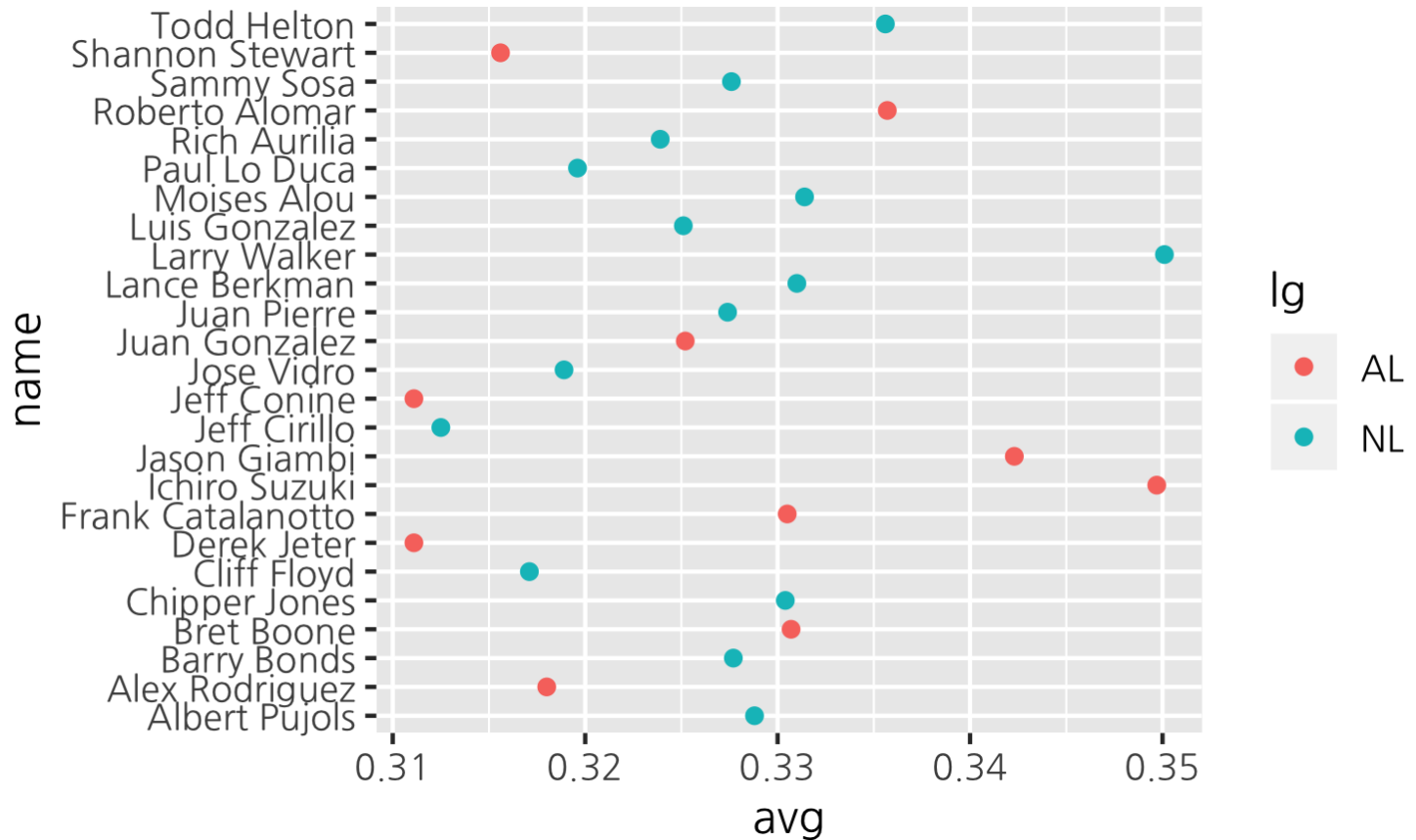
- 사용데이터 gcookbook::tophitters2001

```
tophit=tophitters2001[1:25,]  
str(tophit)
```

```
'data.frame':    25 obs. of  26 variables:  
 $ id   : Factor w/ 144 levels "abreubo01","alfoned01",...: 138 128 41 3 59 4  
 $ first: chr  "Larry" "Ichiro" "Jason" "Roberto" ...  
 $ last : chr  "Walker" "Suzuki" "Giambi" "Alomar" ...  
 $ name : chr  "Larry Walker" "Ichiro Suzuki" "Jason Giambi" "Roberto Alomar"  
 $ year : int   2001 2001 2001 2001 2001 2001 2001 2001 2001 2001 2001 ...  
 $ stint: int    1 1 1 1 1 1 1 1 1 1 1 ...  
 $ team : Factor w/ 30 levels "ANA","ARI","ATL",...: 10 25 21 9 10 13 13 25 29  
 $ lg   : Factor w/ 2 levels "AL","NL": 2 1 1 1 2 2 2 1 1 2 ...  
 $ g    : int   142 157 154 157 159 136 156 158 133 159 ...  
 $ ab   : int   497 692 520 575 587 513 577 623 463 572 ...  
 $ r    : int   107 127 109 113 132 79 110 118 77 113 ...  
 $ h    : int   174 242 178 193 197 170 191 206 153 189 ...  
 $ 2b   : int    35 34 47 34 54 31 55 37 31 33 ...  
 $ 3b   : int     3 8 2 12 2 1 5 3 5 5 ...  
 $ hr   : int    38 8 38 20 49 27 34 37 11 38 ...  
 $ rbi  : int   123 69 120 100 146 108 126 141 54 102 ...  
 $ sb   : int    14 56 2 30 7 5 7 5 15 9 ...  
 $ cs   : int     5 14 0 6 5 1 9 5 5 10 ...  
 $ bb   : int    82 30 129 80 98 57 92 40 39 98 ...  
 $ so   : int   103 53 83 71 104 57 121 110 55 82 ...  
 $ ibb  : int     6 10 24 5 15 14 5 5 3 20 ...
```

