

Out[1]:

[Click here to toggle on/off the raw code.](#)

[One Point Tutorial] Visualization II - seaborn

Python을 활용한 데이터 시각화

December, 2019

2. **seaborn** 을 배워 보자 !

- **seaborn**이란?

Matplotlib을 기반으로 다양한 색상 테마와 통계용 차트 등의 기능을 추가한 시각화 패키지

학습 목표

seaborn 의 핵심적인 시각화 기법을 이해하고 활용함

목차

1. Scatter Plot
2. Box Plot
3. Violin Plot
4. Bar Plot
5. Rug plot
6. KDe Plot(Line Histogram)
7. Count Plot
8. Joint Plot
9. Pair Plot
10. Reg plot & LM Plot

Import module (모듈 설치 후 불러오기)

seaborn 모듈이 설치 안 되어 있다면, 설치부터 하기!

Current Working Directory is changed.

Import data

seaborn 패키지에는 데이터들이 내장되어 있다.
불러와보자!

Out[6]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

2.1. Scatter Plot

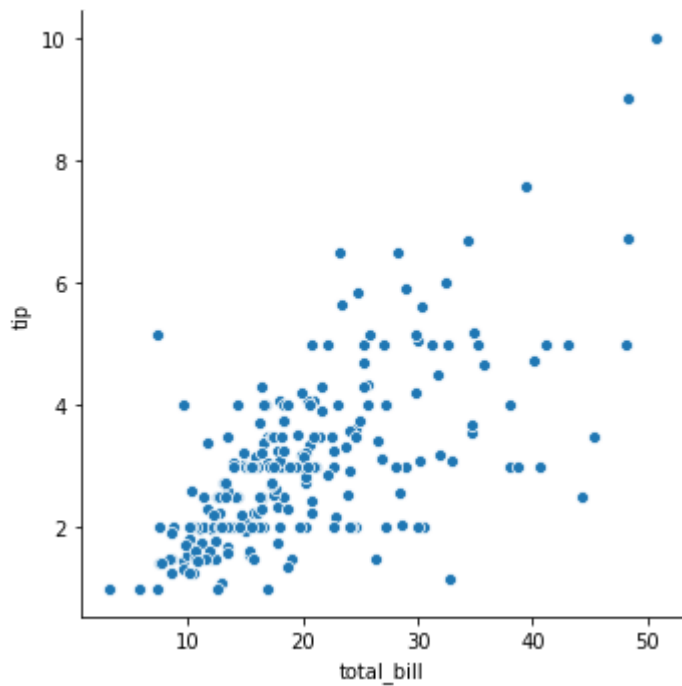
2.1.1. 첫번째 그래프: 분석하고자 하는 데이터가 numerical 일 때 !

사용 방법 :

1. sns.relplot(x축 데이터 , y축 데이터 , data)
2. sns.scatterplot(data , x축 데이터 , y축 데이터)

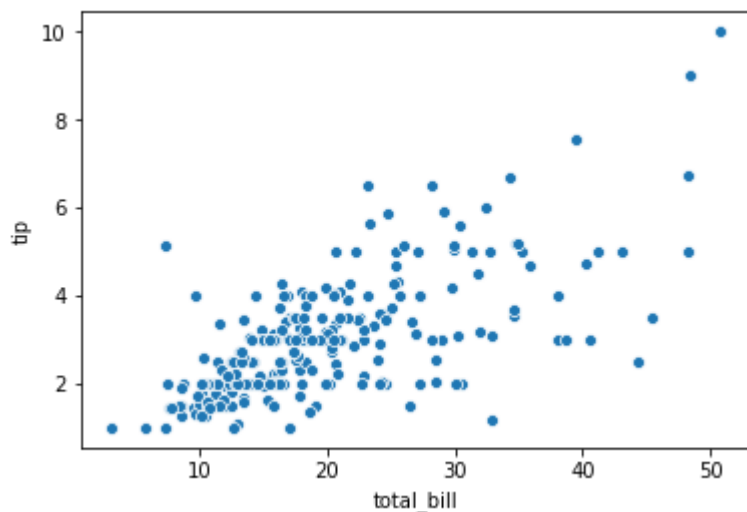
Out[8]:

<seaborn.axisgrid.FacetGrid at 0x1d51e5f3a58>



Out[9]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51e8d7cf8>

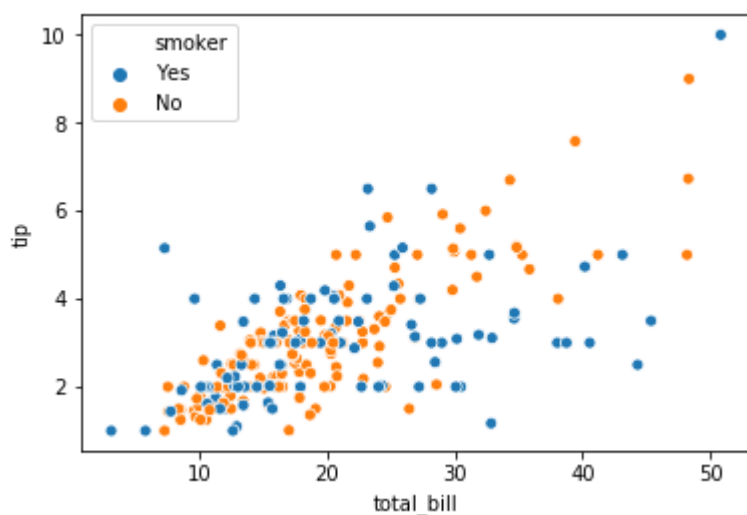


보이기에 따라서 조금 차이가 있지만 둘 다 scatter plot(산포도)을 그리고 있다.

