



빅데이터 처리 (화요일 (1:3교시))

7주차 강의

시각화

2022.04.19



Institutor ; JS LEE

시각화 개요

“ 읽는 ” 데이터에서 “ 보는 ” 데이터로

- 시각화(visualization)
 - ✓ 데이터를 활용하여 차트 또는 그래프로 표현하는 것
 - ✓ 표(table)보다 더욱 빠른 이해와 직관을 주어 의사결정에 기여
- 시각화의 목적
 - ✓ 데이터 분석을 위한 시각화 : 기능적으로 접근 - 데이터 탐색 및 모델링 결과의 이해
 - ✓ 데이터 공유를 위한 시각화 : 정보전달 용이성 및 미적인 측면 고려 - 보고서 등을 통한 분석결과 공유

Visual versus Text: What does the brain prefer?

According to **Zabisco**, the average person responds far better to visual information compared to just plain ol' text. Whether you're buying a product or revising for an exam, visual stimulation over text translation allows the brain to consume the material with more consummate ease.

But why is there such a hunger for images and video? Why would we rather click on a short video and avoid reading a big chunk of text? And why does this trend seem to be increasing as time moves forward?

We'll first off, it could be something to do with the fact that 90% of information transmitted to the brain is visual, and visuals are processed in the brain at 60,000 times the speed of text. In other words, we look at pictures and videos regularly and we consume them more quickly than we do text.

뇌로 전송되는 정보의 90%는 시각적이고 뇌는
텍스트보다 60,000배 빠른 속도로 시각 자료를 처리



시각화 개요

“읽는” 데이터에서 “보는” 데이터로

- 시각화 이점

- ✓ 많은 양이 데이터를 요약하여 표현
: 데이터 패턴과 현상의 식별 및 예측 용이
- ✓ 한 눈에 직관적으로 인지 가능
: 통계나 분석 기술 등의 전문 지식 없이도 쉽게 인사이트 발견
- ✓ 데이터 스토리텔링 활용가능
: 정보 공유와 설득

** 데이터를 이용하여 의미 있는 이야기 전달

시각화 개요

역사속의 시각화



Florence Nightingale

플로렌스 나이팅게일
(1820~1910)

“나는 간호를 받는 사람들의
안녕을 위해 헌신하겠습니다.”

플로렌스 나이팅게일 - 위키백과, 우리 모두의 백과사전

플로렌스 나이팅게일(영어: Florence Nightingale, OM, DStJ, 1820년 5월 12일 ~ 1910년 8월 13일)은 영국의 간호사, 작가, 통계학자이다.

직업: 간호사, **통계학자**

사인: 병사

학력: 독일에서 간호사 수업

종교: 잉글랜드 성공회

생애 · 간호수업 · 간호활동 · 나이팅게일 다시 읽기

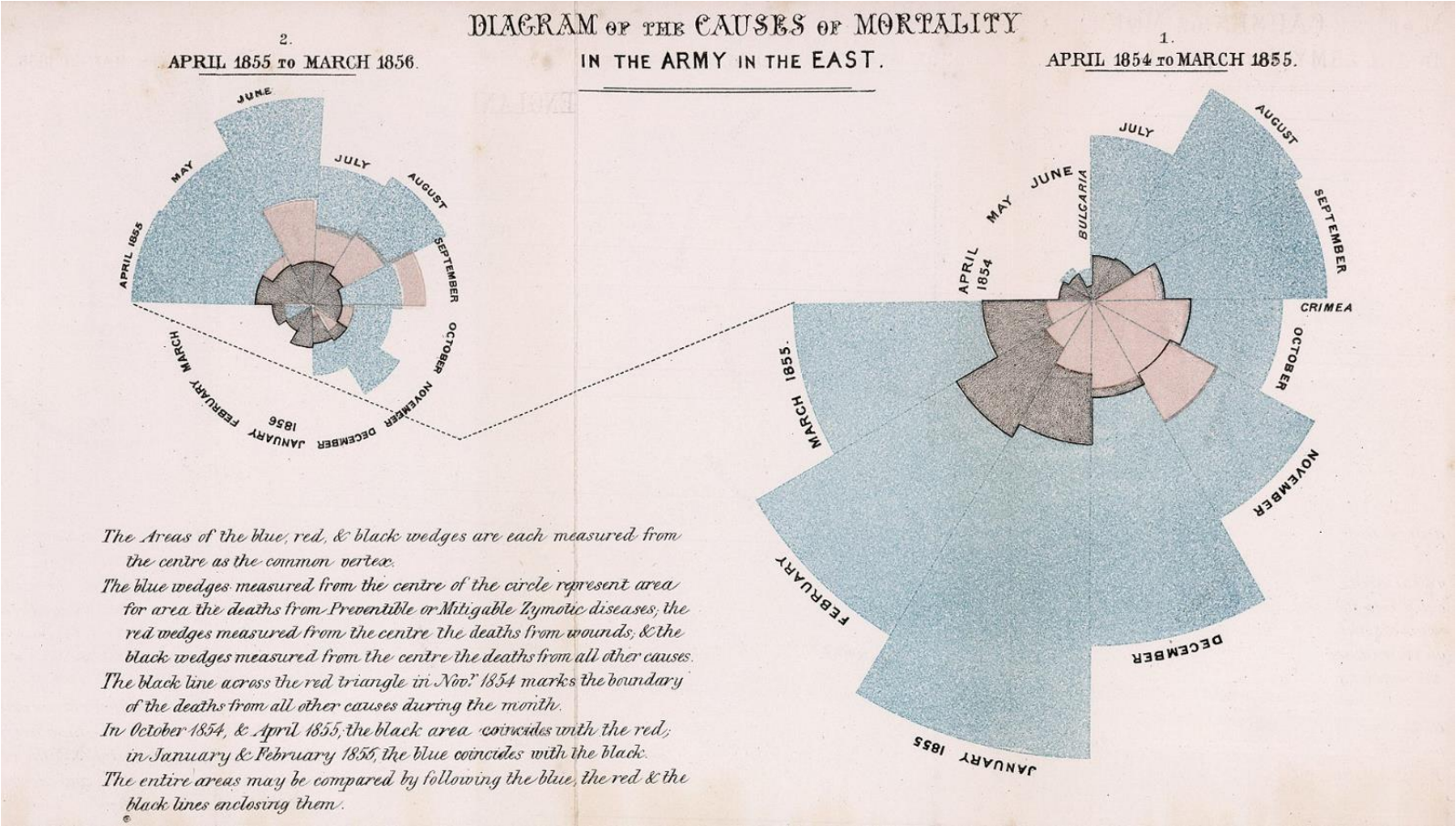


“ 위생의 문제가 군인 사망의 주
원인임일 파이 차트를 이용하여
설득 후 지원 받음 ”

시각화 개요

역사속의 시각화

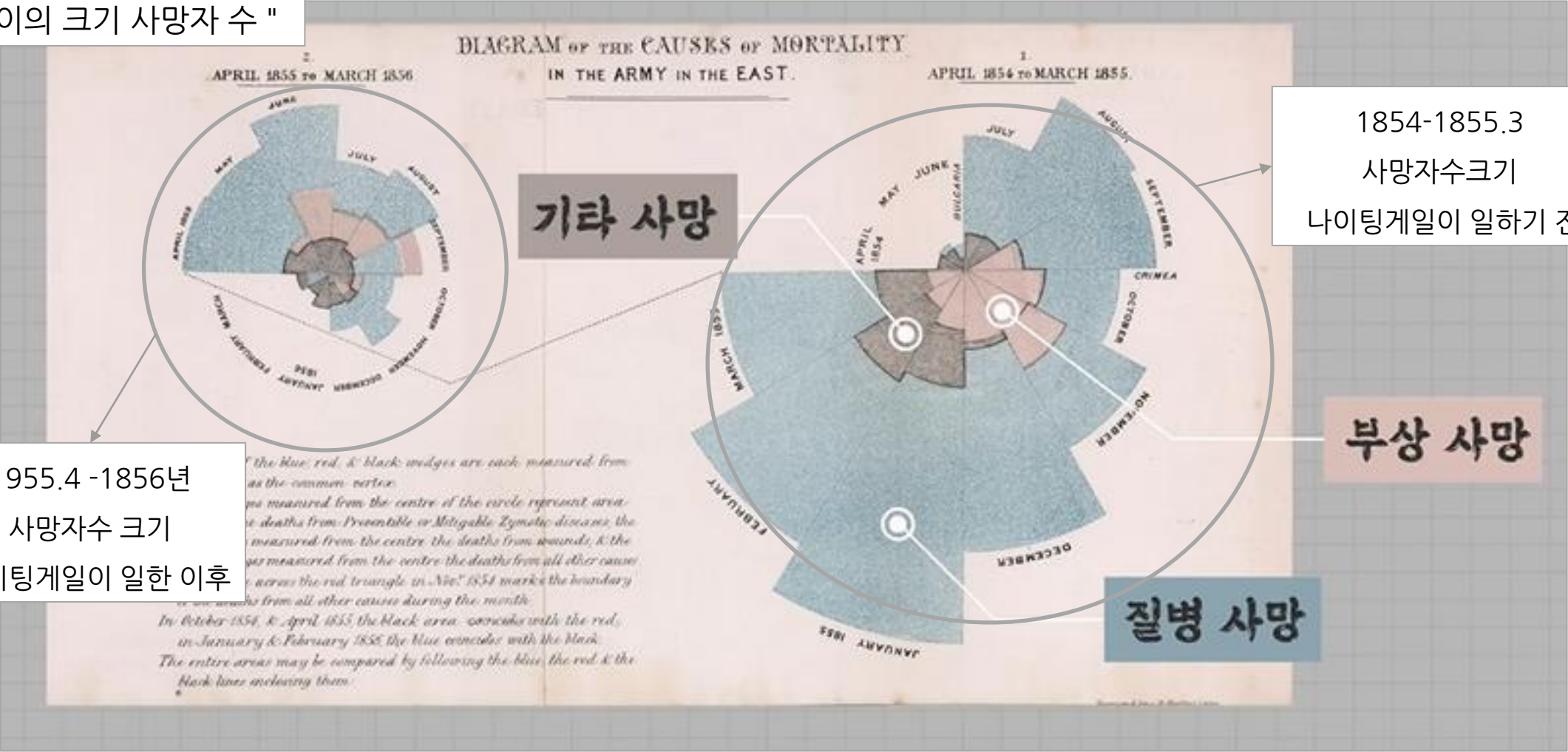
이 시각화의 특징은?



시각화 개요

역사속의 시각화

“파이의 크기 사망자 수 ”

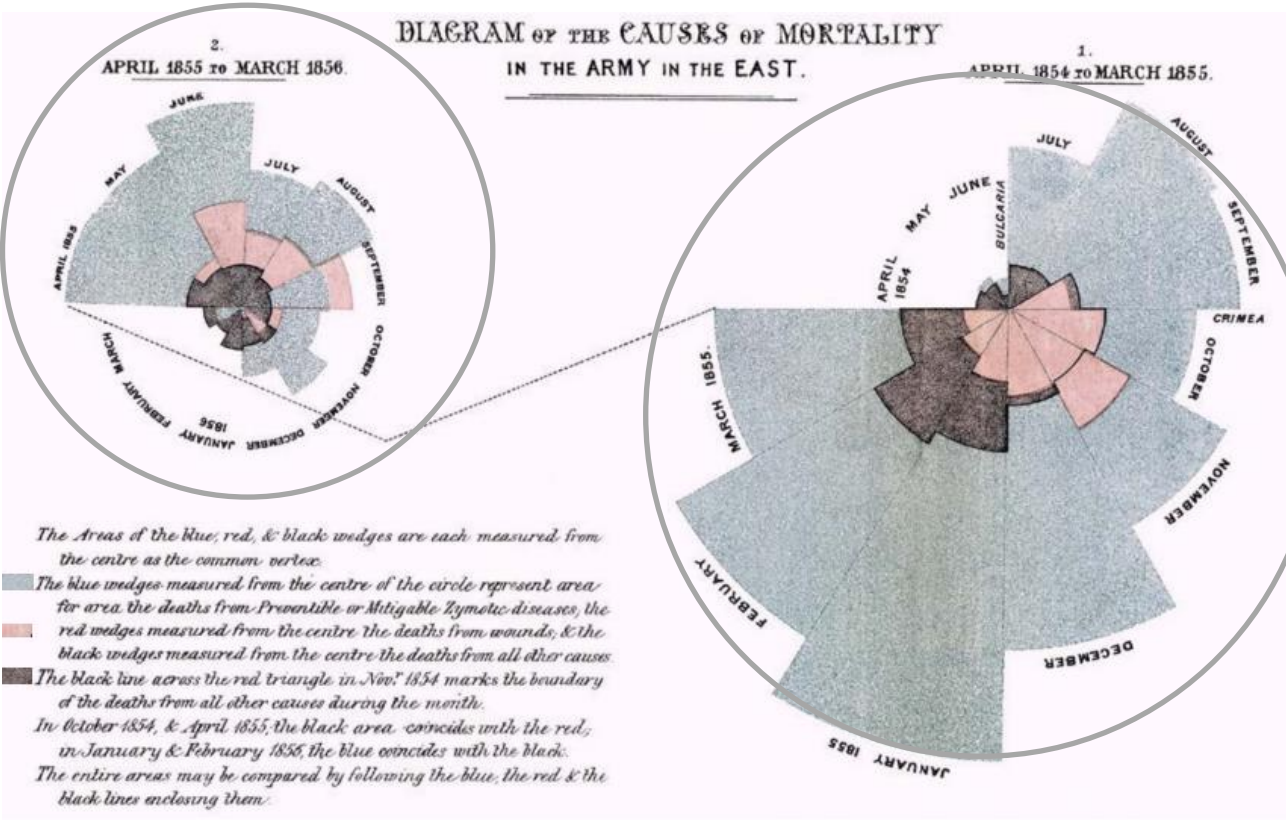


시각화 개요

역사속의 시각화

“파이의 크기 사망자 수 ”

