2강. 다변량 시각화 (2)

- 단변량 그래프
- 이변량 그래프
- 다차원 그래프

1. 단변량 그래프

R 막대그림 및 원그림

◆ 데이터 읽기 (1강 : 자료 1-1)

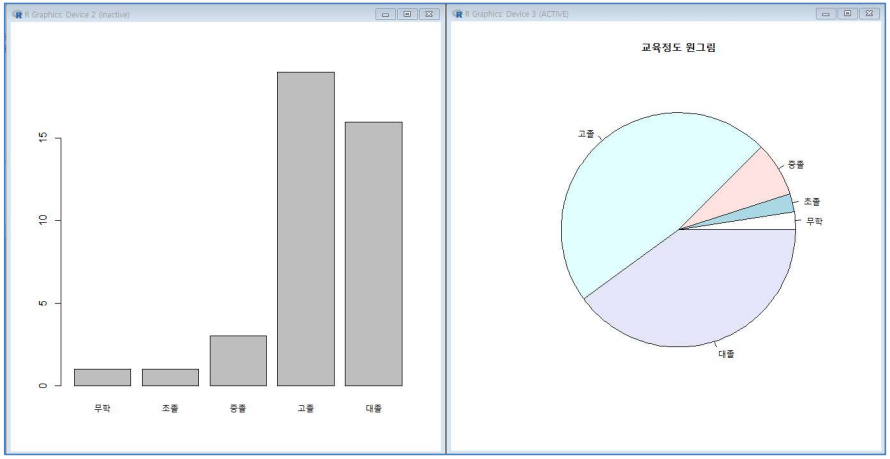
```
> survey = read.csv("c:/data/mva/survey.csv")
```

	A	B	C	D	E	F	G	H
1	seq	sex	marriage	age	job	edu	salary	
2	1	1	1	21	1	4	60	
3	2	1	1	22	5	5	100	
4	3	1	1	33	1	4	200	
5	4	2	2	33	7	4	120	
6	5	1	2	28	1	4	70	
7	6	1	1	21	5	5	80	
8	7	2	2	39	7	4	190	
9	8	1	1	32	1	4	100	
10	9	1	2	44	3	1	120	
11	10	1	2	55	4	4	110	
12	11	2	2	46	7	5	150	
13	12	1	1	20	1	4	50	
14	13	1	2	31	6	4	210	
15	14	1	1	27	1	4	60	
16	15	2	1	21	5	5	80	

survey

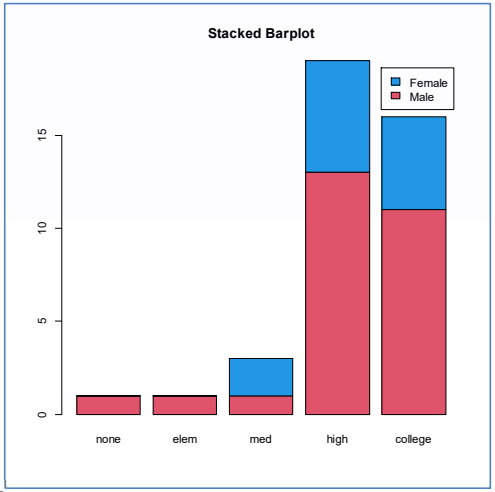
R 막대그림 및 원그림

```
> edu_tb = table(survey$edu)
> edu_tb
  none   elem    med   high college
     1      1      3     19      16
> rownames(edu_tb) = c("무학","초졸", "중졸","고졸","대졸")
> barplot(edu_tb)
> dev.new()
> pie(edu_tb, main="교육정도 원그림")
> dev.off()
```



R 겹친막대그림

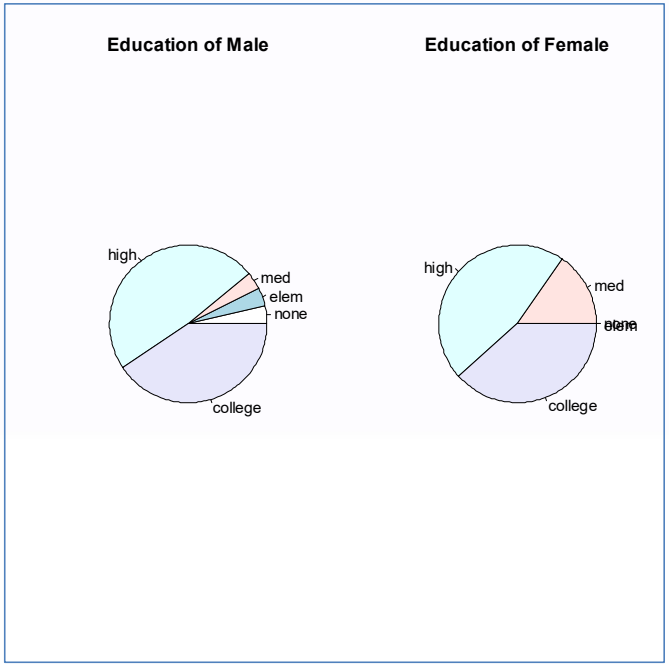
```
>  sex_edu = list(survey$sex, survey$edu)
>  sex_edu_tb = table(sex_edu)
> sex_edu_tb
        sex_edu.2
sex_edu.1 none elem med high college
   Male      1    1   1   13       11
   Female    0    0   2    6        5
>  barplot(sex_edu_tb, legend.text=rownames(sex_edu_tb), col=c(2,4))
> title("Stacked Barplot")
```



한 화면에 여러 개의 그림 그리기 : par문 이용

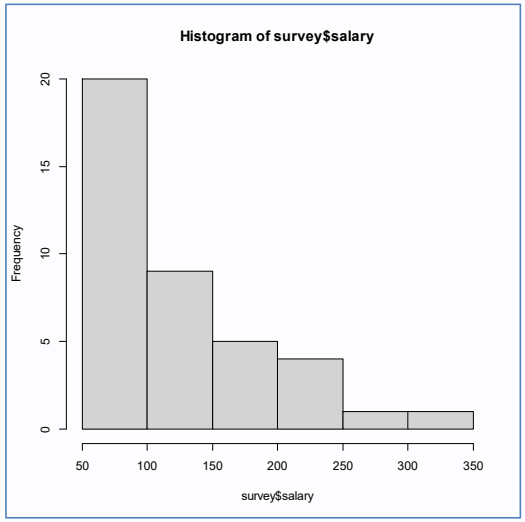
- 남녀별로 교육정도의 원그림 그리기

```
> par(mfrow=c(1,2))
> pie(sex_edu_tb[1,])
> title("Education of Male")
> pie(sex_edu_tb[2,])
> title("Education of Female“)
```