- 카테고리형 변수가 섞여 있는 경우!
hue 파라미터에 카테고리 변수 이름을 지정하면 카테고리 값에 따라 색상이 달라짐
style 파라미터에 카테고리 변수 이름을 지정하면 카테고리 값에 따라 모양이 달라짐

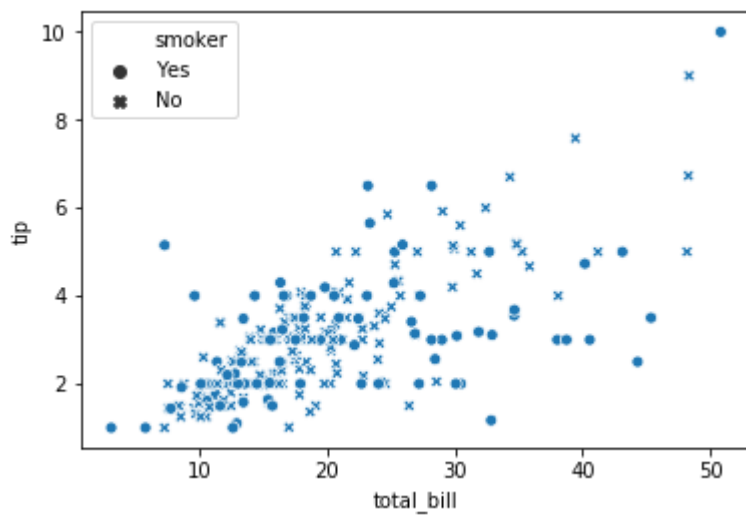
Out[11]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51e9854a8>



Out[12]:

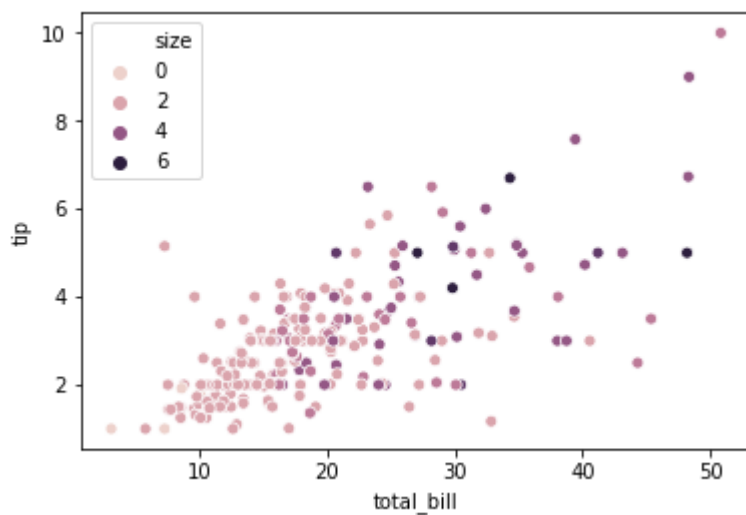
<matplotlib.axes._subplots.AxesSubplot at 0x1d51e9daf98>



- 실수형 데이터 또한 함께 표현하고 싶은 경우!
hue 에 numerical 변수 이름을 지정하면 변수 값에 따라 색상이 달라짐
size 에 numerical 변수 이름을 지정하면 변수 값에 따라 크기가 달라짐

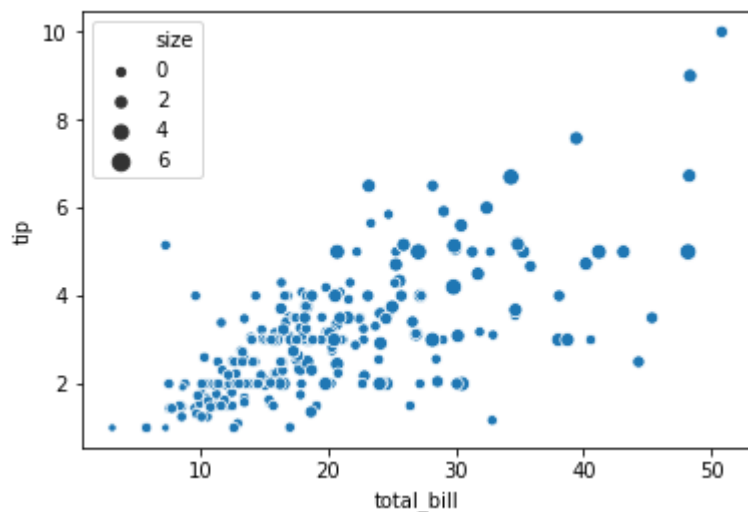
Out[13]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ea5b940>