1955.4 -1856년
사망자수크기
나이팅게일이 일한 이후



1854-1855.3
사망자수크기
나이팅게일이 일하기 전

시각화 개요

역사속의 시각화

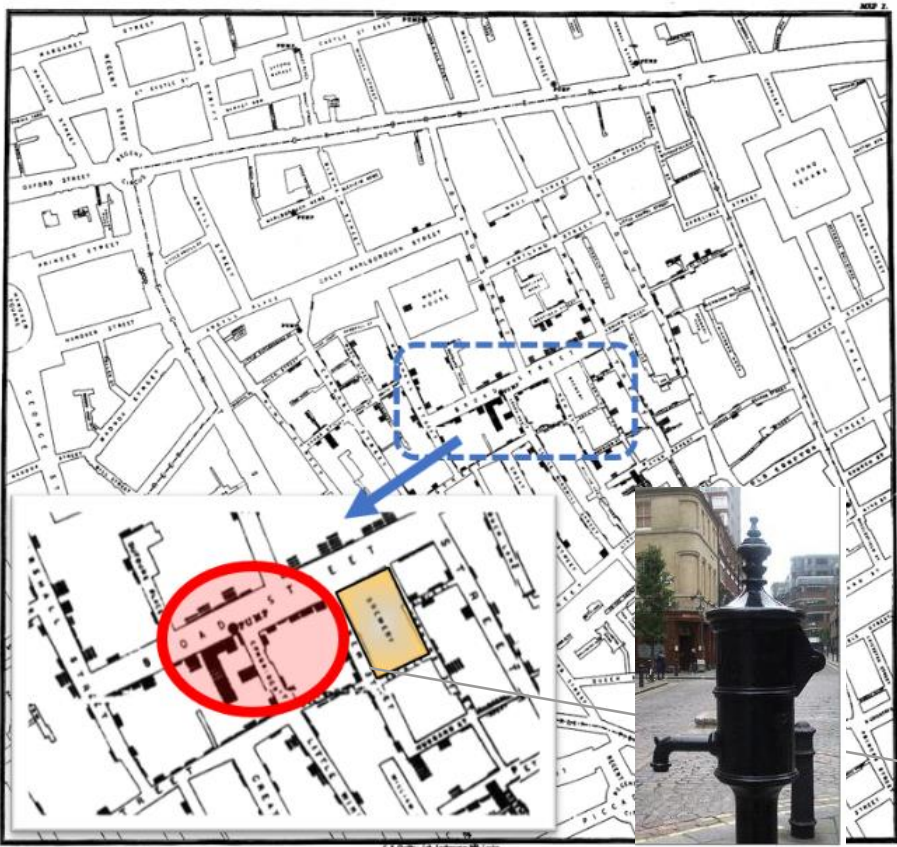
“존 스노우의 콜레라 맵”

1854년 영국 런던 소호지구 “콜레라로 사람들이 죽기 시작함“

의사 존 스노우는 원인 규명 시작

보유 데이터는 주소와 사망자 수

이를 기반으로 주소에 사망자의 수를 막대그래프로 표시



△ 존 스노의 콜레라 지도

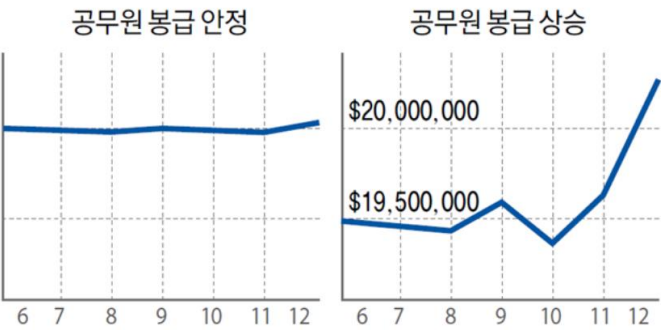
‘브로드 가(Broad Street)’
그 중앙에 ‘공용 펌프(Pump)’ 존재
사망자 83명 중 73명 : 이 펌프 가까이 거주
73명 중 61명 - 펌프 물을 일상적으로 마심
펌프로부터 229미터 반경내 - 2주간 900명 사망
브로드가 주민 896명 중 90명 사망

refer :<https://www.erc.re.kr/webzine/vol33/sub24.jsp><https://twdatastory.tistory.com/entry>
데이터인문학-사람을-향하는-데이터-주제-강연

시각화 개요

바른 판단 및 과장된 표현

- ✓ 세로축 변경으로 인한 과장된 표현



- ✓ 축의 기준 국가별로 상이

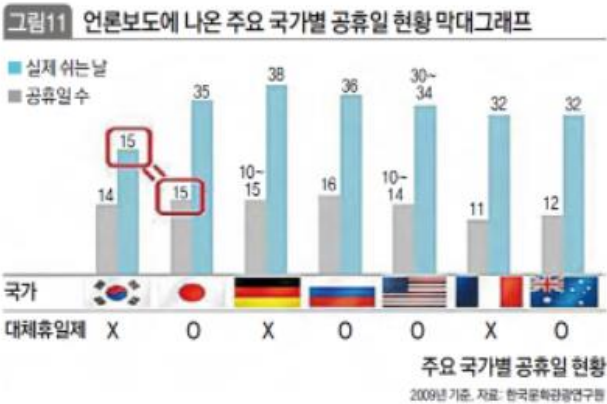


그림2 영어성적 그래프 2

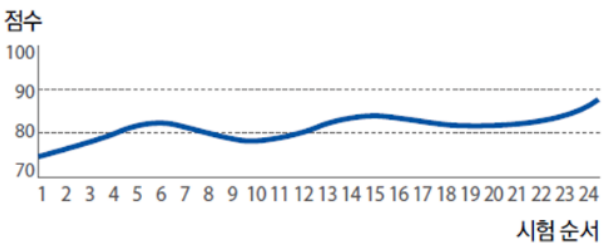


그림3 영어성적 그래프 3

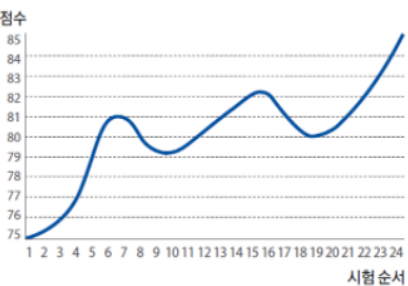
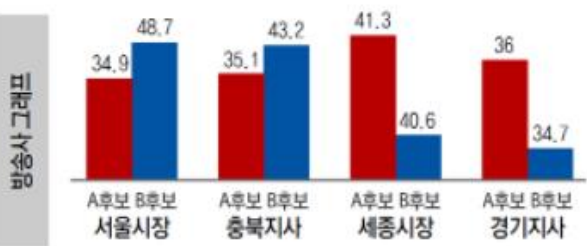


그림9 한 방송사 보도에 나온 지지도 그래프



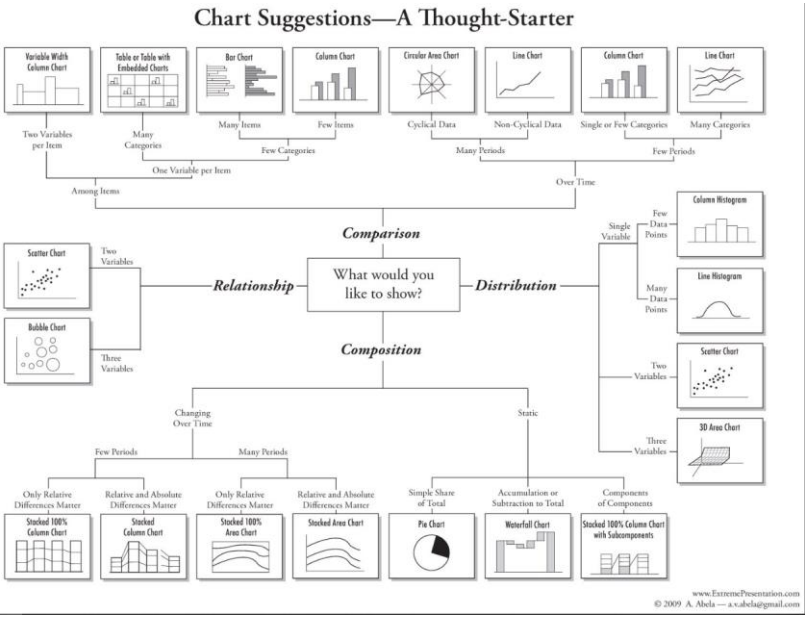
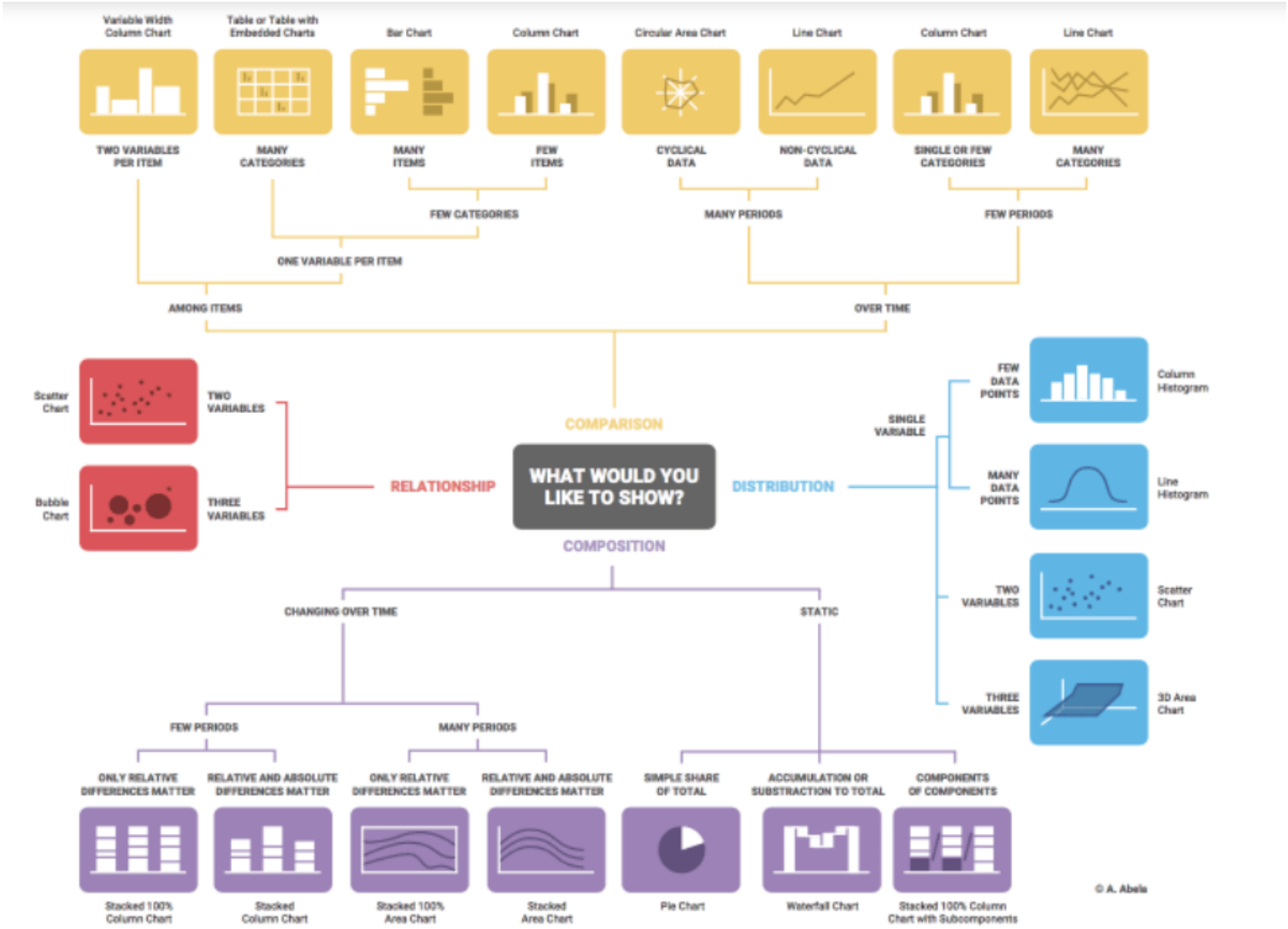
시각화 개요