R 히스토그램, 줄기-잎 그림

```
> hist(survey$salary)
```

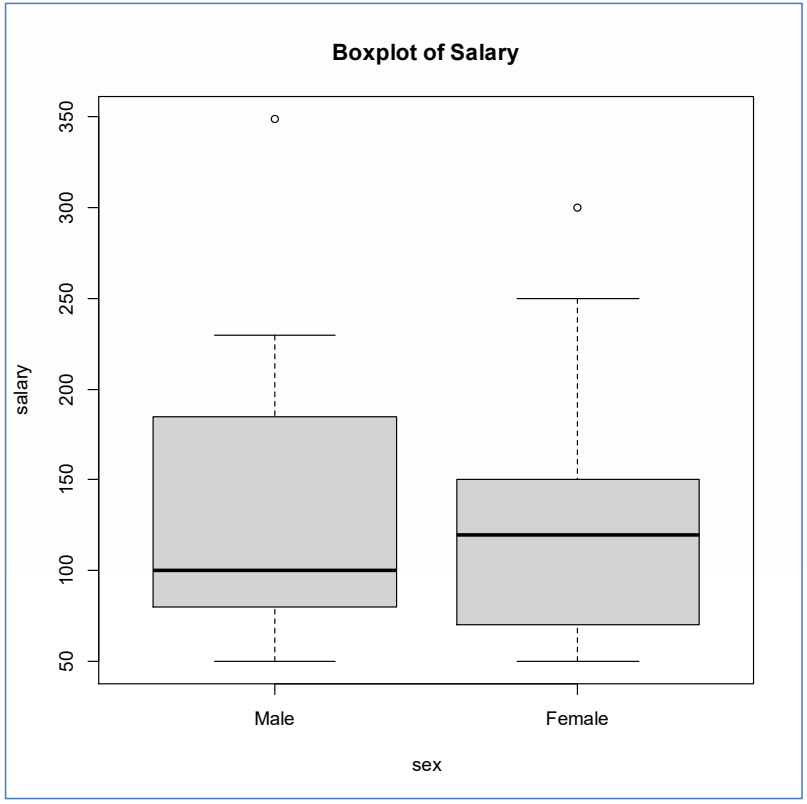



```
> stem(survey$salary)

  The decimal point is 2 digit(s) to the right of the |

  0 | 555666677788889
  1 | 00000122233
  1 | 55579
  2 | 000123
  2 | 5
  3 | 0
  3 | 5
```

```
> stem(survey$salary, scale=2)

  The decimal point is 1 digit(s) to the right of the |

   4 | 000
   6 | 0000000
   8 | 00000
  10 | 000000
  12 | 00000
  14 | 000
  16 | 0
  18 | 0
  20 | 0000
  22 | 00
  24 | 0
  26 |
  28 |
  30 | 0
  32 |
  34 | 9
```

R 상자그림

```
> boxplot(salary ~ sex, data=survey)
> title("Boxplot of Salary")
```

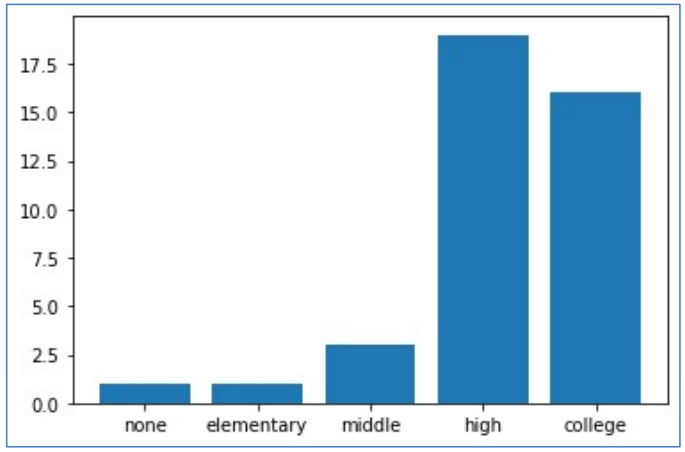


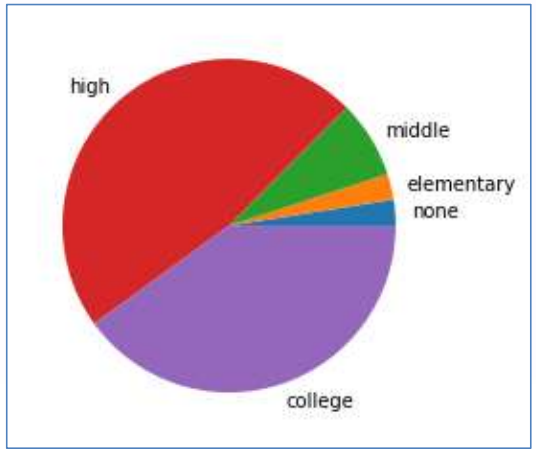
파이썬 막대그림 및 원그림

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
# 데이터 읽기
survey = pd.read_csv("c:/data/mva/survey.csv")
# 빈도수 구하기
edu_freq = pd.crosstab(index=survey.edu, columns='count')
edu_freq
Out[2]:
col_0  count
edu
1          1
2          1
3          3
4         19
5         16
# 케이스 라벨 지정하기
edu_freq.index = ["none", "elementary", "middle", "high", "college"]
```

파이썬 막대그림 및 원그림

```
# 막대그림 그리기
plt.bar(edu_freq.index, edu_freq["count"])

# 원그림 그리기
plt.pie(edu_freq["count"], labels=edu_freq.index)
```





파이썬 겹친막대그림

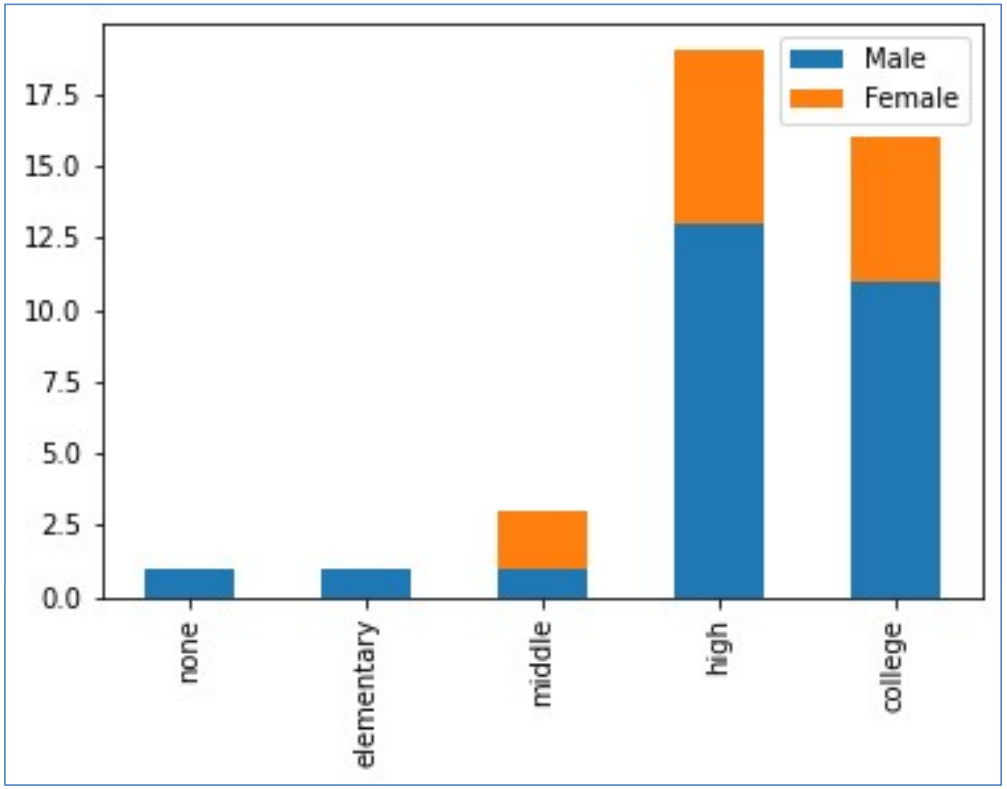
```
# (edu, sex) 분할표 구하기
edu_sex_tb = pd.crosstab(index=survey.edu, columns=survey.sex)