Out[14]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ead8b00>



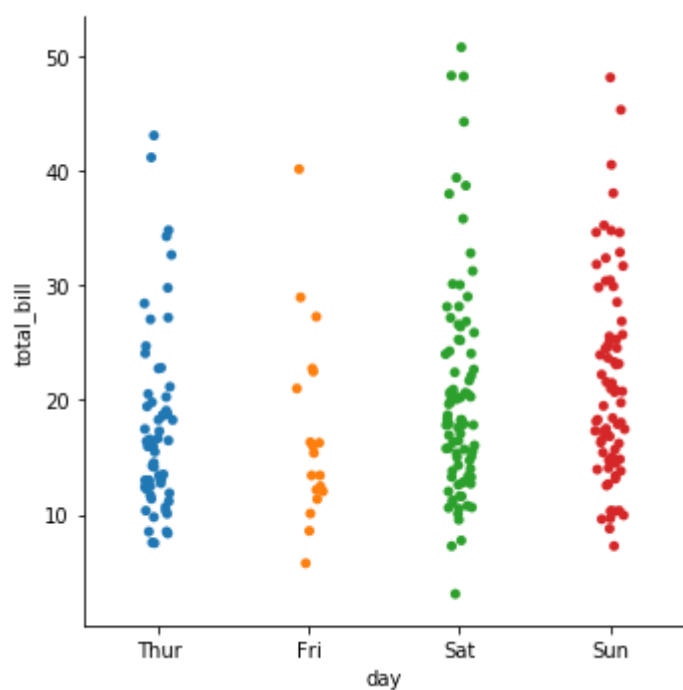
2.1.2. 두 그래프: 분석하고자 하는 데이터가 categorical 일 때 !

사용 방법:

sns.catplot(x축 데이터,y축 데이터,data)

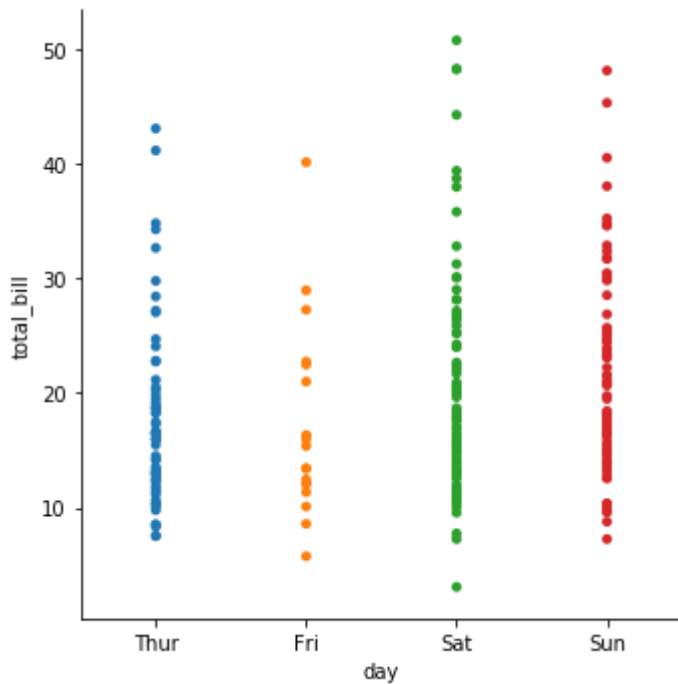
Out[16]:

<seaborn.axisgrid.FacetGrid at 0x1d51ea8aba8>



Out[18]:

<seaborn.axisgrid.FacetGrid at 0x1d51eba7630>

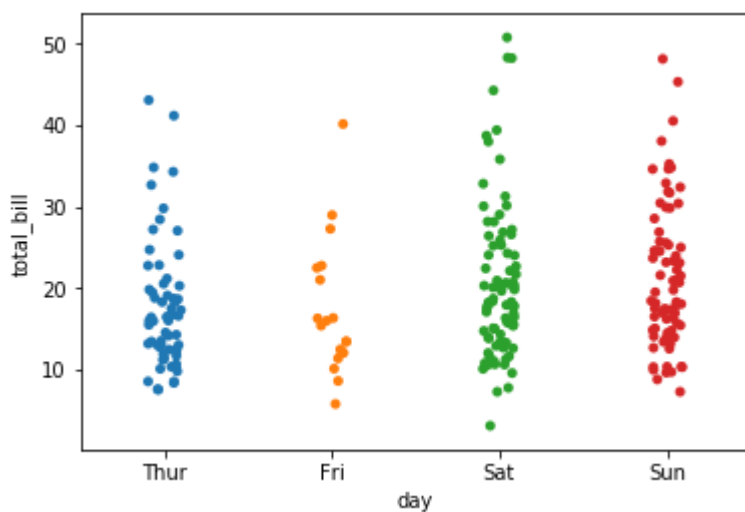


여기서 잠깐 ! 비슷한 형태의 plot들을 살펴보고 넘어가자.

- stripplot
stripplot은 마치 스캐터 플롯처럼 모든 데이터를 점으로 그려준다.
사용 방법:
`sns.stripplot(x축 데이터 , y축 데이터 , data)`

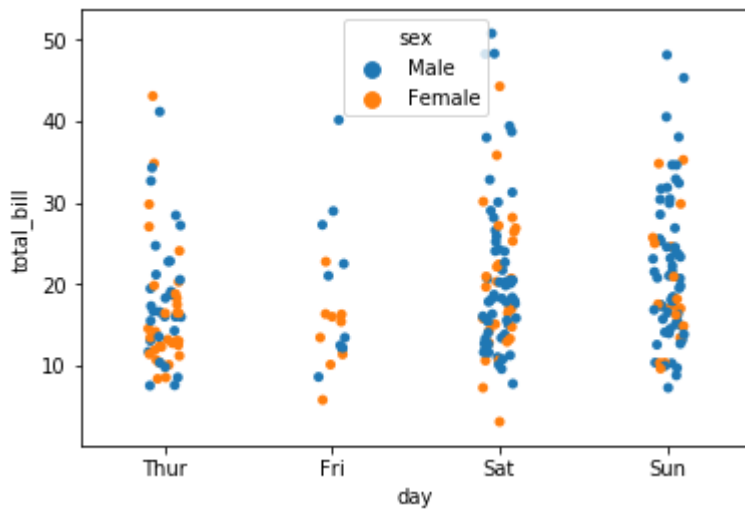
Out[20]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ea8ab70>