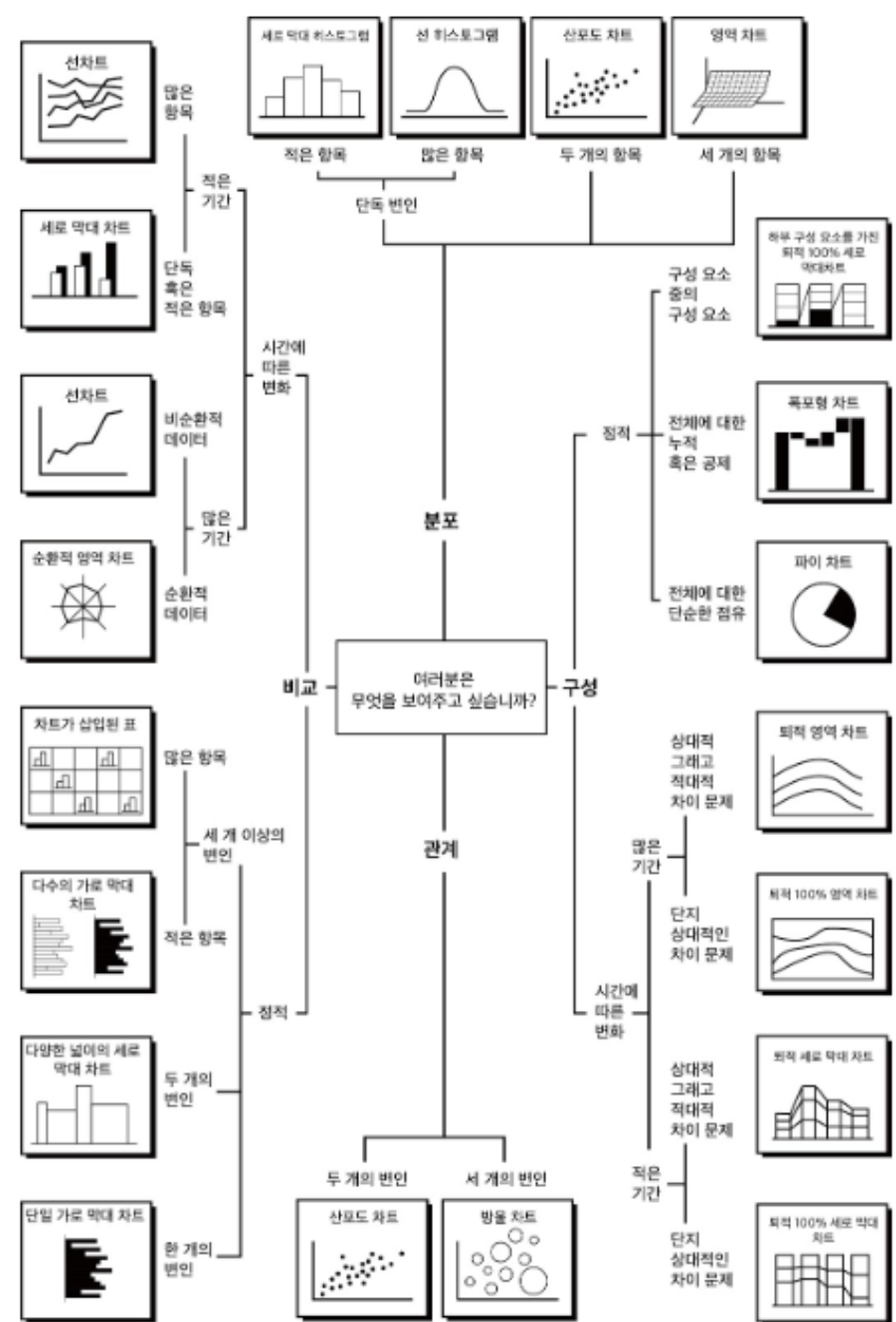
시각화(그래프) 유형

- 시각화의 유형을 선택에는 어려움이 존재함
- 시각화이론에서 많이 사용되는 시각화 선택방법은 앤드류 아벨라(Andrew Abela)의 차트 선택방법 임
 - 비교(Comparison)
 - 구성(Composition)
 - 분포(Distribution)
 - 관계(Relationship)
- 시각화 표출 유형의 결정 시 고려사항
 - ✓ 얼마나 많은 변수들이 하나의 그래프에서 표출되기를 원하는가?
 - ✓ 각 변수에 대하여 얼마나 많은 데이터 점들이 표현되어질 것인가?
 - ✓ 시점 또는 항목간 또는 집단간 값들을 비교하고자 하는가?

시각화 개요

앤드류 아벨라(Andrew Abela)의 차트 선택방법





R - 시각화 - ggplot

ggplot개요

- Hadley Wickham 교수에 의해 2005년부터 개발되고 있으며 ‘Grammar of Graphics’의 개념을 적용한 Plot으로 기본 R 그래픽스에서 제공하는 대부분의 작업을 효과적으로 수행함.
- geom은 좌표체계와 데이터의 점들을 표현하는 시각적 부호(표시)
 - +geom(F, A)
- 데이터의 값들을 표현하기 위하여 크기, 색상, 좌표(x, y)의 위치 등의 속성들을 geom의 미학(aesthetics)적으로 매핑(mapping)함

R 시각화 : ggplot

ggplot의 그래픽의 주요문법 (1,2,3 까지 지정이 기본, 4,5는 선택)

1. data: 사용할 정리된 데이터

1. Tidy Data

```
p <- ggplot(data = gapminder, ...
```

gdp	lifexp	pop	continent
340	65	31	Euro
227	51	200	Amer
909	81	80	Euro
126	40	20	Asia


2. Mapping : 시각적 속성 맵핑

2. Mapping

```
p <- ggplot(data = gapminder,  
  mapping = aes(x = gdp,  
    y = lifexp, size = pop,  
    color = continent))
```

3. Geometric : 형태 지정(점,선, 면적 등)

3. Geom

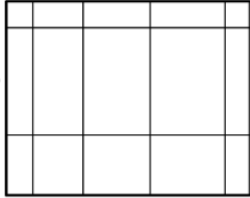


```
p + geom_point()
```

4. coordinate : 좌표계 및 척도 (로그, 맵, 데카르트 좌표, ..)

4. Co-Ordinates & Scales

```
p + coord_cartesian() +  
  scale_x_log10()
```

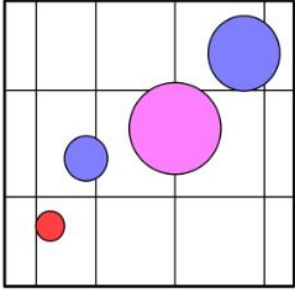


5. Label : 레이블 및 안내선

5. Labels & Guides

```
p + labs(x = "log GDP",  
  y = "Life Expectancy",  
  title = "A Gapminder Plot")
```

A Gapminder Plot

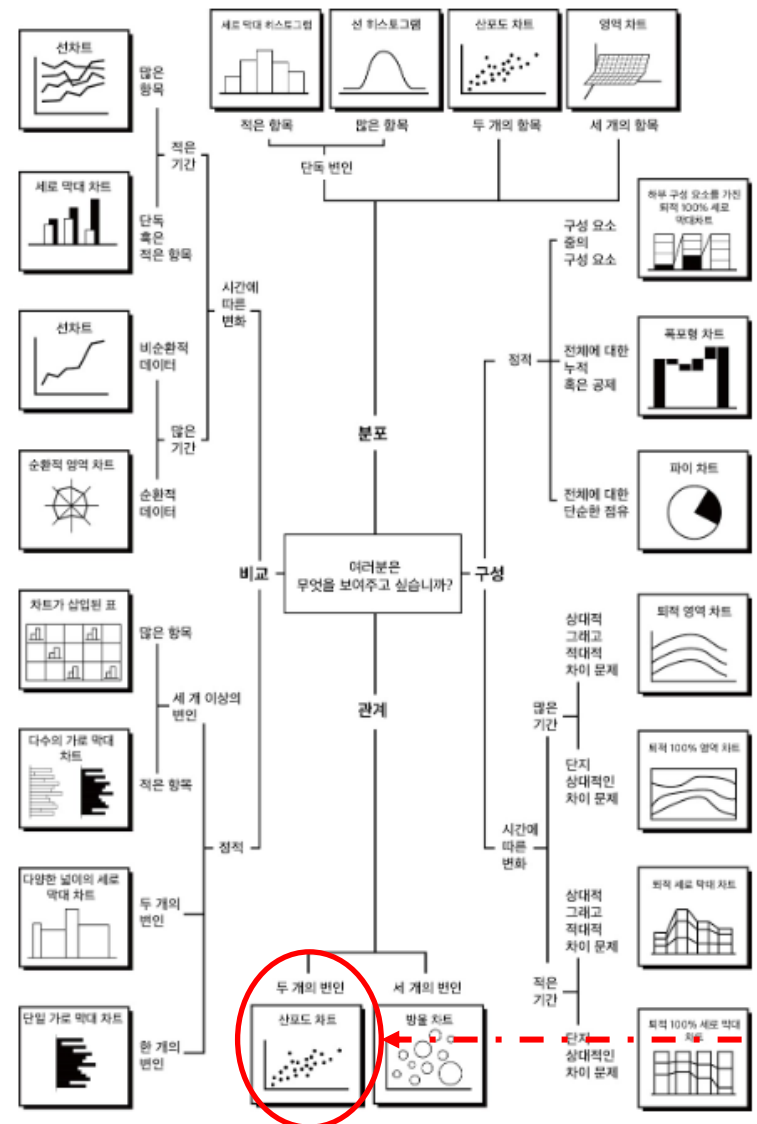


- Asia
- Euro
- Amer

- 0-35
- 36-100
- >100

R 시각화 : ggplot

gapminder데이터 셋을 이용하여, 경제수준(1인당GDP)과 기대수명의 관계를 살펴보자



* 관계 / 두 변수 / 두변수는 연속형 변수

ggplot의 산점도 그려보기

1. data: 사용할 정리된 데이터

1. Tidy Data

```
p <- ggplot(data = gapminder, ...)
```

gdp	lifexp	pop	continent
340	65	31	Euro
227	51	200	Amer
909	81	80	Euro
126	40	20	Asia

2. Mapping : 시각적 속성 맵핑

2. Mapping

```
p <- ggplot(data = gapminder, mapping = aes(x = gdp, y = lifexp, size = pop, color = continent))
```

3. Geometric : 형태 지정(점, 선, 면적 등)

3. Geom

```
p + geom_point()
```

4. coordinate : 좌표계 및 척도 (로그, 맵, 데카르트 좌표, ..)

4. Co-Ordinates & Scales

```
p + coord_cartesian() + scale_x_log10()
```

5. Label : 레이블 및 안내선

5. Labels & Guides

```
p + labs(x = "log GDP", y = "Life Expectancy", title = "A Gapminder Plot")
```

A Gapminder Plot

개요 – 필요 패키지

데이터 및 그래프 관련

```
install.packages("tidyverse")
```

"tidyverse" (설치 시, ggplot2도 설치됨)

```
install.packages("gapminder")
```

색상

```
install.packages('nord')
```

```
install.packages('viridis')
```

설치 후, library()이용하여 로딩하기

애니메이션

```
install.packages('gganimate')
```

```
install.packages('gifski')
```

```
install.packages('av')
```

지도 관련

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

```
install.packages("maps")
```

```
install.packages("mapproj")
```


R 시각화 : ggplot

데이터 셋 : gapminder

1. data: 사용할 정리된 데이터

1. Tidy Data

```
p <- ggplot(data = gapminder, ...
```

gdp	lifexp	pop	continent
340	65	31	Euro
227	51	200	Amer
909	81	80	Euro
126	40	20	Asia

1. 데이터 불러온 후, 데이터 프레임구조화 하기

```
data("gapminder")
gapminder <- as.data.frame(gapminder)
```

2. gapminder 객체명을 입력하여 구조를 간단히 살펴보기

```
gapminder
?gapminder
```

5대륙의 142개 국가의 1인당 GDP와 기대수명 등을 정리한 DataSet

	country	continent	year	lifeExp	pop	gdpPercap
	<fct>	<fct>	<int>	<dbl>	<int>	<dbl>
1	Afghanistan	Asia	1952	28.8	8425333	779.
2	Afghanistan	Asia	1957	30.3	9240934	821.
3	Afghanistan	Asia	1962	32.0	10267083	853.
4	Afghanistan	Asia	1967	34.0	11537966	836.
5	Afghanistan	Asia	1972	36.1	13079460	740.
6	Afghanistan	Asia	1977	38.4	14880372	786.
7	Afghanistan	Asia	1982	39.9	12881816	978.
8	Afghanistan	Asia	1987	40.8	13867957	852.
9	Afghanistan	Asia	1992	41.7	16317921	649.
10	Afghanistan	Asia	1997	41.8	22227415	635.
#	... with 1,694 more rows					

Gapminder data.

Description

Excerpt of the Gapminder data on life expectancy, GDP per capita, and population by country.

Usage

gapminder

Format

The main data frame `gapminder` has 1704 rows and 6 variables:

country

factor with 142 levels

continent

factor with 5 levels

year

ranges from 1952 to 2007 in increments of 5 years

lifeExp

life expectancy at birth, in years

pop

population

gdpPercap

GDP per capita (US\$, inflation-adjusted)

The supplemental data frame `gapminder_unfiltered` was not filtered on year or for complete data and has 3313 rows.