edu_sex_tb
Out[8]:
sex    1   2
edu
1      1   0
2      1   0
3      1   2
4     13   6
5     11   5
```

```
edu_sex_tb
Out[9]:
             Male  Female
none            1       0
elementary      1       0
middle          1       2
high           13       6
college        11       5
```

```
# 케이스 및 변수이름 지정하기
edu_sex_tb.index = ["none", "elementary", "middle", "high", "college"]
edu_sex_tb.columns = ["Male", "Female"]
```

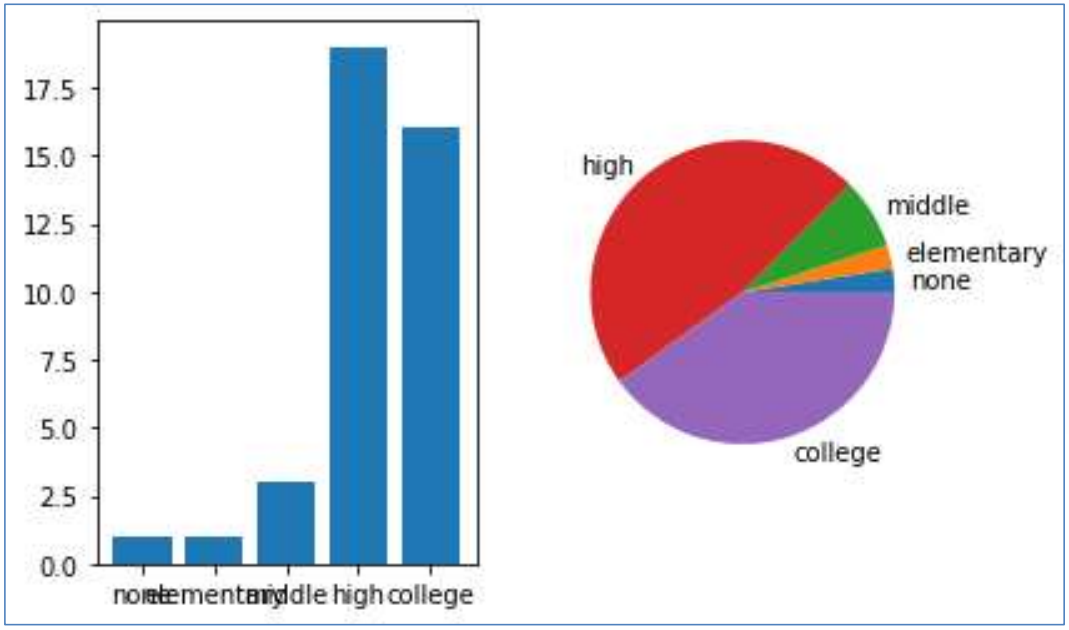
파이썬 겹친막대그림

```
# 겹친 막대그림 그리기
edu_sex_tb.plot.bar(stacked=True)
```



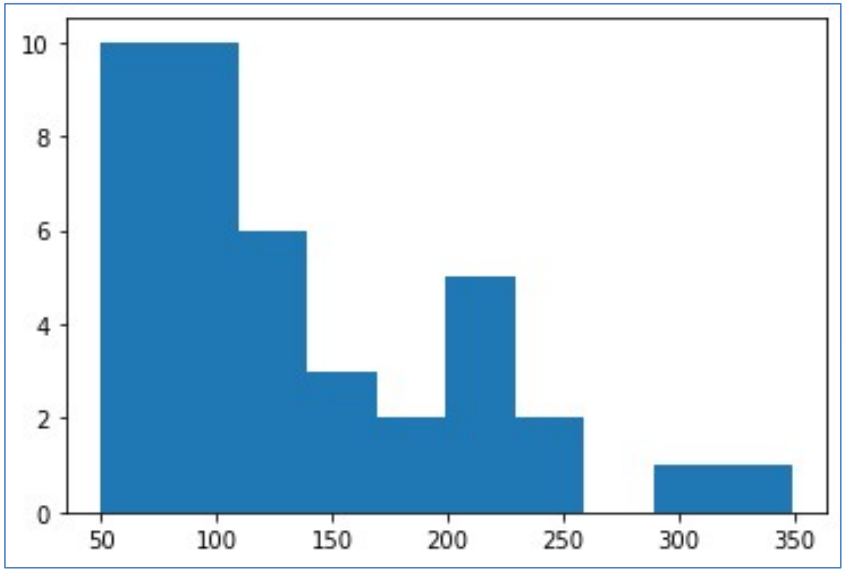
파이썬 한 화면에 여러 개의 그림 그리기

```
plt.figure()
plt.subplot(121)
plt.bar(edu_freq.index, edu_freq["count"])
plt.subplot(122)
plt.pie(edu_freq["count"], labels=edu_freq.index)
```



파이썬 히스토그램, 줄기-잎 그림