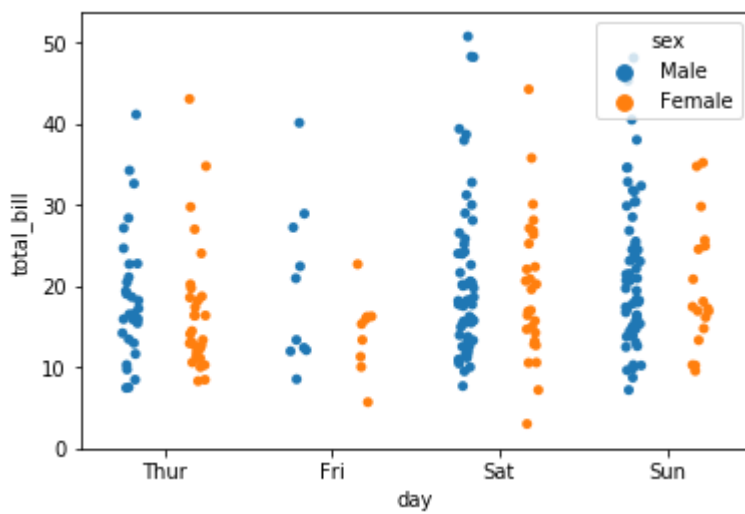
Out[21]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ec54ef0>



Out[22]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ecb6be0>

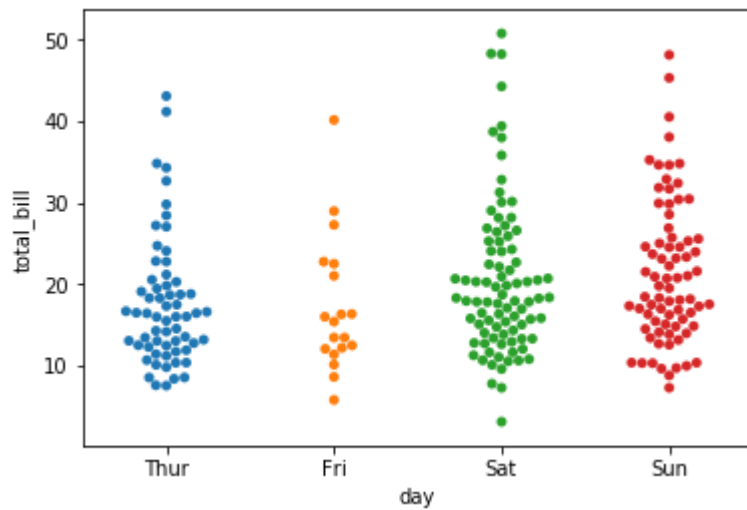


- `swarmplot`

`swarmplot`은 `stripplot`과 비슷하지만 데이터를 나타내는 점이 겹치지 않도록 옆으로 이동 사용 방법:
`sns.swarmplot(x축 데이터 , y축 데이터 , data)`

Out[23]:

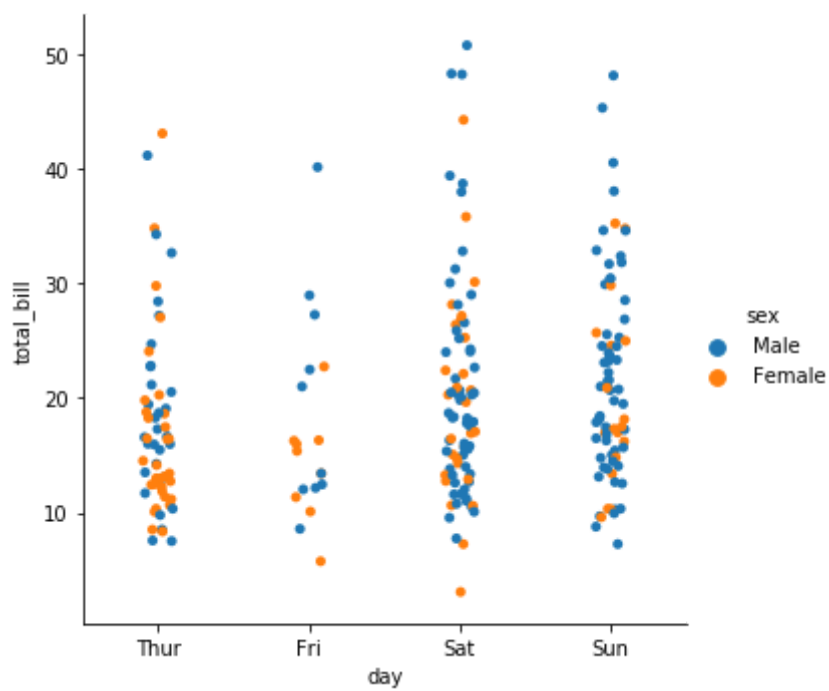
<matplotlib.axes._subplots.AxesSubplot at 0x1d51ed28668>



다시 catplot으로!