3. ggplot이라는 함수에 사용할 데이터 알려주고, ggplot()에 정의한 내용은 P에 지정

```
p <- ggplot(data=gapminder)
```



R 시각화 : ggplot

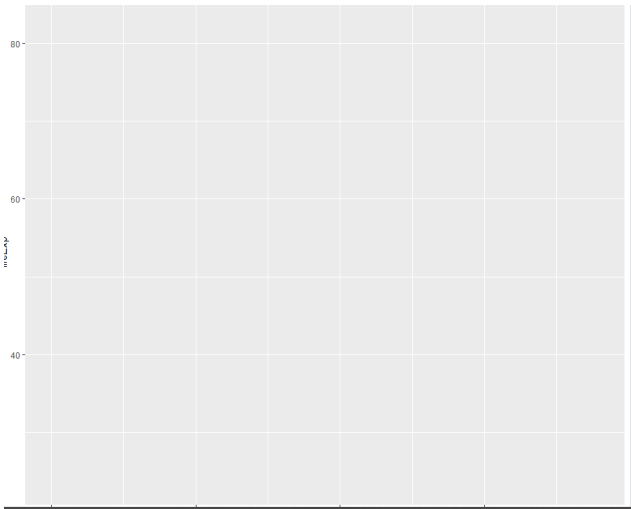
2. aes() 함수를 이용하여 데이터의 어떠한 것들을 시각적으로 맵핑할 것인지를 알려주기

1인당GDP(gdpPercap 와 기대수명(lifeExp)을 살펴보자

```
p <- ggplot(data = gapminder,
            mapping = aes(x = gdpPercap, y = lifeExp))
```

객체 p를 살펴보자

```
p
```



x축과 y축의
존재하는 빈도표만
존재

객체 p에 어떠한 정보들이 담겨있는지 살펴보자

```
str(p)
```

2. Mapping : 시각적 속성 맵핑

```
2. Mapping
p <- ggplot(data = gapminder,
            mapping = aes(x = gdp,
                          y = lifexp, size = pop,
                          color = continent))
```

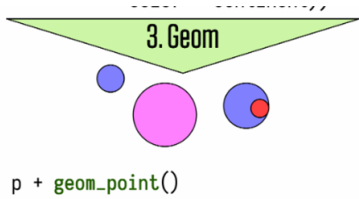
R 시각화 : ggplot

3. geom_ 함수를 이용하여 도표에 미학적 요소들을 추가하자

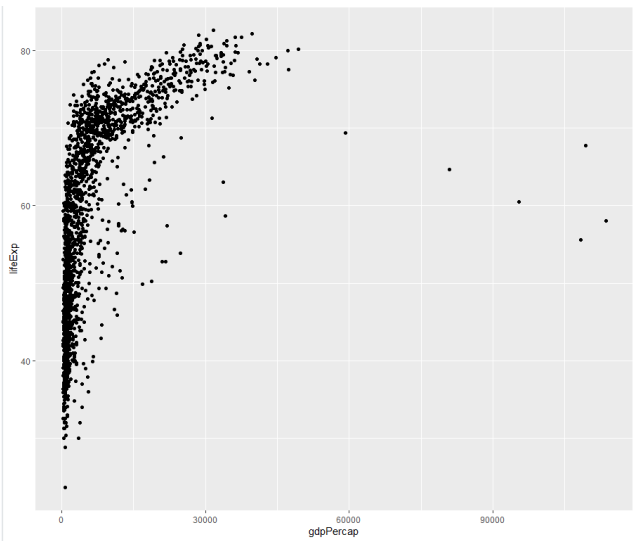
1인당GDP(gdpPercap 와 기대수명(lifeExp)의 관계를 위해 1) 산점도(geom_point())를 활용하고
2) 추세선을 표현 및 표준오차(각 점들이 추세선에서 떨어져 있는 정도)가 어느 정도인지를 나타내
보자 (geom_smooth())

3) 1)번과 2)번을 동시에 표현 해보자
미학적 요소 추가, 도표의 각종 요소 추가는 + 를 이용함

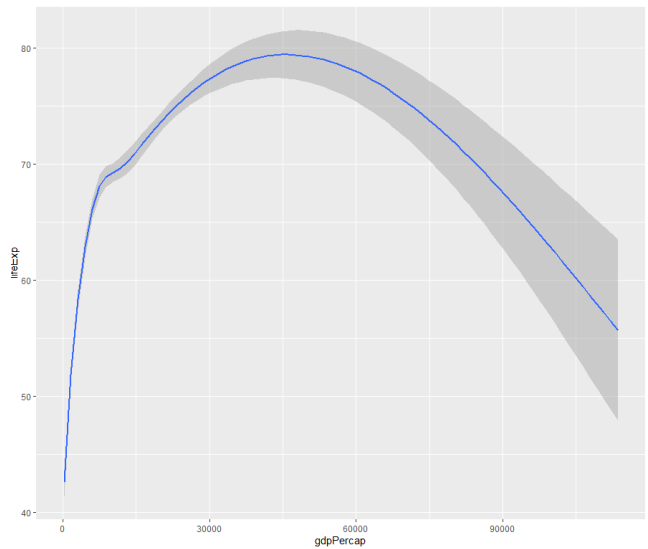
3. Geometric : 형태 지정(점,선, 면적 등)



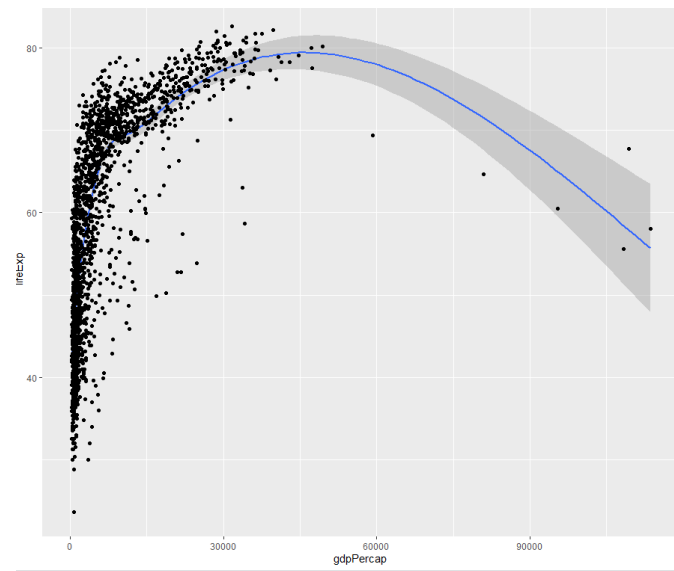
p+ geom_point()



p+ geom_smooth()



p+geom_smooth()+geom_point()

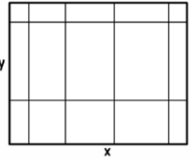


R 시각화 : ggplot

4. `coordinate` : 좌표계 및 척도 (로그, 맵, 데카르트 좌표, ..)

4. Co-Ordinates & Scales

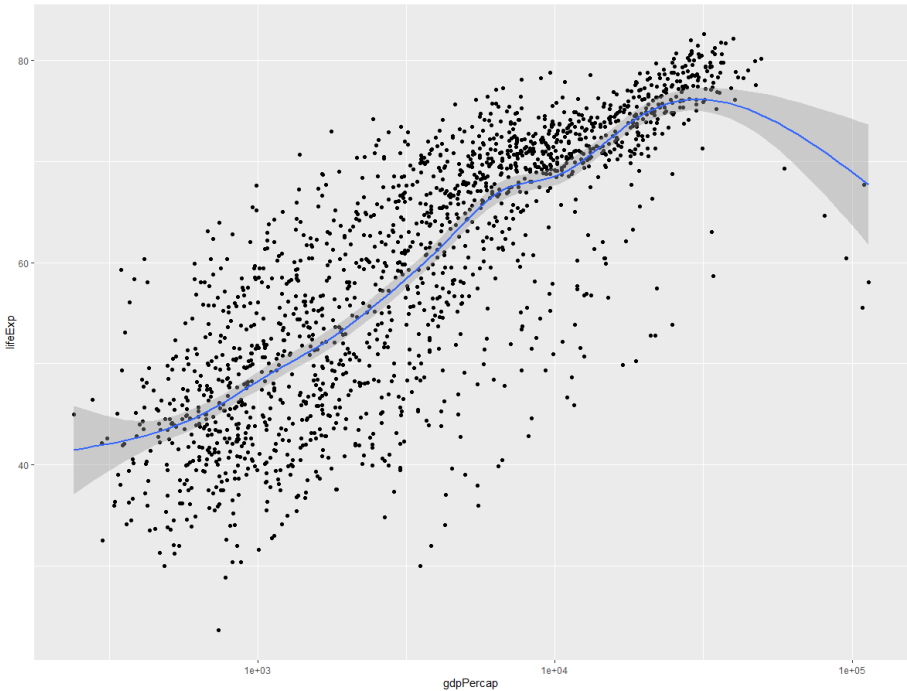
```
p + coord_cartesian() +  
  scale_x_log10()
```



4. 축의 척도 변환을 해보자

1인당 국민 총생산은 골고루 분포되어 있지 않고 밀집되어 있기에, x축 스케일을 로그 변환을 해보자
`scale_x_log10()` 함수를 이용하자

```
p + geom_point() + geom_smooth(method = "gam") + scale_x_log10()
```

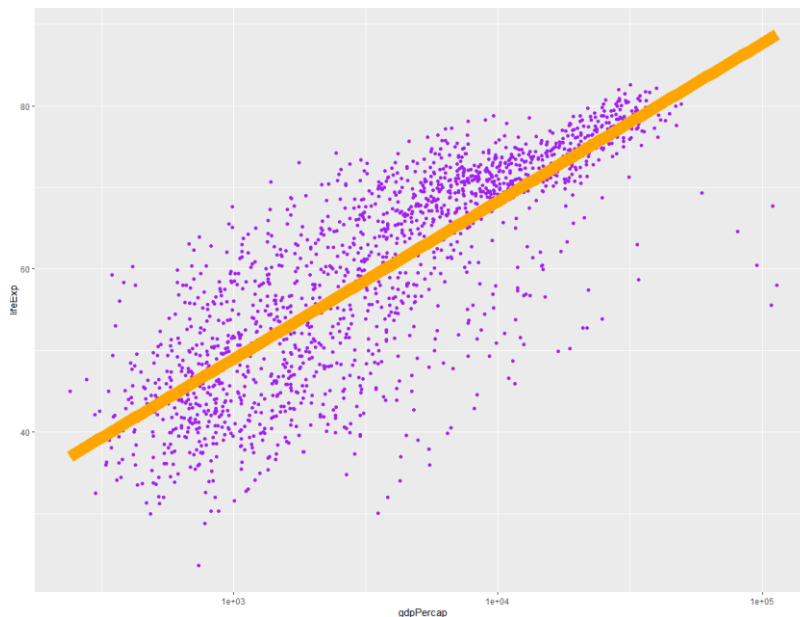


R 시각화 : ggplot

* geom_ 함수에 미적인 요소를 추가해보자

- 1) 산점도의 색상은 보라(color=purple), 2) 추세선은 gam(일반화 가법 모형, default) 방법이 아닌 lm(선형모형)을 이용하고, 3) 추세선의 색상은 오렌지(orange) 그리고, 4) 표준오차를 없애고(se=FALSE), 5) 추세선을 두껍게(size = 6) 하고, 6) x축 스케일을 상용로그변환(scale_x_log10())

```
p + geom_point(color= "purple") +geom_smooth(method="lm",  
color='orange',se=FALSE, size=6 ) +scale_x_log10()
```

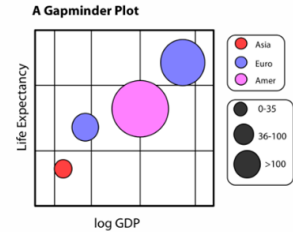


R 시각화 : ggplot

5. Label : 레이블 및 안내선

5. Labels & Guides

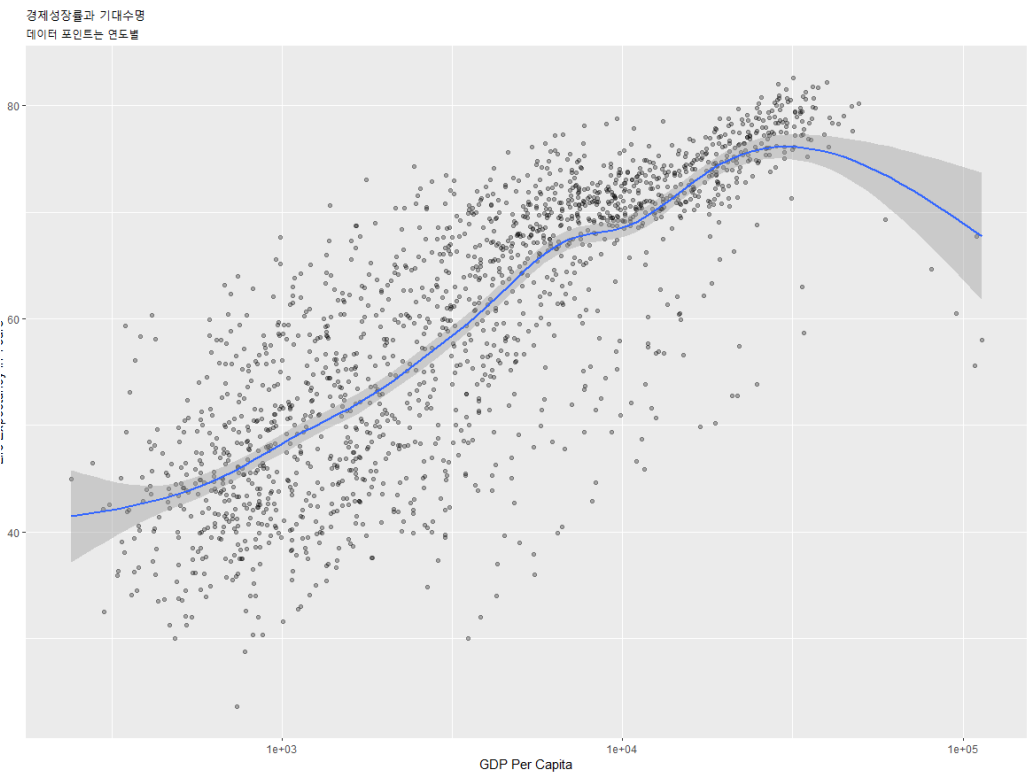
```
p + labs(x = "log GDP",  
y = "Life Expectancy",  
title = "A Gapminder Plot")
```



5. 레이블을 추가하자

x축, y축 명과, 제목, 부제목, 캡션 등을 추가해보자.

```
p + geom_point(alpha=0.3) + geom_smooth(method = 'gam') + scale_x_log10() +  
  labs(x = 'GDP Per Capita', y = 'Life Expectancy in Years', title = '경제성장률과 기대수명',  
    subtitle = '데이터 포인트는 연도별', caption = '자료:갭마인더.')
```