```
import matplotlib.pyplot as plt
# 히스토그램 그리기
plt.hist(survey["salary"])
Out[14]:
(array([10., 10.,  6.,  3.,  2.,  5.,  2.,  0.,  1.,  1.]),
 array([ 50. ,  79.9, 109.8, 139.7, 169.6, 199.5, 229.4, 259.3, 289.2,
        319.1, 349. ]),
```



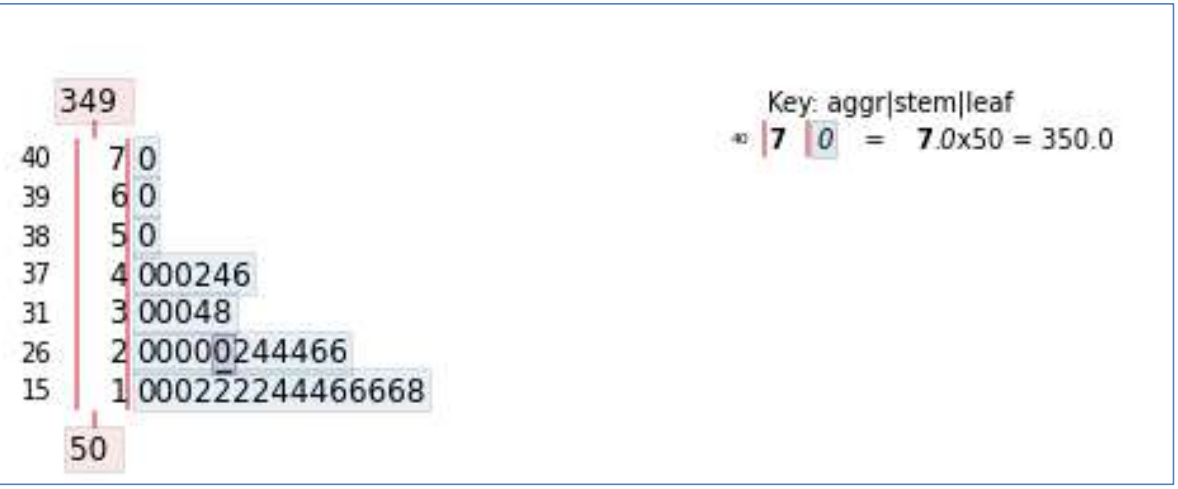
help(plt.hist)
Help on function hist in module matplotlib.pyplot:

hist(x, bins=None, range=None, density=False, weights=None, cumulative=False, bottom=None, histtype='bar', align='mid', orientation='vertical', rwidth=None, log=False, color=None, label=None, stacked=False, *, data=None, **kwargs)
...

Returns
----
n : array or list of arrays
bins : array
The edges of the bins.

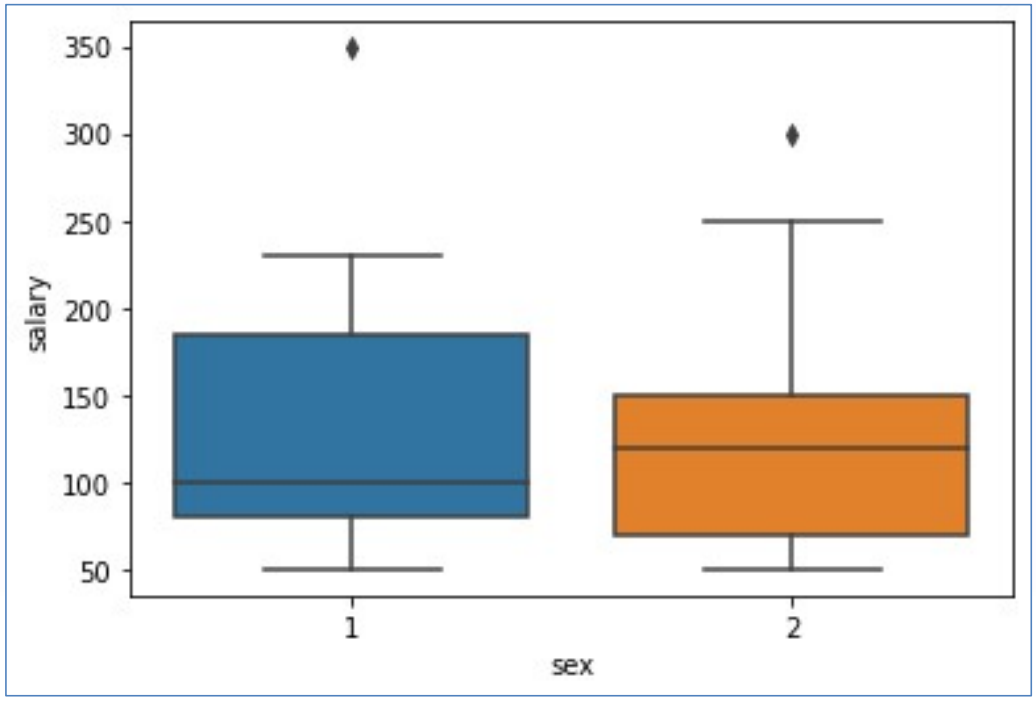
파이썬 줄기-잎 그림

```
# 줄기-잎 그림 그리기
# pip install stemgraphic (in DOS prompt)
import stemgraphic
stemgraphic.stem_graphic(survey.salary, scale=50)
```



파이썬 상자그림

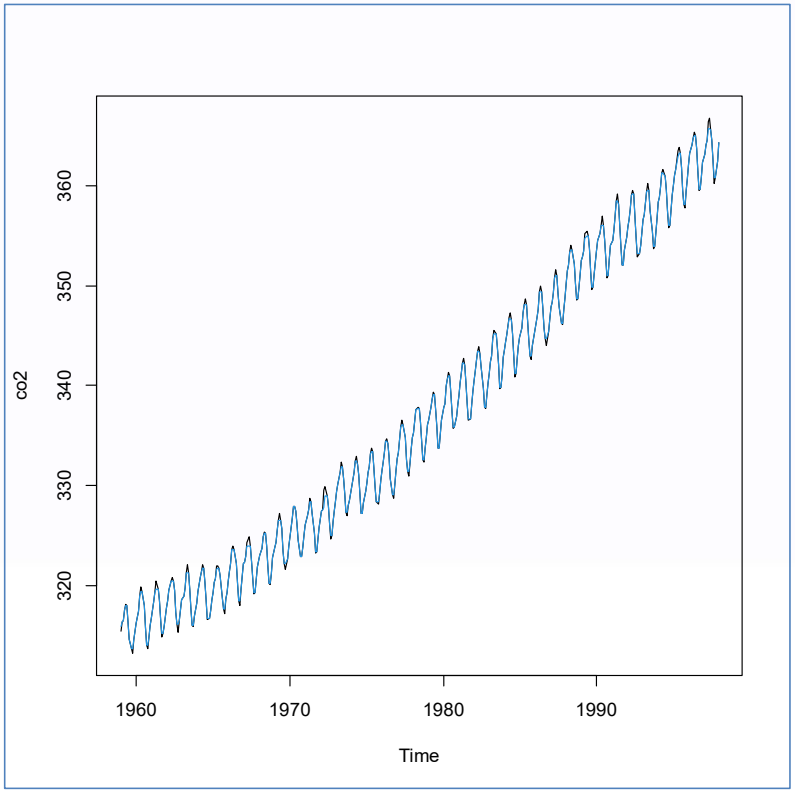
```
import seaborn as sns
sns.boxplot(x="sex", y="salary", data=survey)
```



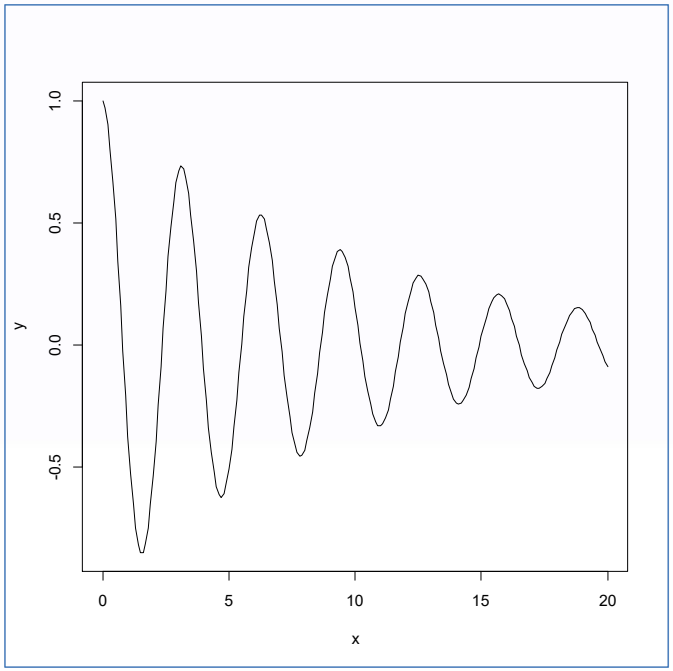
2. 이변량 그래프

R 이변량 그래프