Out[25]:

<seaborn.axisgrid.FacetGrid at 0x1d51ebe6630>



2.2. Box Plot

박스는 실수 값 분포에서 1사분위수(Q1)와 3사분위수(Q3)를 뜻하고, 3사분위수와 1사분위의 차이($Q3 - Q1$)는 IQR이라고 한다.

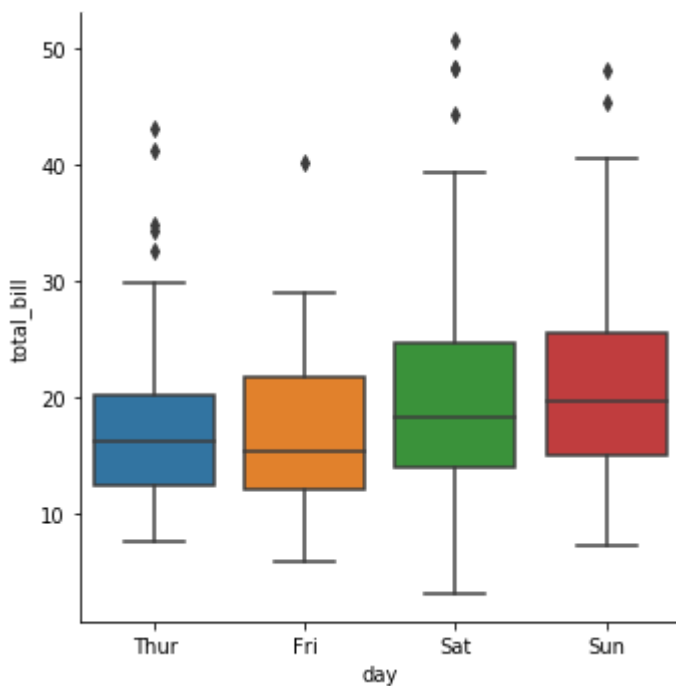
1사분위 수보다 $1.5 \times \text{IQR}$ 만큼 낮은 값과 3사분위 수보다 $1.5 \times \text{IQR}$ 만큼 높은 값의 구간을 기준으로 그 바깥의 점은 outlier(이상치)이다.

사용 방법:

```
sns.catplot( x축 데이터 , y축 데이터 , kind= box , data )
```

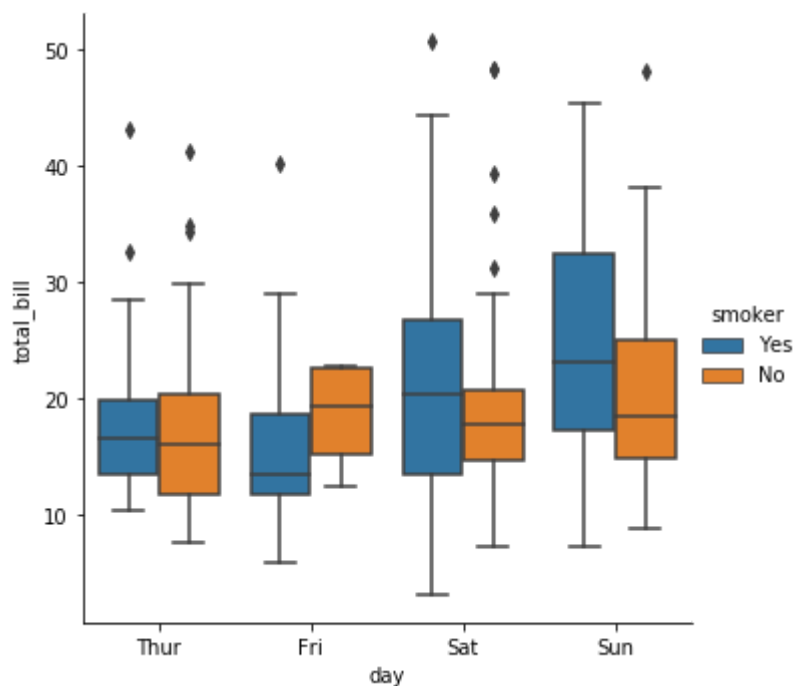
Out[27]:

<seaborn.axisgrid.FacetGrid at 0x1d51ed70630>



Out[28]:

<seaborn.axisgrid.FacetGrid at 0x1d51eec45c0>



2.3. Violin Plot

boxplot 은 분포의 간략한 특성만 보여주지만 violinplot 은 카테고리값에 따른 각 분포의 실제 데이터나 전체 형상을 보여준다는 장점이 있다.

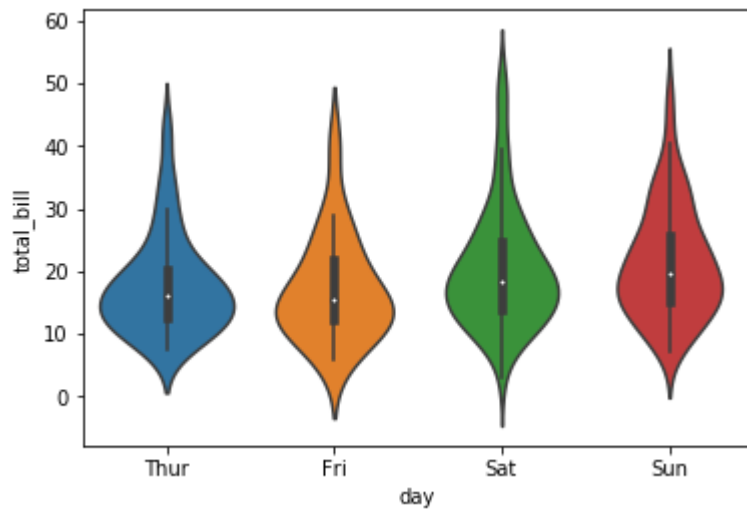
violinplot 은 세로 방향으로 커널 밀도 히스토그램을 그려주는데 왼쪽과 오른쪽이 대칭이 되도록 하여 바이올린처럼 보인다고 하여 붙은 이름이다.