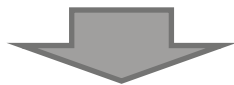
geom_point에서
alpha: 투명도

- 전체적으로, 경제수준이 좋아짐에 따라 평균수명도 늘어남을 알 수 있음



R 시각화 : ggplot

- 전체적으로, 경제수준이 좋아짐에 따라 평균수명도 늘어남을 알 수 있음



- 경제수준과 평균수명의 관계가 대륙별, 인구규모별 어떤 차이가 있는지 알아보자

=> 4개의 차원을 기술해야 함 (축으로는 2차원만 가능), 그 이상의 차원은 **색상**이나, **크기** 등으로 표현 가능

p1이라는 새로운 객체를 지정하여, 대륙별, 인구규모별 차이를 살펴보자

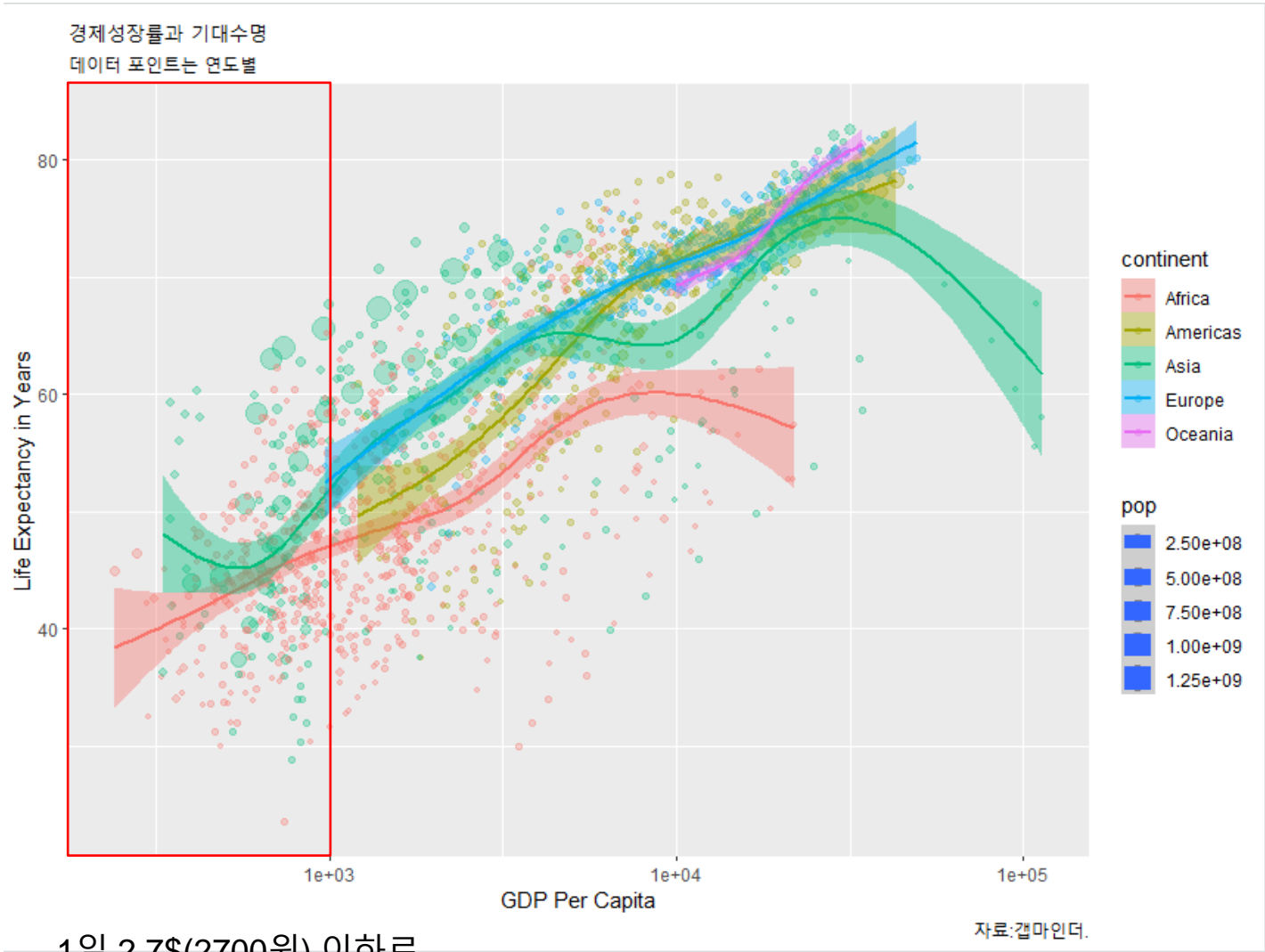
```
p1 <- ggplot(data = gapminder,
             mapping = aes(x = gdpPercap, y = lifeExp,
                          color = continent, size = pop, fill = continent))
```

* fill=continent => se를 continent색으로 채우기

크기 : 규모를 나타낼 수 있는 변수
예> 인구규모, 연령 등 (연속형변수)
색상: 구분을 하는 변수
예> 대륙, 성별 등 (범주형 변수)

```
p1 + geom_point(alpha=0.3) + geom_smooth(method = 'gam') +
scale_x_log10() +
labs(x = 'GDP Per Capita', y = 'Life Expectancy in Years',
     title = '대륙별 경제성장률과 기대수명',
     subtitle = '데이터 포인트는 연도별',
     caption = '자료:갭마인더.')
```

R 시각화 : ggplot



오세아니아 대륙의 국가들은 대부분 GDP도 높고 평균수명도 김

유럽과 아메리카 대륙은 1,000달러 이하의 국가가 없으며, 평균수명도 김

아시아 대륙은 1인당 GDP가 높다고 반드시 평균수명이 길지 않은 국가들이 존재함
인구규모가 큰 나라가 많음

아프리카대륙은 GDP도 평균 수명도 가장 짧음

저소득이면서, 평균수명이 낮은 대륙은 : 아시아, 아프리카

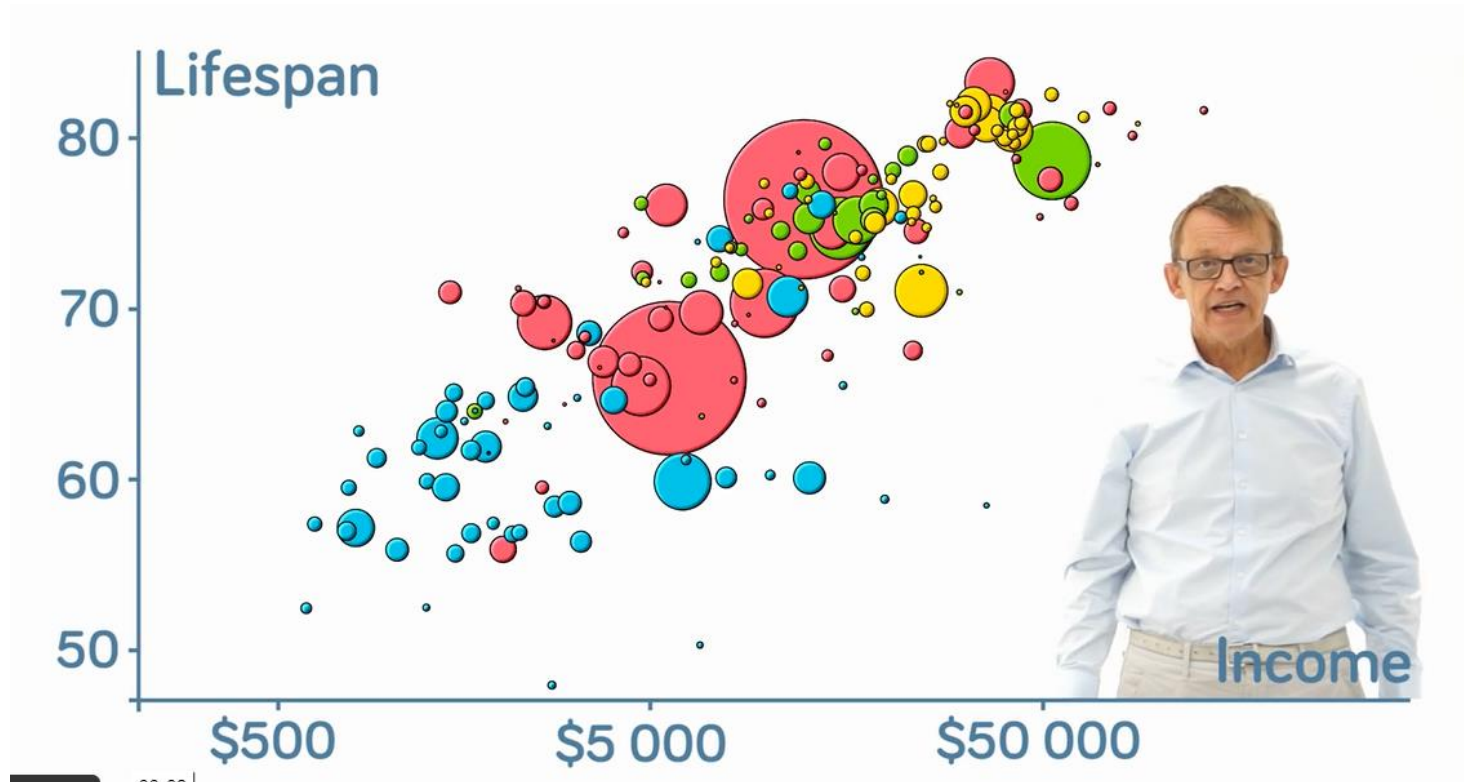
1일 2.7\$(2700원) 이하로 사는 국가

options("scipen" = 100) : 지수를 자연수 형태로

R 시각화 : ggplot

gapminder.org

gapminder



R 시각화 : ggplot

“ R을 활용하여 그래프에 애니메이션을 입혀봅시다. “

필요패키지

```
install.packages('gganimate') # 애니메이션  
install.packages('nord') # 색상  
install.packages('viridis') # 색상  
install.packages('gifski')  
install.packages('av')
```

동작의 기준이 되는 변수
:year
year가 변할 때 마다 걸리는 시간:1초

```
ani1 <- p1+geom_point( alpha=0.5)+  
  scale_color_viridis(option = "C",discrete = TRUE) + # 색상  
  scale_x_log10()+  
  theme_minimal()+ # 배경  
  theme(legend.position='right')+ # 범례위치  
  # 애니메이션 길이  
  transition_states(year,  
    state_length=1)+  
  }  
# 그래프 타이틀  
ggtitle('Now showing {closest_state}')  
  
ani1
```