```
# plot using lines
plot(co2)
lines(smooth(co2),col="BLUE")
```

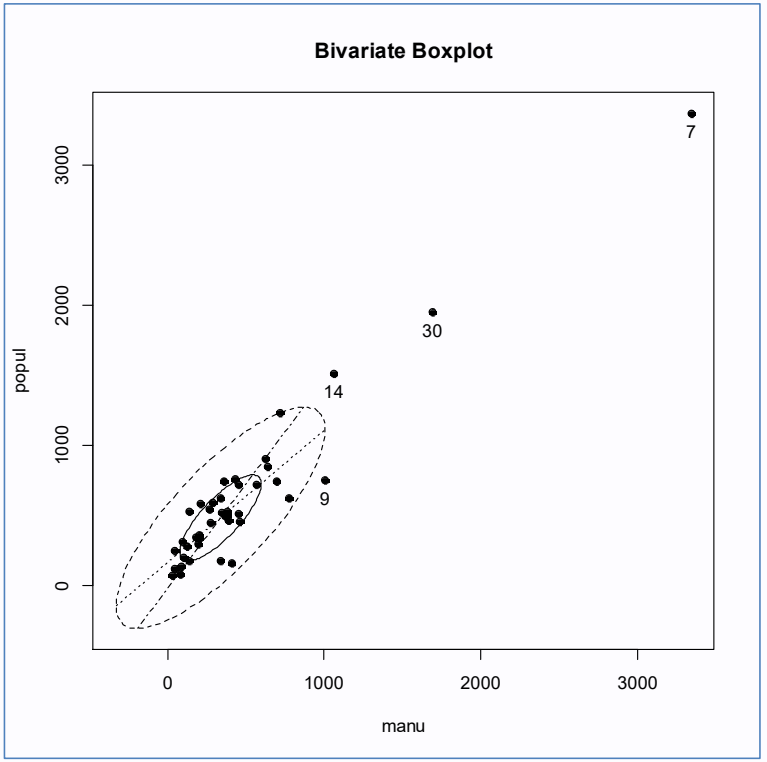



```
# plot of mathematical functions
x <- seq(0, 20, 0.1)
y <- exp(-x/10)*cos(2*x)
plot(x,y,type="l")
```



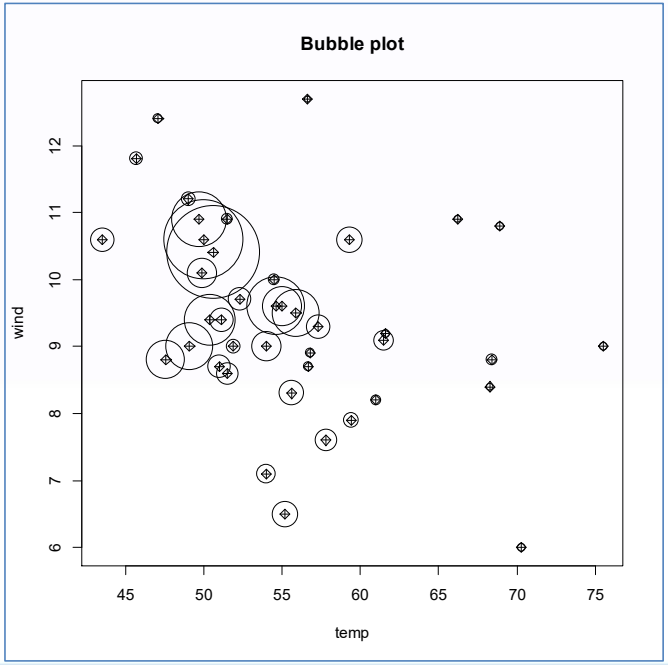
R bivariate boxplot

```
> install.packages("HSAUR2")
> library(HSAUR2)
> install.packages("MVA")
> library(MVA)
> data(USairpollution)
> head(USairpollution, 3)
             SO2 temp manu popul wind precip predays
Albany        46 47.6   44   116  8.8  33.36     135
Albuquerque   11 56.8   46   244  8.9   7.77      58
Atlanta       24 61.5  368   497  9.1  48.34     115
> x = USairpollution[, c(3,4)]
> bvbox(x, xlab="manu", ylab="popul", pch=19)
>  title("Bivariate Boxplot")
> identify(x)
[1]  7  9 14 30
> rownames(x) [c(7,9,14,30)]
[1] "Chicago"    "Cleveland"  "Detroit"    "Philadelphia"
```



R Bubble plot

```
〉 plot(wind~temp, data=USairpollution, pch=9)
〉# symbols(USairpollution$temp, USairpollution$wind, USairpollution$circle=SO2,
〉# inches=0.5, add=T))
〉with(USairpollution, symbols(temp, wind, circle=SO2, inches=0.5, add=T))
〉title("Bubble plot")
〉
```

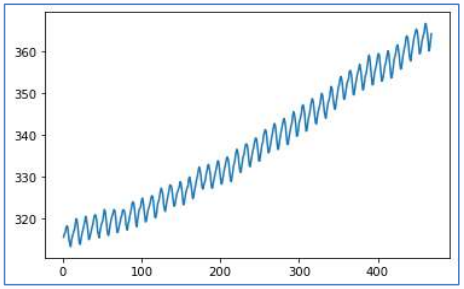


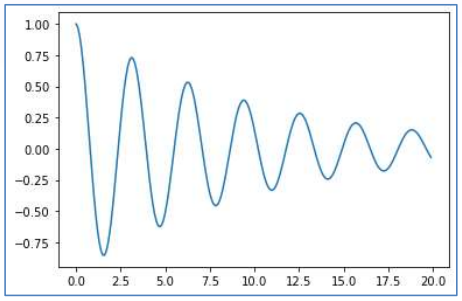
(temp, wind)의 산점도에 제3의 변수인 SO2의
정보의 크기에 따라 원으로 나타낸 그림

파이썬 이변량 그래프

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
# 데이터 읽기
co2 = pd.read_csv("c:/data/mva/co2.csv")
co2.head(2)
Out[18]:
   Unnamed: 0       x
0           1  315.42
1           2  316.31

# 변수이름 지정하기
co2.columns = ["seq", "x"]
co2.head(2)
Out[19]:
   seq       x
0    1  315.42
1    2  316.31
```

```
# 선그리기
plt.plot(co2.seq, co2.x)
# plot of mathematical functions
x = np.arange(0, 20, 0.1)
y = np.exp(-x/10)*np.cos(2*x)
plt.plot(x, y)
```