사용 방법:

```
sns.violinplot(x축 데이터,y축 데이터,data)
```

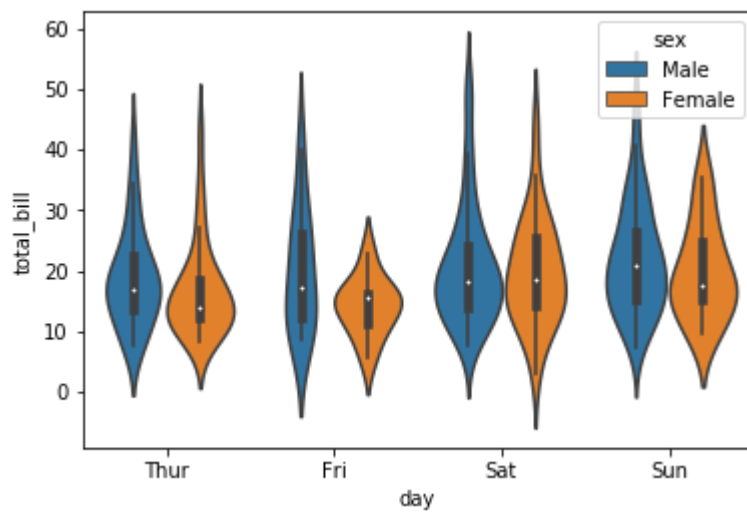
Out[32]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d520179b00>



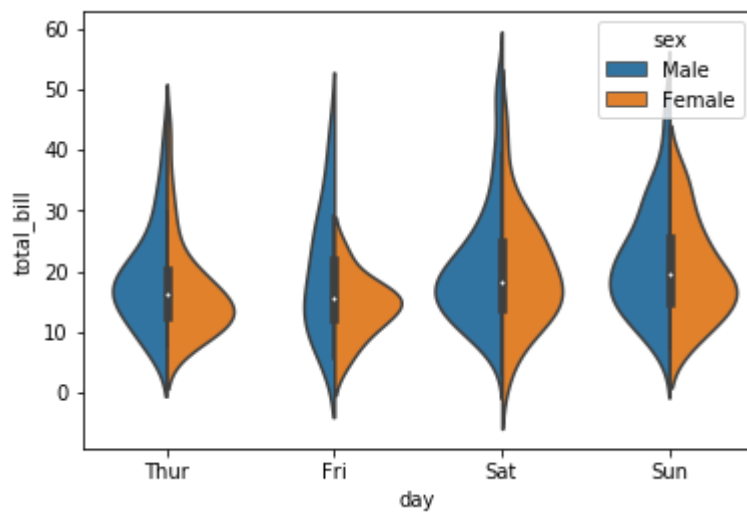
Out[33]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d5201ceb70>



Out[34]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d520294fd0>

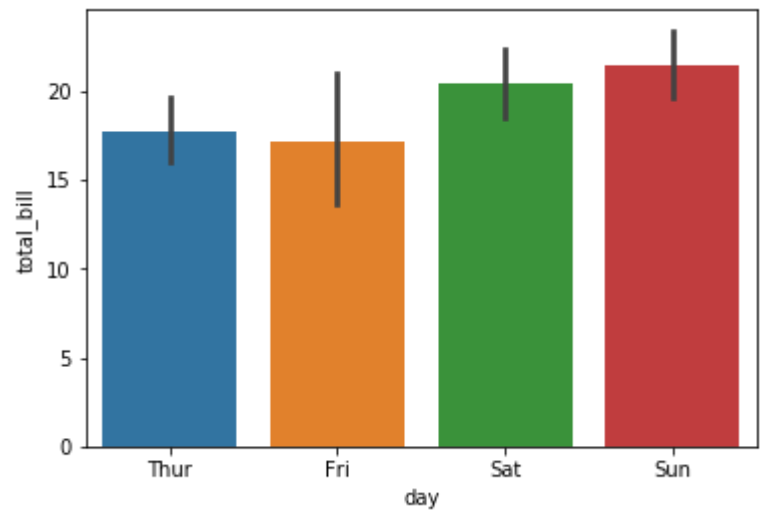


2.4. Bar Plot

카테고리 값에 따른 실수 값의 평균과 편차를 표시하는 기본적인 바 차트를 생성

사용 방법:
sns.barplot(x축 데이터 , y축 데이터 , data)