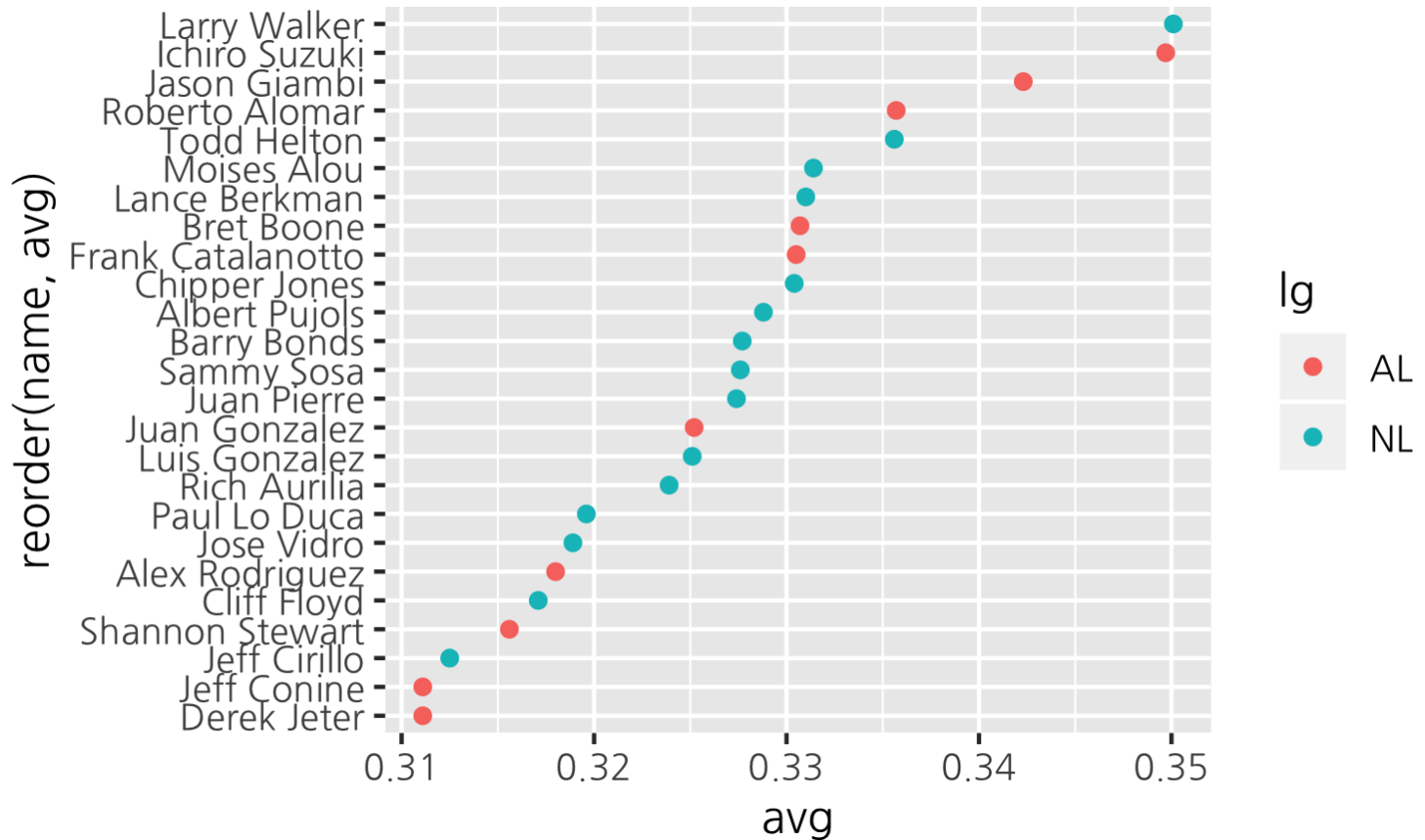
# 산점도

```
ggplot(data=tophit, aes(x=avg, y=name, colour=lg)) +  
  geom_point()
```