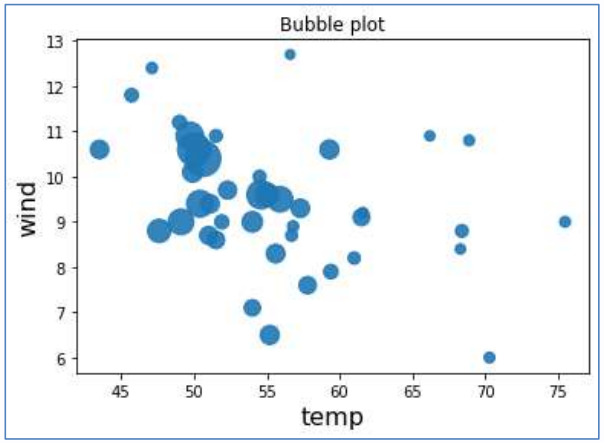


파이썬 Bubble plot

```
# 데이터 읽기
USairpollution = pd.read_csv("c:/data/mva/USairpollution.csv")
USairpollution.head(3)
```

Out[23]:

	state	SO2	temp	manu	popul	wind	precip	predays
0	Albany	46	47.6	44	116	8.8	33.36	135
1	Albuquerque	11	56.8	46	244	8.9	7.77	58
2	Atlanta	24	61.5	368	497	9.1	48.34	115




```
# SO2 변숫값 * 5
USairpollution["SO2"] = USairpollution["SO2"] * 5
# 버블차트 그리기
plt.scatter('temp', 'wind', s='SO2', alpha=0.9, data=USairpollution)
plt.xlabel("temp", size=16)
plt.ylabel("wind", size=16)
plt.title("Bubble plot")

# help(plt.scatter)
```

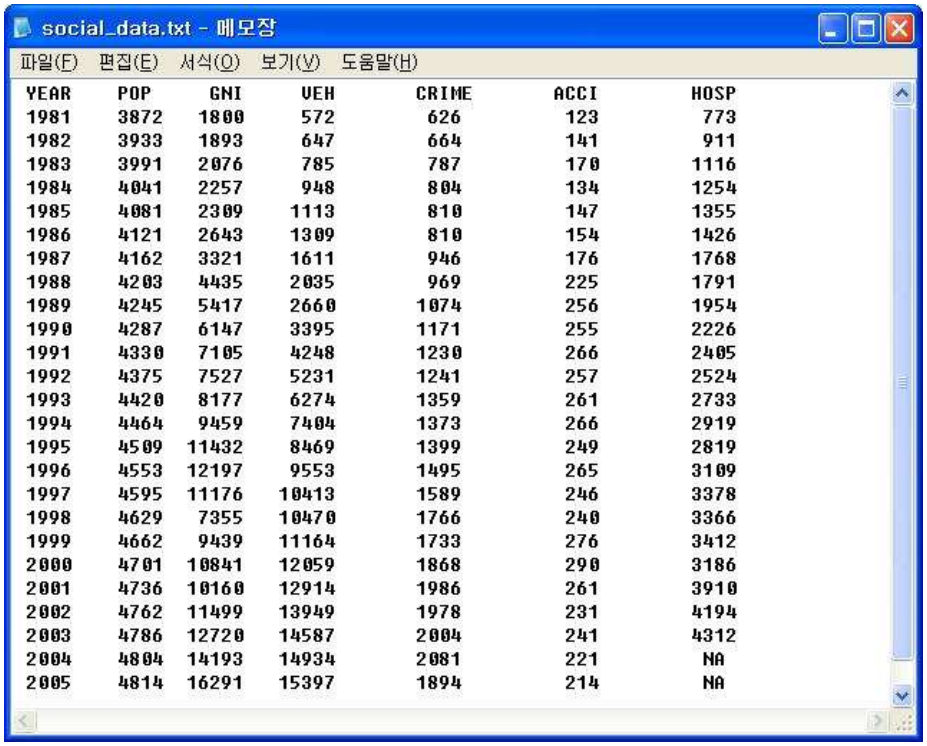
3. 다차원 그래프

산점도 행렬

예 제

■예제) 한국의 각종 사회 통계(2006 한국의 사회지표)에서 산점도행렬을 그려서 변수들 간의 관계를 살펴보아라.

⟨social.txt⟩



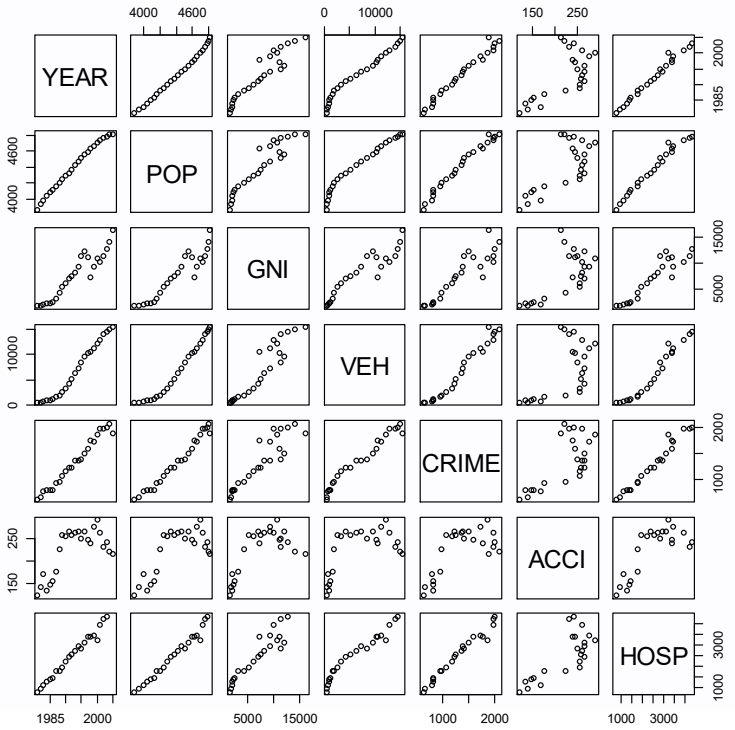
social_data.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