Out[35]:
<matplotlib.axes._subplots.AxesSubplot at 0x1d5203222e8>



타이타닉 데이터에 적용해보기

- 1. 타이타닉 데이터 확인

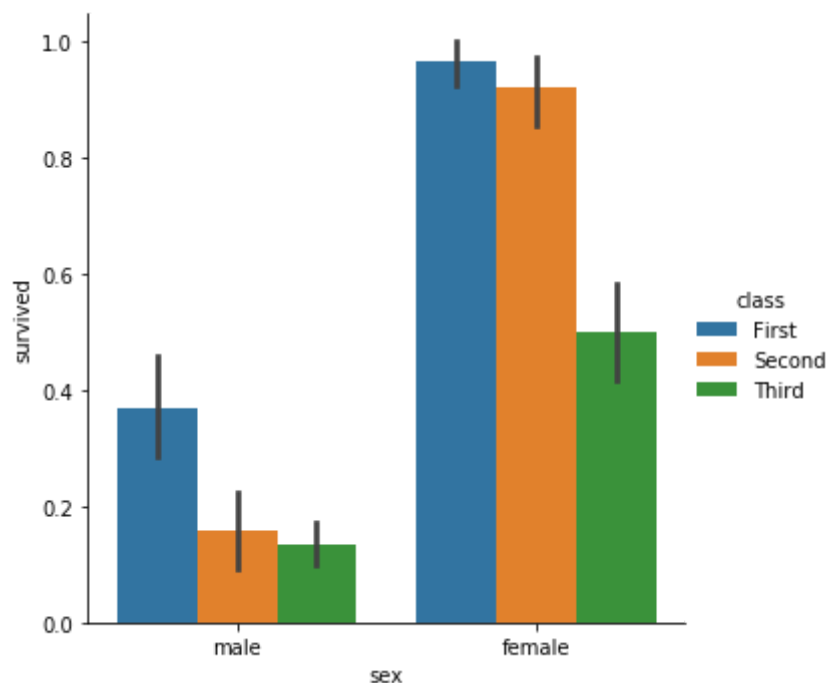
Out[36]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_ma
0	0	3	male	22.0	1	0	7.2500	S	Third	man	Tr
1	1	1	female	38.0	1	0	71.2833	C	First	woman	Fal
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	Fal
3	1	1	female	35.0	1	0	53.1000	S	First	woman	Fal
4	0	3	male	35.0	0	0	8.0500	S	Third	man	Tr

- 1. bar plot 그리기 (catplot을 이용해서 그려보자!)

Out[37]:

<seaborn.axisgrid.FacetGrid at 0x1d5202ff550>



2.5. Rug Plot

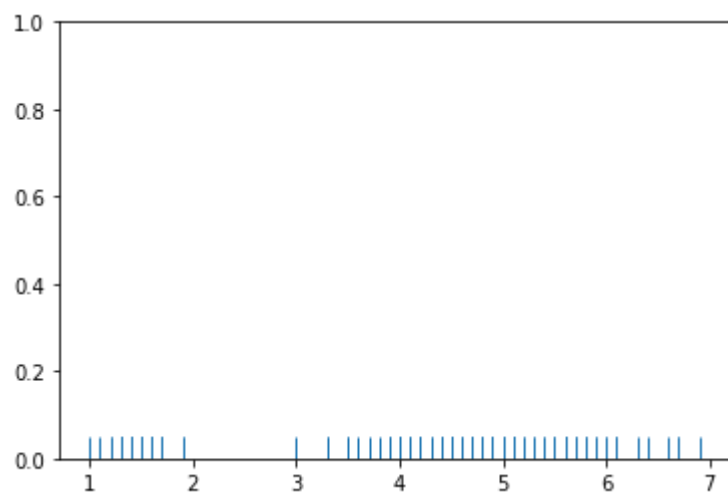
러그(rug) 플롯은 데이터 위치를 x축 위에 작은 선분(rug)으로 나타내어 실제 데이터들의 위치를 보여줌

사용 방법:

```
sns.rugplot( data )
```

Out[39]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d5203fd438>



Out[40]:

```
array([1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4,
       1.1, 1.2, 1.5, 1.3, 1.4, 1.7, 1.5, 1.7, 1.5, 1. , 1.7, 1.9, 1.6,
       1.6, 1.5, 1.4, 1.6, 1.6, 1.5, 1.5, 1.4, 1.5, 1.2, 1.3, 1.4, 1.3,
       1.5, 1.3, 1.3, 1.3, 1.6, 1.9, 1.4, 1.6, 1.4, 1.5, 1.4, 4.7, 4.5,
       4.9, 4. , 4.6, 4.5, 4.7, 3.3, 4.6, 3.9, 3.5, 4.2, 4. , 4.7, 3.6,
       4.4, 4.5, 4.1, 4.5, 3.9, 4.8, 4. , 4.9, 4.7, 4.3, 4.4, 4.8, 5. ,
       4.5, 3.5, 3.8, 3.7, 3.9, 5.1, 4.5, 4.5, 4.7, 4.4, 4.1, 4. , 4.4,
       4.6, 4. , 3.3, 4.2, 4.2, 4.2, 4.3, 3. , 4.1, 6. , 5.1, 5.9, 5.6,
       5.8, 6.6, 4.5, 6.3, 5.8, 6.1, 5.1, 5.3, 5.5, 5. , 5.1, 5.3, 5.5,
       6.7, 6.9, 5. , 5.7, 4.9, 6.7, 4.9, 5.7, 6. , 4.8, 4.9, 5.6, 5.8,
       6.1, 6.4, 5.6, 5.1, 5.6, 6.1, 5.6, 5.5, 4.8, 5.4, 5.6, 5.1, 5.1,
       5.9, 5.7, 5.2, 5. , 5.2, 5.4, 5.1])
```