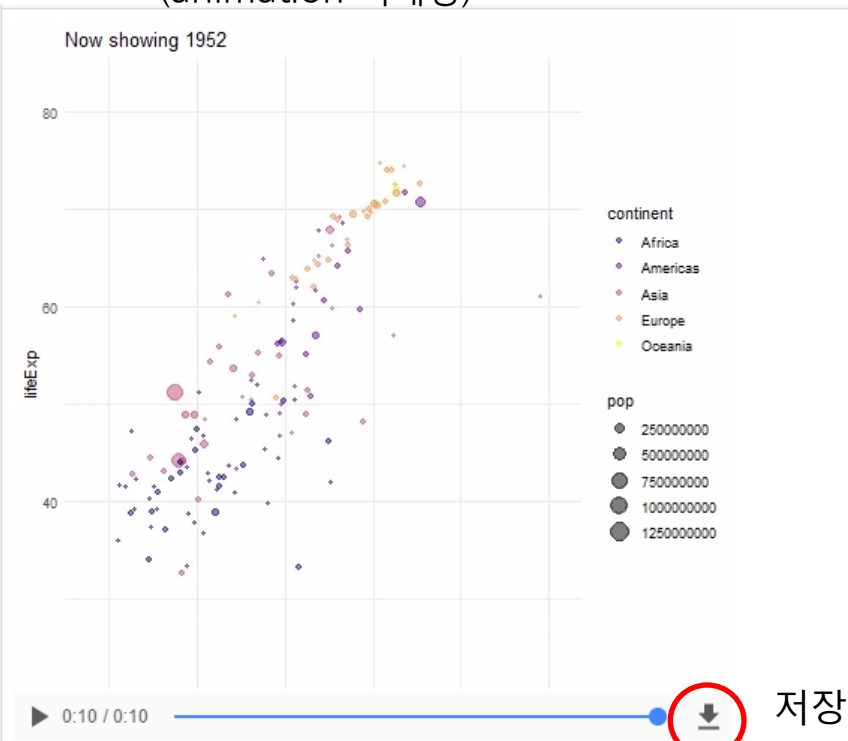
R 시각화 : ggplot

“ R을 활용하여 그래프에 애니메이션을 입혀봅시다. “

내가 만든 분석 animation저장하기

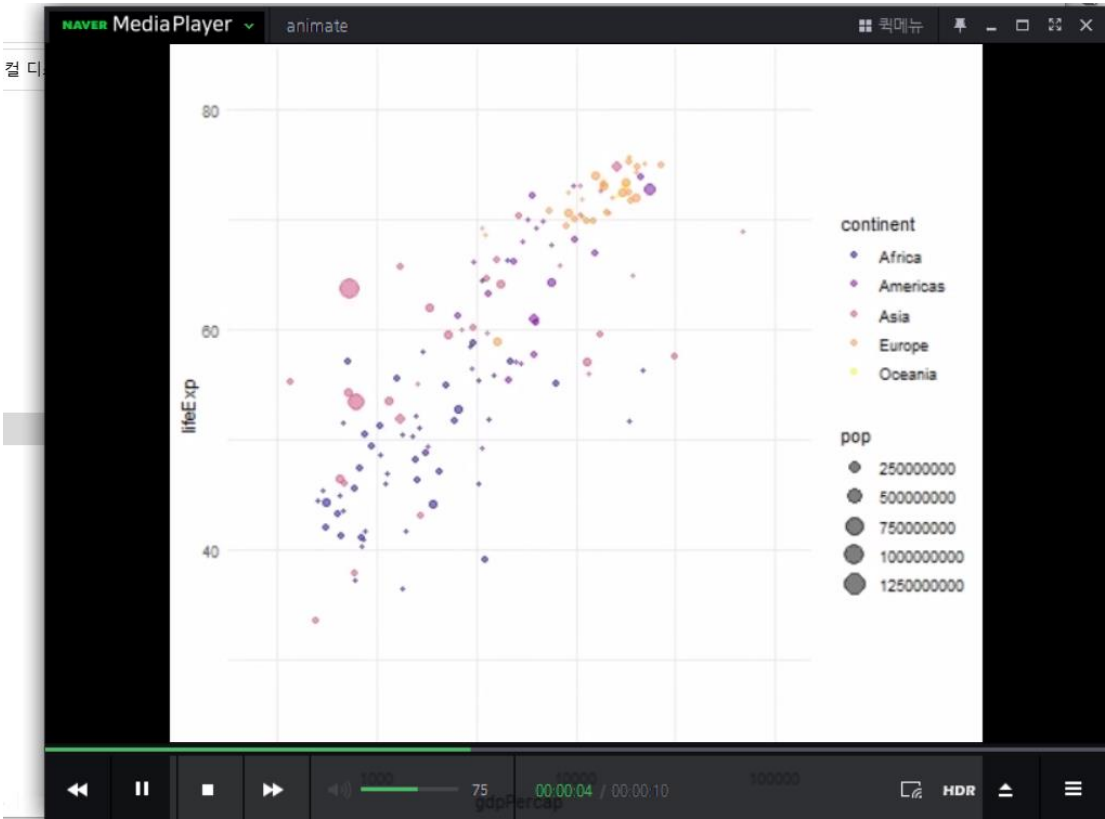
```
animate(ani1, renderer=av_renderer())
```

(animation 객체명)



저장

저장한 파일을 미디어 플레이어로 실행하기



R 시각화 : ggplot

주요 geom_ 함수 종류

함수	도형	도형의 속성
geom_bar()	Bar chart	color, fill, alpha
geom_boxplot()	Box plot	color, fill, alpha, notch, width
geom_density()	Density plot	color, fill, alpha, linetype
geom_histogram()	Histogram	color, fill, alpha, linetype, binwidth
geom_hline()	Horizontal lines	color, alpha, linetype, size
geom_jitter()	Jittered points	color, size, alpha, shape
geom_line()	Line graph	color, alpha, linetype, size
geom_point()	Scatterplot	color, alpha, shape, size
geom_rug()	Rug plot	color, side
geom_smooth()	Fitted line	method, formula, color, fill, linetype, size
geom_text()	Text annotations	많은 옵션이 있으므로 도움말 참조
geom_violin()	Violin plot	color, fill, alpha, linetype
geom_vline()	Vertical lines	color, alpha, linetype, size

ggplot_cheat sheet참조
(7주차 강의안 폴더)

Data Visualization with ggplot2 : : CHEAT SHEET

Basics

ggplot2 is based on the **grammar of graphics**, the idea that you can build every graph from the same components: a **data set**, a **coordinate system**, and **geoms**—visual marks that represent data points.

data + **geom** + **coordinate system** = **plot**

$F = Y + A$

To display values, map variables in the data to visual properties of the geom (**aesthetics**) like **size**, **color**, and **x** and **y** locations.

data + **geom** + **coordinate system** = **plot**

$F = Y + A$

Complete the template below to build a graph.

```
ggplot(data = <DATA>) +  
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>)) +  
  <COORDINATE_FUNCTION> +  
  <FACET_FUNCTION> +  
  <THEME_FUNCTION>
```

ggplot(data = mpg, aes(x = city, y = hwy)) Begins a plot that you finish by adding layers. Add one geom function per layer.

ggplot2 + city, y = hwy, data = mpg, geom = "point") Creates a complete plot with given data, geom, and mappings. Supplies many useful defaults.

test_plot() Returns the last plot

ggsave("plot.png", width = 5, height = 5) Saves last plot as 5" x 5" file named "plot.png" in working directory. Matches file type to file extension.

Geoms

Use a geom function to represent data points, use the geom's aesthetic properties to represent variables. Each function returns a layer.

GRAPHICAL PRIMITIVES

a = ggplot(economics, aes(date, unemploy))
b = ggplot(seals, aes(x = long, y = lat))

a + geom_blank() (Useful for expanding limits)

b + geom_curve(aes(vend = lat + 1, xend = long - 1, curvature = c), x = xend, y = yend, alpha, angle, color, curvature, linetype, size)

a + geom_path(aes(linetype = "bust", linejoin = "round", linetype = 1), x, y, alpha, color, group, linetype, size)

a + geom_polygon(aes(group = group), x, y, alpha, color, fill, group, linetype, size)

b + geom_rect(aes(xmin = long, ymin = lat, xmax = long + 1, ymax = lat + 1), xmin, ymin, xmax, ymax, alpha, color, fill, linetype, size)

a + geom_ribbon(aes(vmin = unemploy - 900, ymax = unemploy + 900), x, y, alpha, color, fill, group, linetype, size)

LINE SEGMENTS

b + geom_abline(aes(intercept = 0, slope = 1))
b + geom_hline(aes(intercept = lat))
b + geom_vline(aes(intercept = long))

b + geom_segment(aes(vend = lat + 1, xend = long + 1))
b + geom_spoke(aes(angle = 1:1155, radius = 1))

ONE VARIABLE continuous

c = ggplot(mpg, aes(hwy))
c2 = ggplot(mpg)

c + geom_area(aes("bla"), x, y, alpha, color, fill, linetype, size)

c + geom_density(breaked = "gaussian", x, y, alpha, color, fill, group, linetype, size, weight)

c + geom_dotplot(x, y, alpha, color, fill)

c + geom_freqpoly(x, y, alpha, color, group, linetype, size)

c + geom_histogram(binwidth = 5), x, y, alpha, color, fill, linetype, size, weight

c2 + geom_qq(aes(sample = hwy)), x, y, alpha, color, fill, linetype, size, weight

discrete

d = ggplot(mpg, aes(lt))
d + geom_bar()

TWO VARIABLES

continuous x, continuous y

e = ggplot(mpg, aes(city, hwy))

e + geom_label(aes(label = city, nudges = 1, nudges_y = 1, check_overlap = TRUE), x, y, label, alpha, angle, color, family, fontface, hjust, linetype, size, vjust)

e + geom_litter(height = 2, width = 2), x, y, alpha, color, fill, shape, size, stroke

e + geom_point(), x, y, alpha, color, fill, shape, size, stroke

e + geom_quantile(), x, y, alpha, color, group, linetype, size, weight

e + geom_rug(sides = "bl", x, y, alpha, color, fill, linetype, size)

e + geom_smooth(method = lm), x, y, alpha, color, fill, group, linetype, size, weight

e + geom_text(aes(label = city, nudges = 1, nudges_y = 1, check_overlap = TRUE), x, y, label, alpha, angle, color, family, fontface, hjust, linetype, size, vjust)

continuous bivariate distribution

h = ggplot(diamonds, aes(carat, price))

h + geom_bin2d(binwidth = c(0.25, 500)), x, y, alpha, color, fill, linetype, size, weight

h + geom_density_2d(), x, y, alpha, color, group, linetype, size

h + geom_hex(), x, y, alpha, color, fill, size

continuous function

i = ggplot(economics, aes(date, unemploy))

i + geom_area()

i + geom_line()

i + geom_step(direction = "hv"), x, y, alpha, color, group, linetype, size

discrete x, continuous y

f = ggplot(mpg, aes(class, hwy))

f + geom_col(), x, y, alpha, color, fill, group, linetype, size

f + geom_boxplot(), x, y, lower, middle, upper, ymax, ymin, alpha, color, fill, group, linetype, size, weight

f + geom_dotplot(binaxis = "y", stackdir = "center"), x, y, alpha, color, fill, group

f + geom_violin(scale = "area"), x, y, alpha, color, fill, group, linetype, size, weight

visualizing error

jl = data.frame(gp = c("M", "F"), fit = 4.5, se = 1.2)

j = ggplot(fit, aes(gp, fit, ymin = fit - se, ymax = fit + se))

j + geom_crossbar(fill = 2)

j + geom_errorbar(), x, ymax, ymin, alpha, color, fill, group, linetype, size

j + geom_errorbarh(), x, ymax, ymin, alpha, color, fill, group, linetype, size

j + geom_linerange()

j + geom_pointrange()

maps

data = data.frame(murder = USArrests\$Murder, state = tolower(names(USArrests)))

map = map_data("state")

k = ggplot(data, aes(fill = murder))

k + geom_map(aes(map_id = state), map = map)

k + geom_raster(aes(fill = 2), hjust = 0.5, vjust = 0.5, interpolate = FALSE), x, y, alpha, fill)

k + geom_tile(aes(fill = z), x, y, alpha, color, fill, linetype, size, width)

THREE VARIABLES

seals2 = with(seals, sort(delta, long^2 + delta, las = 2))

l = ggplot(seals2, aes(long, lat))

l + geom_contour(aes(z = 2)), x, y, z, alpha, color, group, linetype, size, weight

l + geom_raster(aes(fill = z), hjust = 0.5, vjust = 0.5, interpolate = FALSE), x, y, alpha, fill)

l + geom_tile(aes(fill = z), x, y, alpha, color, fill, linetype, size, width)

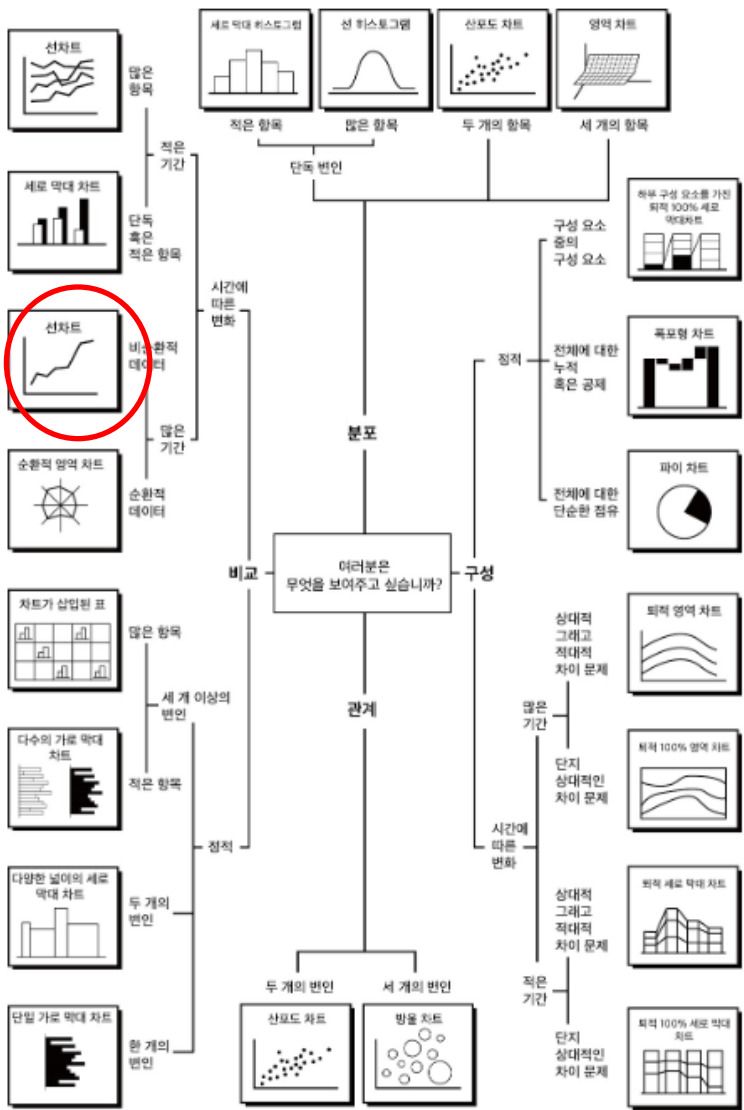
R Studio

RStudio® is a trademark of RStudio, Inc. • CC BY SA RStudio • info@rstudio.com • 844-448-1212 • rstudio.com • Learn more at <http://ggplot2.tidyverse.org> • ggplot2 3.3.0 • Updated: 2018-12