YEAR	POP	GNI	VEH	CRIME	ACCI	HOSP
1981	3872	1800	572	626	123	773
1982	3933	1893	647	664	141	911
1983	3991	2076	785	787	170	1116
1984	4041	2257	948	804	134	1254
1985	4081	2309	1113	810	147	1355
1986	4121	2643	1309	810	154	1426
1987	4162	3321	1611	946	176	1768
1988	4203	4435	2035	969	225	1791
1989	4245	5417	2660	1074	256	1954
1990	4287	6147	3395	1171	255	2226
1991	4330	7105	4248	1230	266	2405
1992	4375	7527	5231	1241	257	2524
1993	4420	8177	6274	1359	261	2733
1994	4464	9459	7404	1373	266	2919
1995	4509	11432	8469	1399	249	2819
1996	4553	12197	9553	1495	265	3109
1997	4595	11176	10413	1589	246	3378
1998	4629	7355	10470	1766	240	3366
1999	4662	9439	11164	1733	276	3412
2000	4701	10841	12059	1868	290	3186
2001	4736	10160	12914	1986	261	3910
2002	4762	11499	13949	1978	231	4194
2003	4786	12720	14587	2004	241	4312
2004	4804	14193	14934	2081	221	NA
2005	4814	16291	15397	1894	214	NA

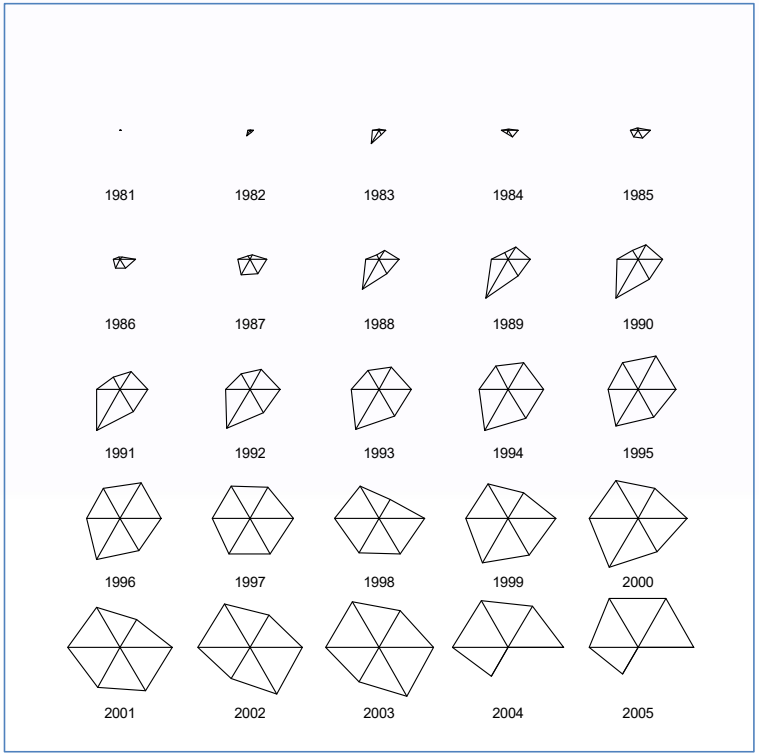
R 산점도 행렬

```
> social = read.table("c:/data/mva/social.txt", header=T)
> head(social, 3)
  YEAR  POP  GNI VEH CRIME ACCI HOSP
1 1981 3872 1800 572   626  123  773
2 1982 3933 1893 647   664  141  911
3 1983 3991 2076 785   787  170 1116

> pairs(social)

> round(cor(social, use="complete.obs"), 3)
        YEAR   POP   GNI   VEH CRIME  ACCI  HOSP
YEAR   1.000 0.998 0.935 0.981 0.993 0.788 0.991
POP    0.998 1.000 0.939 0.974 0.989 0.804 0.988
GNI    0.935 0.939 1.000 0.925 0.907 0.820 0.934
VEH    0.981 0.974 0.925 1.000 0.984 0.697 0.972
CRIME  0.993 0.989 0.907 0.984 1.000 0.762 0.983
ACCI   0.788 0.804 0.820 0.697 0.762 1.000 0.770
HOSP   0.991 0.988 0.934 0.972 0.983 0.770 1.000
```



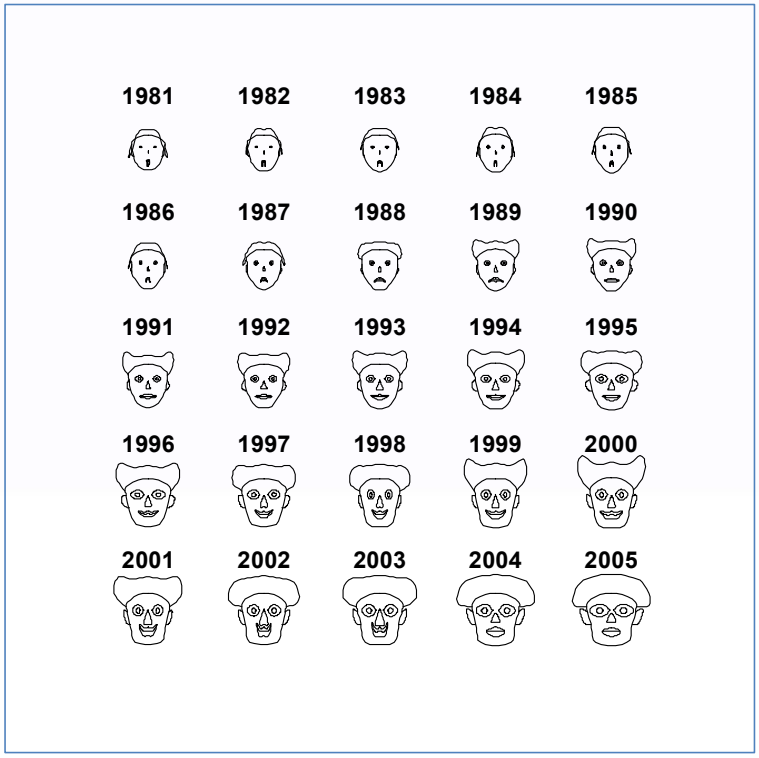
R 별그림(Star Plot)

```
> social2 = social[, -1]
> year = social[,1]
> rownames(social2) = year
> stars(social2)
```