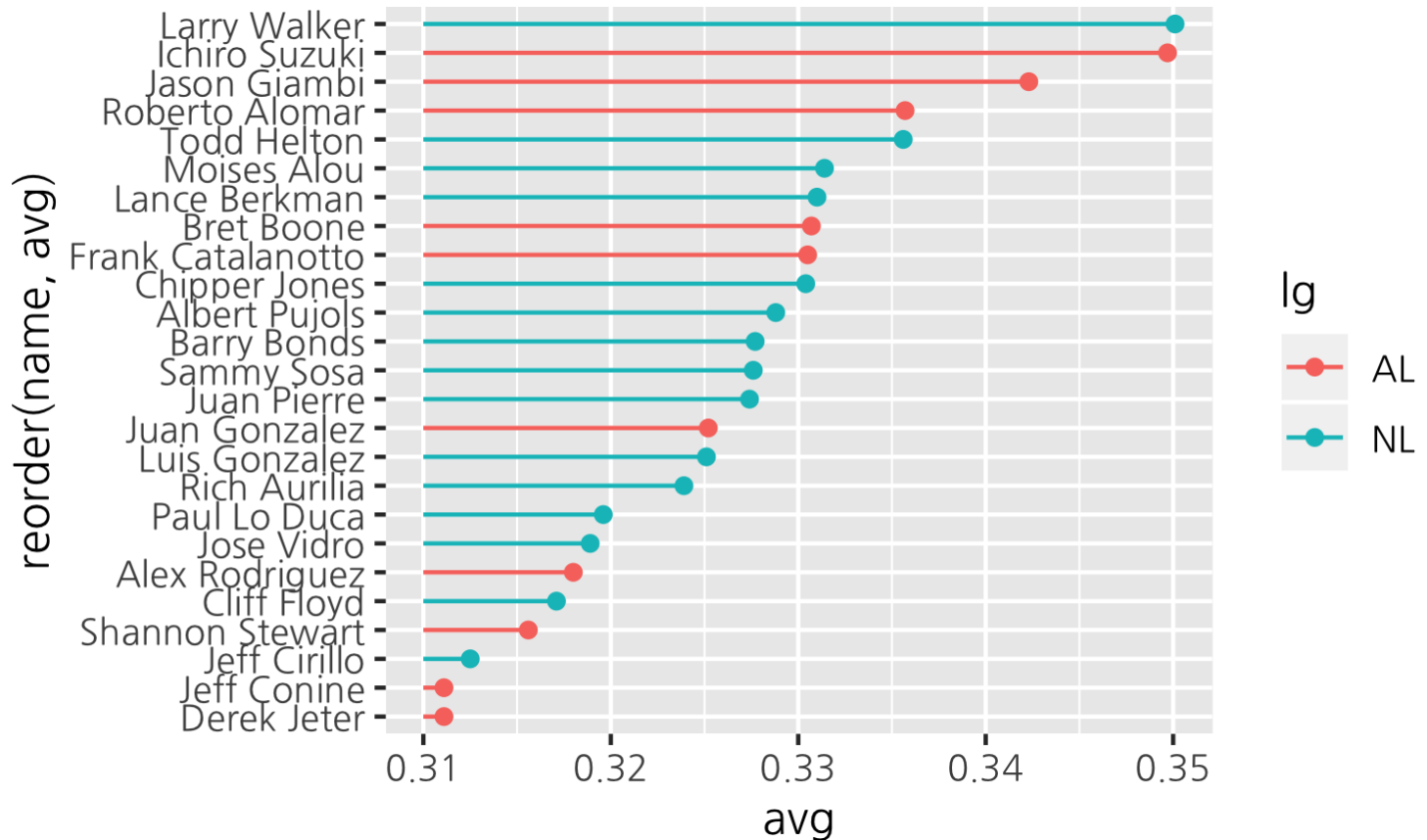
# 이름을 성적순으로

```
p <- ggplot(data=tophit,aes(x=avg,y=reorder(name,avg),colour=lg))  
p + geom_point()
```