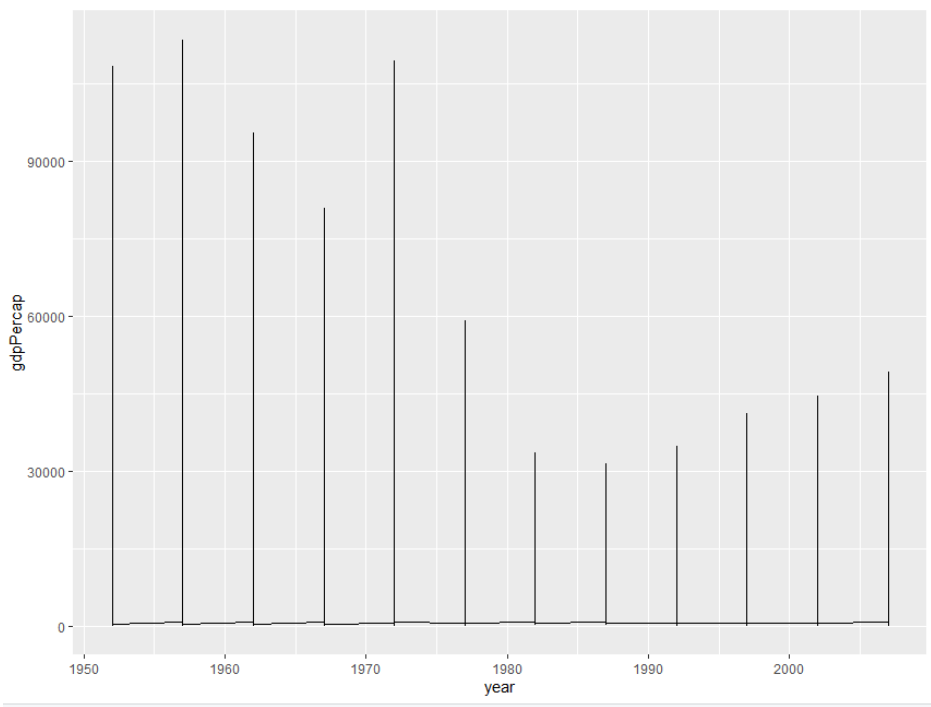


R 시각화 : ggplot

- 연도별 경제수준의 차이를 알아보자 : 시간에 따른 비교 ➔ 라인차트



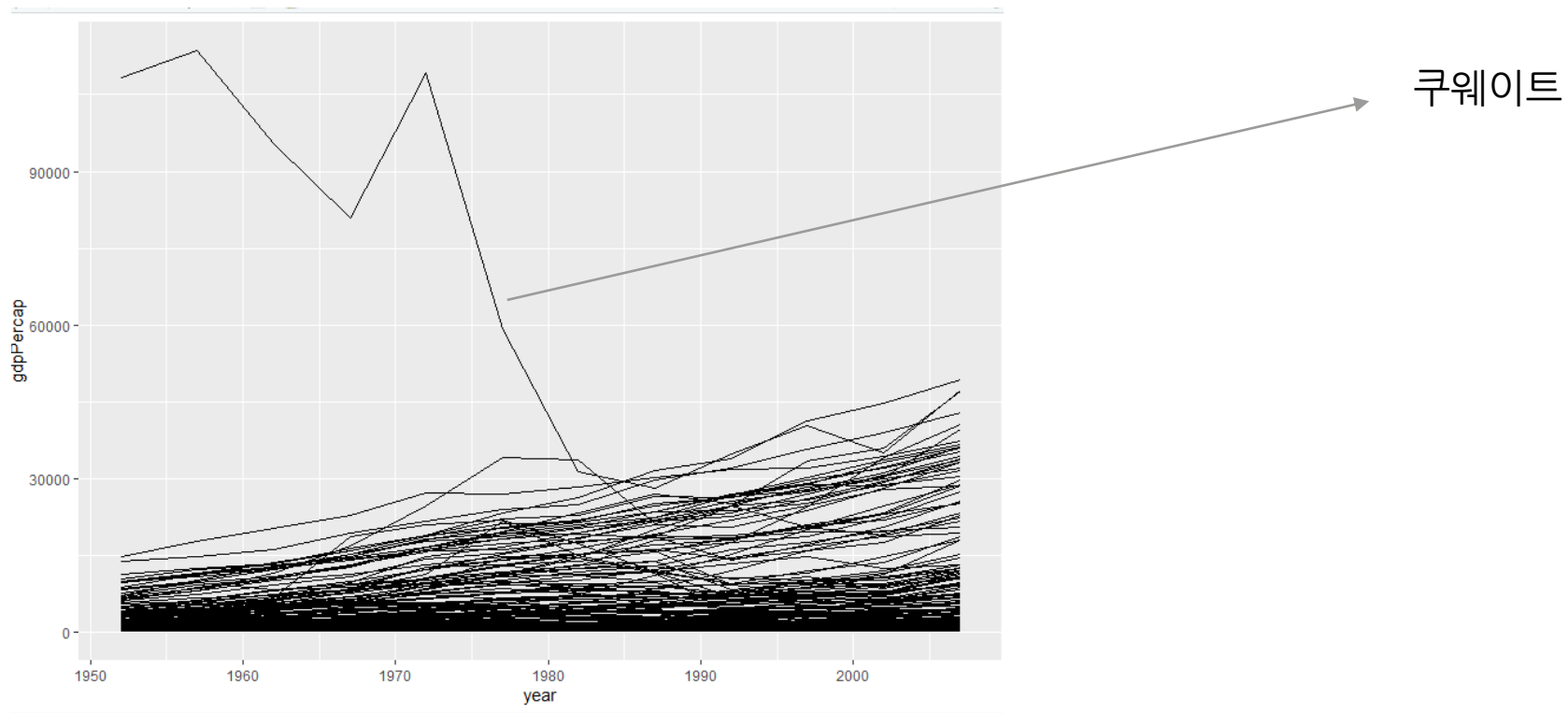
```
q <- ggplot(data = gapminder,
             mapping = aes(x = year, y = gdpPercap))
q + geom_line()
```



R 시각화 : ggplot

연도별 경제수준의 차이

- 데이터 구조를 보면, 연간 관측치가 국가별로 그룹이 되어 있음
- geom_line()함수는 같은 국가에 대해 연도별로 그룹화를 하지 않고, 데이터 셋에 있는 연도별 관측값 순서대로 결합하여 라인화 함
- 연도별로 국가가 그룹화되어 있음을 인지 시키는 과정이 필요함



R 시각화 : ggplot

- 연도별 **대륙별** 경제수준의 차이

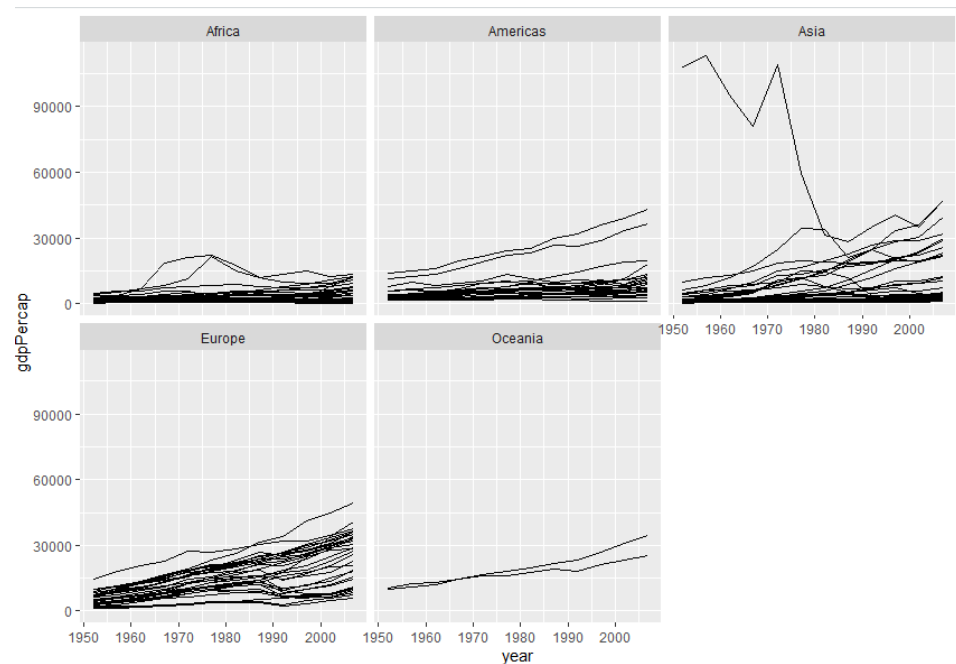
=> 소규모 다중 도표르 만드는 패싯(facet)

Faceting

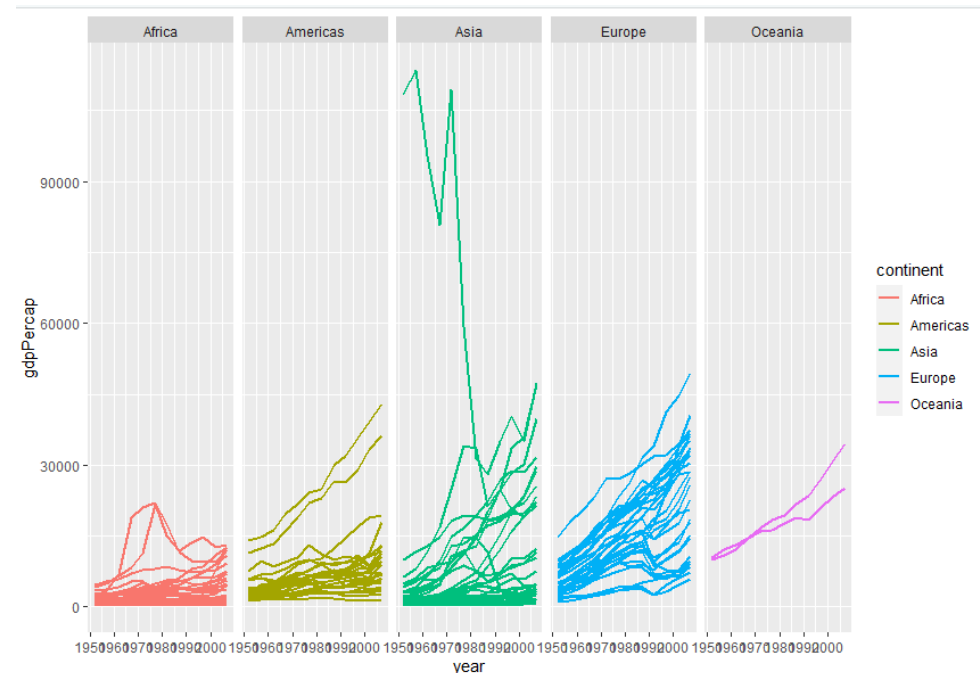
- Facets divide a plot into subplots based on the values of one or more discrete variables
: 전체 플롯을 하나 또는 그 이상의 이산형 변수로 플롯을 나누는 것
- facet_wrap(), facet_grid() 함수 : 요인 변수로 쪼개진 데이터에 대해 시각화해서 집합으로 묶어 그리기
- facet_grid()
 - ✓ facet into columns based on "~ x", " y ~ ", or "y~ x" : 가로 세로 지정
- facet_wrap()
 - ✓ wrap facets into a rectangular layout : 직사각형 형태로
 - ✓ ncol : 컬럼 나눌 수

R 시각화 : ggplot

```
q <- ggplot(data = gapminder,
  mapping = aes(x = year, y = gdpPercap))
  + geom_line(aes(group = country))
+ facet_wrap(~ continent)
```



```
q <- ggplot(data = gapminder,
  mapping = aes(x = year, y = gdpPercap))
  + geom_line(aes(group = country))
+ facet_wrap(~ continent, ncol=5)
```

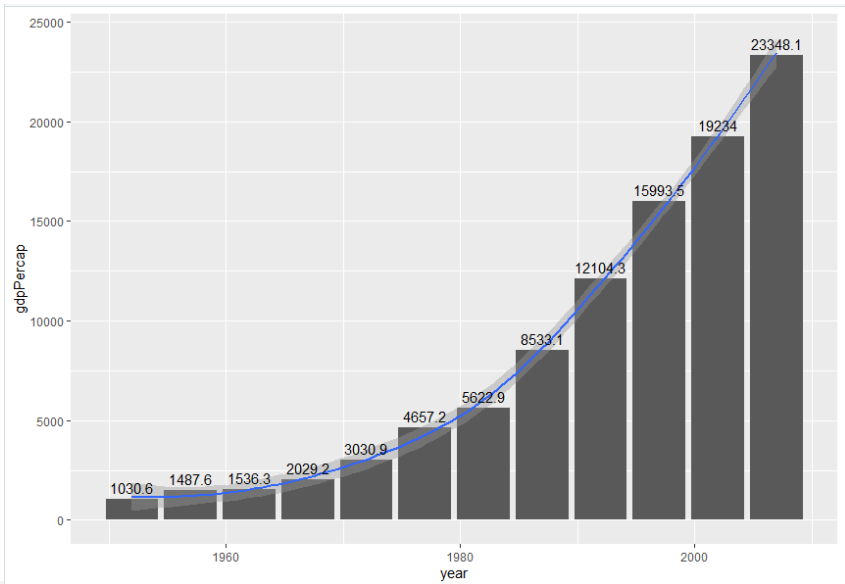


```
q <- ggplot(data = gapminder,
  mapping = aes(x = year, y = gdpPercap, color=continent))
```

R 시각화 : ggplot

- 1953년 부터, 2007년까지 한국의 경제수준을 그래프화 하기

- 1) 막대그래프와 추세선을 그리기 (geom_bar, geom_smooth())
- 2) Gapminder 데이터 셋에서 한국만 가져오기 (country = Korea, Rep.)
- 3) x축: 년도 , y축 : 경제수준 (gdpPercap)
- 4) bar 높이 : gdpPercap (not count)
- 5) 추세선 그리기
- 6) 막대에 경제수준정도를 표현하기



```
q1 <- ggplot(data = (gapminder %>% filter (country =='Korea, Rep.')),  
             mapping = aes(x = year, y=gdpPercap ))  
  
q1+geom_bar(stat='identity')+  
geom_smooth() +  
geom_text(aes(y=gdpPercap, label=round(gdpPercap,1),vjust = -0.5))
```