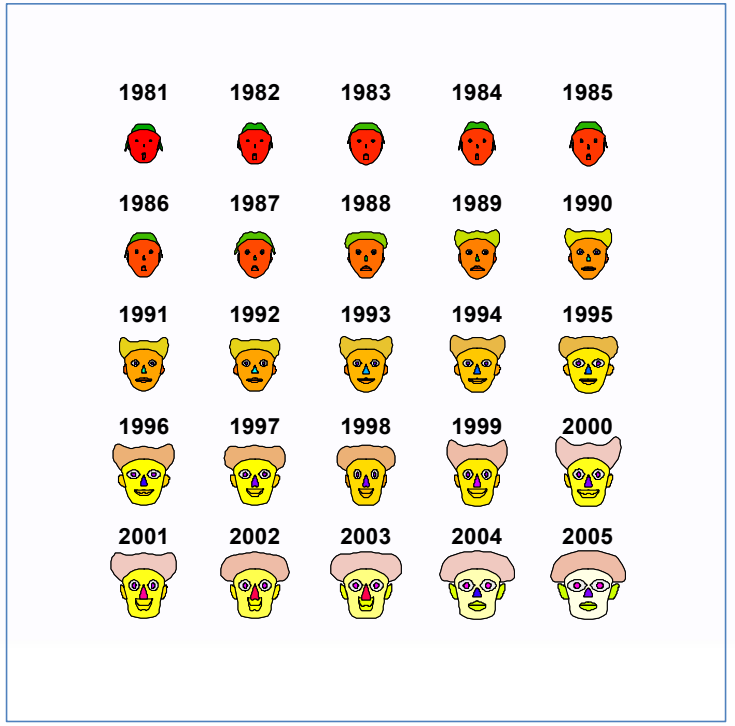


R 얼굴그림(faces plot)

```
> install.packages("aplpack")
> library(aplpack)
> # faces(social2, face.type=0, na.rm=TRUE)
> faces(social2, face.type=0)
```



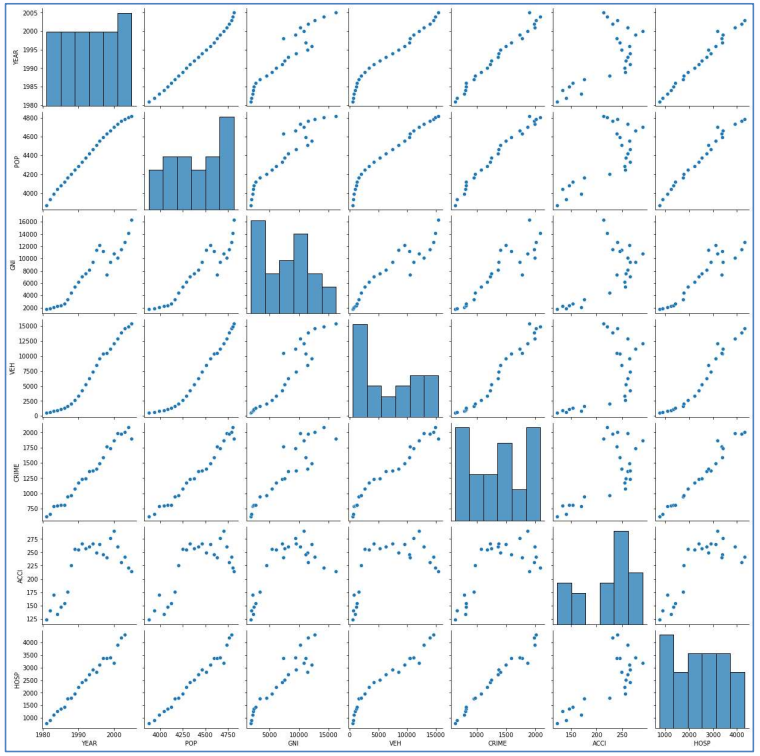

```
>  faces(social2, face.type=1)
```



파이썬 산점도 행렬

```
import pandas as pd
# 데이터 읽기
social = pd.read_csv("c:/data/mva/social.csv")

# (행의 수, 열의 수)
social.shape
Out[28]: (25, 7)

# seaborn을 이용하여 산점도행렬 그리기
import seaborn as sns
sns.pairplot(social)

# 상관계수 행렬 구하기 - 소수점 이하 3자리 반올림
round(social.corr(), 3)
Out[30]:
```

	YEAR	POP	GNI	VEH	CRIME	ACCI	HOSP
YEAR	1.000	0.996	0.948	0.985	0.989	0.680	0.991
POP	0.996	1.000	0.941	0.977	0.989	0.721	0.988
GNI	0.948	0.941	1.000	0.940	0.911	0.676	0.934
VEH	0.985	0.977	0.940	1.000	0.982	0.599	0.972
CRIME	0.989	0.989	0.911	0.982	1.000	0.683	0.983
ACCI	0.680	0.721	0.676	0.599	0.683	1.000	0.770
HOSP	0.991	0.988	0.934	0.972	0.983	0.770	1.000

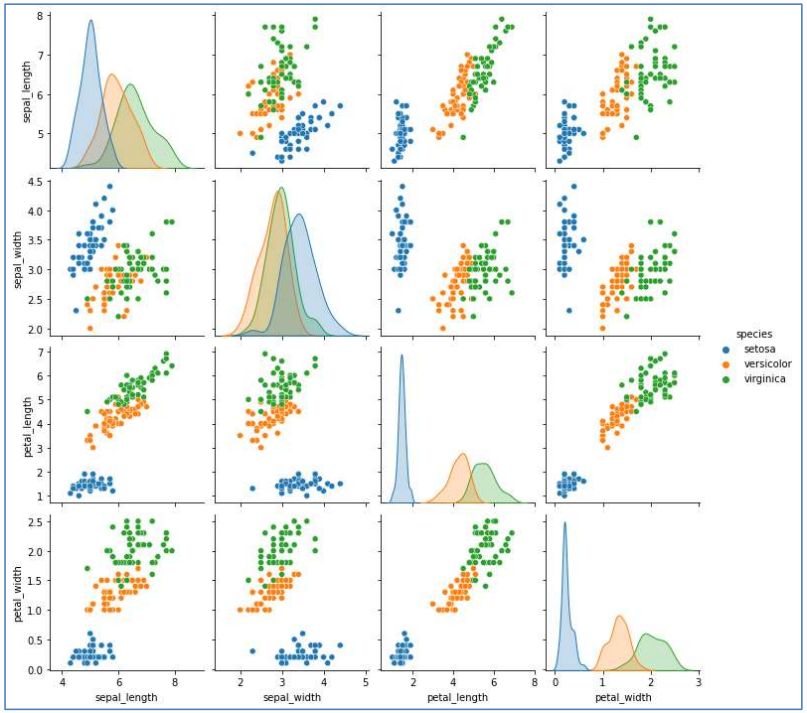


파이썬 iris 산점도 행렬

```
import seaborn as sns
# seaborn에 내장된 iris 데이터 가져오기
iris = sns.load_dataset("iris")
iris.head()
Out[31]:
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
# species로 구분된 산점도행렬 그리기 - 대각선은 각 그룹별 분포
sns.pairplot(iris, hue='species', height=2.5)
```



다음시간에는

3강 주성분분석

수고했습니다.