# 산점도+선분추가

```
p + geom_point() +  
  geom_segment(aes(xend=0.31, yend=name))
```



## 산점도+선분추가+테마 적용

```
p + geom_point() +  
  geom_segment(aes(xend=0.31,yend=name)) +  
  theme_bw()
```

# 면분할

```
p + geom_point() +  
  geom_segment(aes(xend=0.31,yend=name)) +  
  theme_bw() +  
  facet_grid(lg~.)
```



## 면분할 옵션조절

```
p + geom_point() +  
  geom_segment(aes(xend=0.31,yend=name)) +  
  theme_bw() +  
  facet_grid(lg~.,scales= "free_y",space= "free_y" )
```

# 그래프 가다듬기

```
p + geom_point() +  
  geom_segment(aes(xend=0.31,yend=name)) +  
  theme_bw() +  
  facet_grid(lg~.,scales= "free_y",space= "free_y" ) +  
  theme(legend.position= "none" )+ ylab("")
```

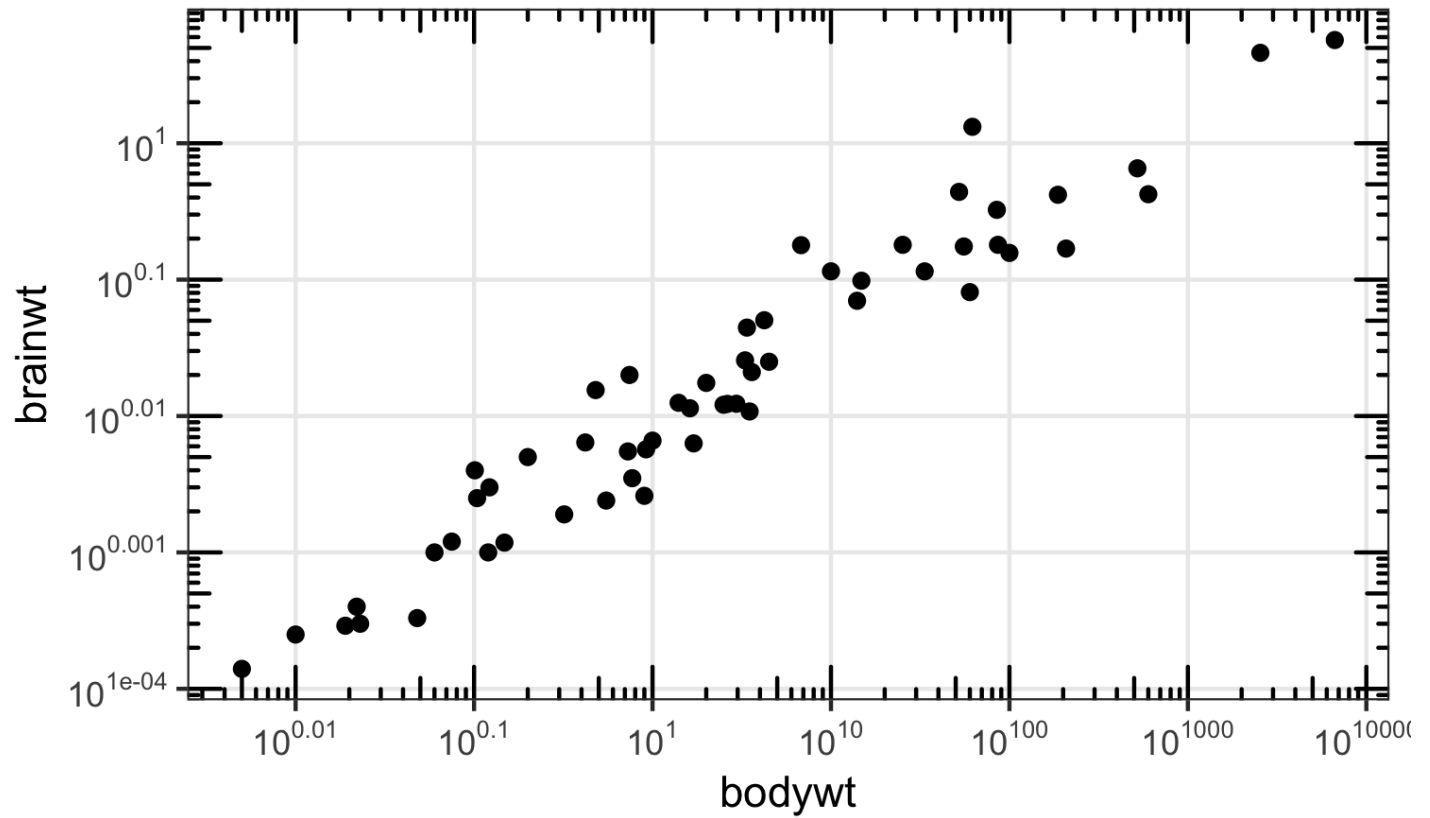
# R package ggplotAssist

- ggplot2를 배울수 있는 shiny app
- Keon-Woong Moon(2017)
- Professor of Cardiology at Catholic University of Korea

```
install.packages("ggplotAssist")
```

<https://github.com/cardiomoon/ggplotAssist>

- 데이터 ggplot2::msleep



```
ggplot(msleep,aes(x=bodywt,y=brainwt))+  
  geom_point()+  
  scale_x_log10(breaks=scales::trans_breaks('log10', function(x) 10^x)) +  
  scale_y_log10(breaks=scales::trans_breaks('log10', function(x) 10^x)) +  
  theme_bw()+  
  annotation_logticks(side='tblr')+  
  theme(panel.grid.minor=element_blank())
```

# R package ggiraphExtra

- For interactive plot
- Keon-Woong Moon(2017)

```
install.packages("ggiraphExtra")
```

<https://github.com/cardiomoon/ggiraphExtra>  
<http://rpubs.com/cardiomoon/231820>