소수점 첫째 자리형식으로 표현 텍스트위치

R 시각화 : ggplot

- 1952년과 2007년의 아시아 국가별 경제 수준(1인당 GDP) 의 변화를 알아보자

막대그래프를 geom_bar를 이용하여 그린 후, animation

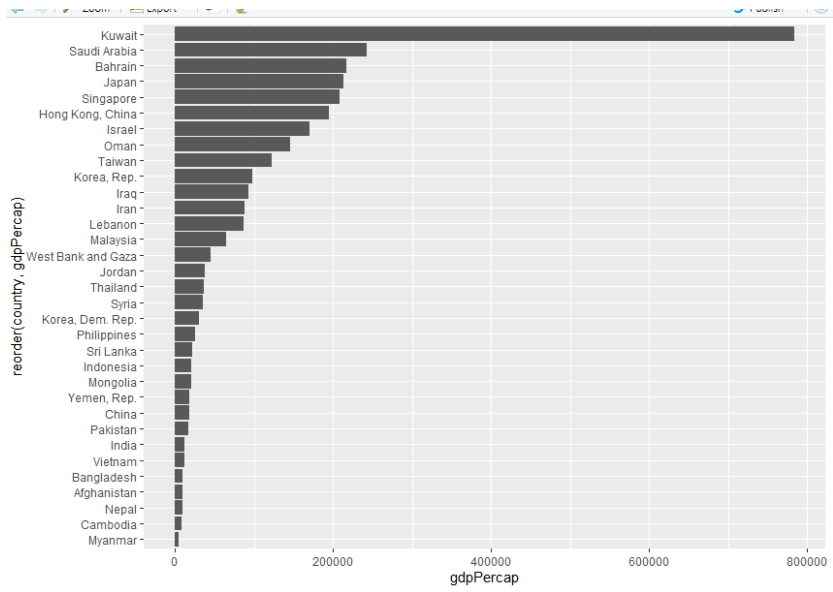
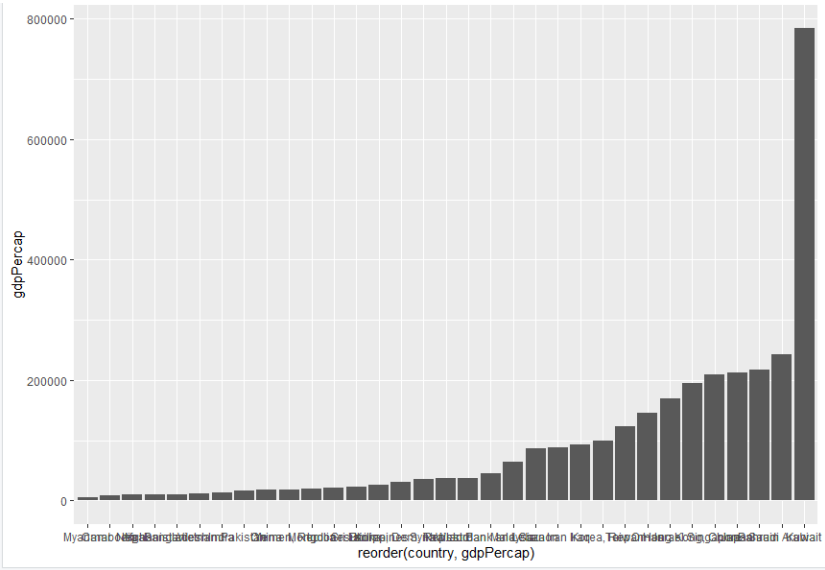
```
q3 <- ggplot(data = (gapminder %>% filter (continent == 'Asia' ) ),  
  mapping = aes(reorder(country,gdpPercap),gdpPercap))
```

county를  gdpPercap 크기 순으로 재배치 하기

x축과 y축 바꾸기

```
q3+geom_bar(stat='identity')
```

```
q3+geom_bar(stat='identity ' ) +coord_flip()
```

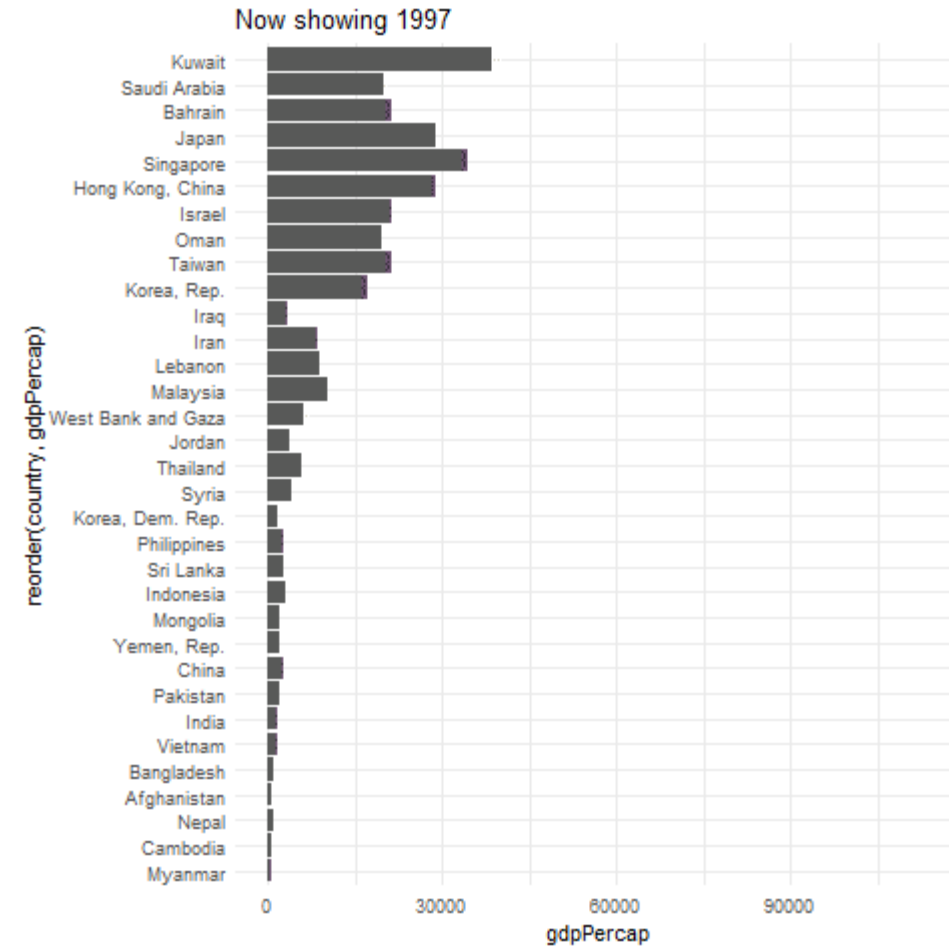


R 시각화 : ggplot

- 아시아의 국가들의 1인당 GDP의 변화를 살펴본다

```
ani_q3 <- q3+geom_bar(stat='identity') +coord_flip()+
  theme_minimal()+ # 배경
  # 애니메이션 길이
  transition_states(year,
    state_length=2)+
  ggtitle('Now showing {closest_state}')
```

ani_q3





R 시각화 : ggplot

지도 그리기

- 단계 구분도
 - 지도에 지역별 특성을 색깔로 표현한 지도를 단계 구분도라 함.
 - 단계 구분도를 보면 인구나 소득 같은 특성이 지역별로 얼마나 차이가 있는 지 용이하게 알 수 있음

- 필요 패키지

단계구분도

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

지도 데이터

```
install.packages("maps")
install.packages("mapproj")
library(mapproj)
library(maps)
```

- 데이터 : USArrests

미국 주별 범죄 데이터 (4개의 범죄 종류, 50개주 데이터)

```
data(USArrests)
USArrests <- data.frame(USArrests)
View(USArrests)
str(USArrests)
```

```
'data.frame': 50 obs. of 4 variables:
 $ Murder : num 13.2 10 8.1 8.8 9 7.9 3.3 5.9 15.4 17.4 ...
 $ Assault : int 236 263 294 190 276 204 110 238 335 211 ...
 $ UrbanPop: int 58 48 80 50 91 78 77 72 80 60 ...
 $ Rape : num 21.2 44.5 31 19.5 40.6 38.7 11.1 15.8 31.9 25.8 ...
> View(USArrests)
```

	Murder	Assault	UrbanPop	Rape
Alabama	13.2	236	58	21.2
Alaska	10.0	263	48	44.5
Arizona	8.1	294	80	31.0
Arkansas	8.8	190	50	19.5
California	9.0	276	91	40.6
Colorado	7.9	204	78	38.7
Connecticut	3.3	110	77	11.1
Delaware	5.9	238	72	15.8
Florida	15.4	335	80	31.9
Georgia	17.4	211	60	25.8
Hawaii	5.3	46	83	20.2
Idaho	2.6	120	54	14.2
Illinois	10.4	249	83	24.0
Indiana	7.2	113	65	21.0
Iowa	2.2	56	57	11.3
Kansas	6.0	115	66	18.0

R 시각화 : ggplot

지도 그리기

- 행 이름을 컬럼변수(state)로 변환, crime이라는 새로운 데이터 셋 생성

```
crime <- rownames_to_column(USArrests, var="state")
str(crime)
```

```
'data.frame':  50 obs. of  5 variables:
 $ state   : chr  "Alabama" "Alaska" "Arizona" "Arkansas" ...
 $ Murder  : num  13.2 10 8.1 8.8 9 7.9 3.3 5.9 15.4 17.4 ...
 $ Assault : int   236 263 294 190 276 204 110 238 335 211 ...
 $ UrbanPop: int    58 48 80 50 91 78 77 72 80 60 ...
 $ Rape    : num   21.2 44.5 31 19.5 40.6 38.7 11.1 15.8 31.9 25.8 ...
```

- 지도 데이터의 주 이름이 소문자이므로 state 컬럼변수의 값을 소문자로 수정

```
crime$state <- tolower(crime$state)
head(crime)
```

- 지도 데이터

```
states_map <- map_data("state")
View(states_map)
```

	long	lat	group	order	region	subregion
38	-88.01770	30.00221	1	38	alabama	NA
39	-88.03497	30.79075	1	39	alabama	NA
40	-88.04642	30.75638	1	40	alabama	NA
41	-88.05215	30.72773	1	41	alabama	NA
42	-88.05215	30.71054	1	42	alabama	NA
43	-88.06361	30.68762	1	43	alabama	NA
44	-88.06934	30.68189	1	44	alabama	NA
45	-88.08080	30.63033	1	45	alabama	NA
46	-88.08080	30.61314	1	46	alabama	NA
47	-88.09799	30.60741	1	47	alabama	NA
48	-88.10944	30.59595	1	48	alabama	NA
49	-88.11518	30.58449	1	49	alabama	NA
50	-88.10944	30.55584	1	50	alabama	NA
51	-88.12091	30.48136	1	51	alabama	NA
52	-88.12091	30.44125	1	52	alabama	NA
53	-88.12664	30.38968	1	53	alabama	NA
54	-88.12337	30.36183	1	54	alabama	NA

R 시각화 : ggplot

지도 그리기

단계 구분도 시각화

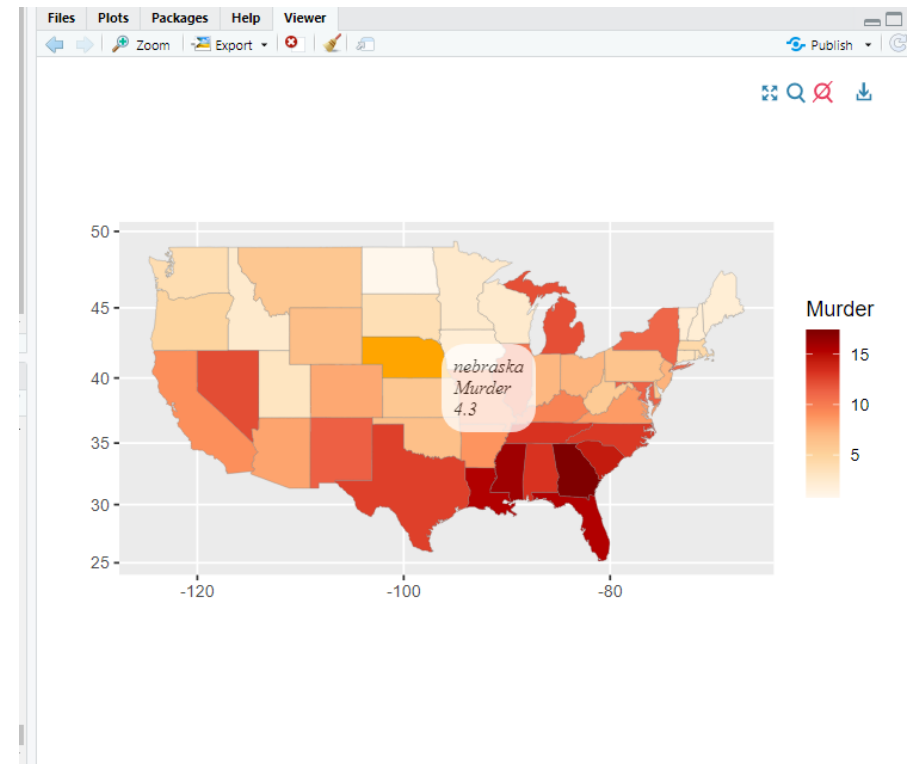
`ggChoropleth(data, aes(fill=, map_id=, map =, interactive =))`

1) `map_id` : 지도로 시각화 할 data와 지도 정보(예. state별 위도, 경도등의 자료) 위치정보가 있는 map 데이터를 연결 할 수 있는 변수 이름

2) `fill` : 지도의 각 그룹을 색깔로 채울 변수

3) `interactive` : 마우스오버 시, 데이터 표시 여부

```
ggChoropleth(data=crime, aes(fill=Murder, map_id=state),
map=states_map, interactive = T)
```



시각화 : 예시

숙명여자대학교 대학 IR 센터

<https://public.tableau.com/views/36780/sheet2?%3AshowVizHome=no&%3Aembed=true>