2.6. KDe Plot

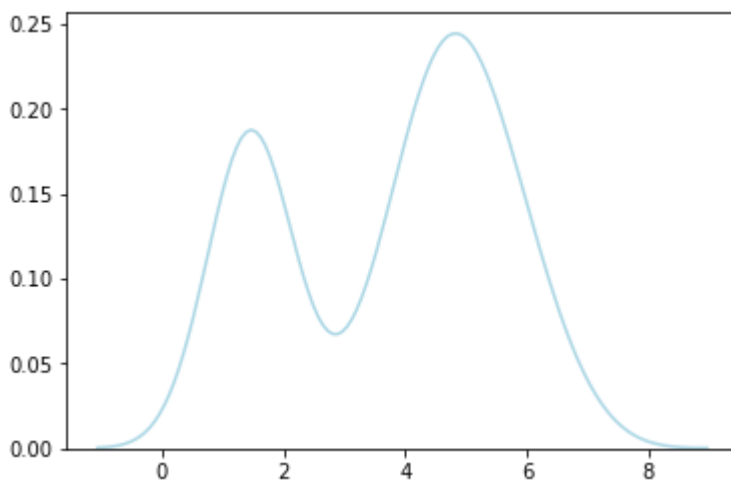
히스토그램보다 부드러운 형태의 분포 곡선을 보여주는 방법

사용 방법:

```
sns.kdeplot( data )
```

Out[41]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d5203d4ac8>



- dist plot

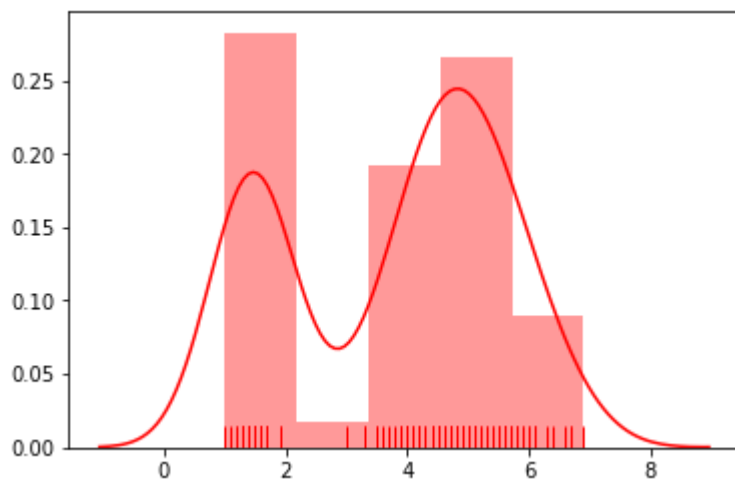
로그 + 커널 밀도 + 히스토그램 표시 기능

사용 방법 :

```
sns.distplot ( data , kde =True or False, rug =True or False)
```

Out[45]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ef5a588>



2.7. Count Plot

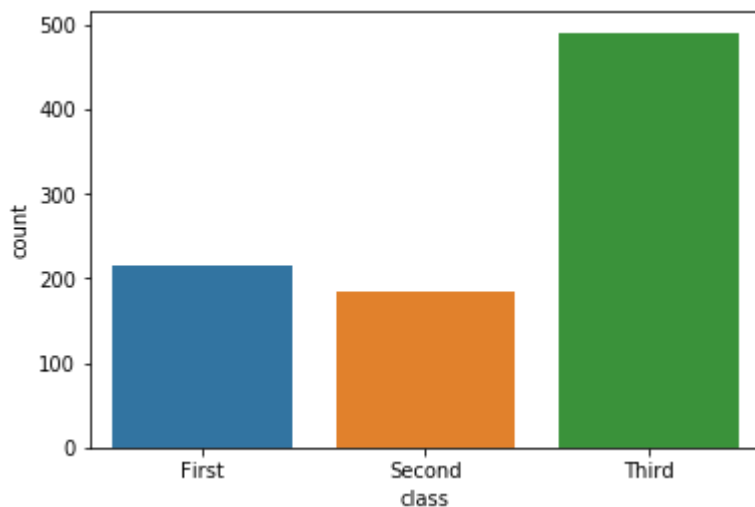
각 카테고리 값별로 데이터가 얼마나 있는지 표시 가능 (빈도 수를 그래프에 표시)

사용 방법:

```
sns.countplot(x=column_name, data)
```

Out[46]:

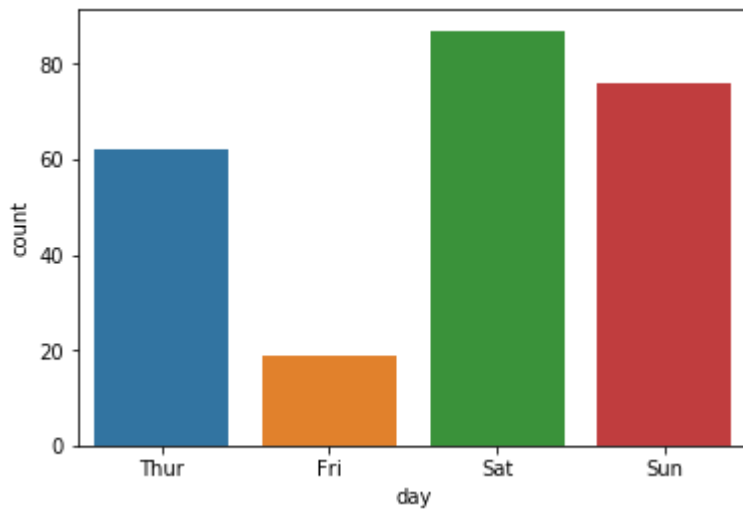
<matplotlib.axes._subplots.AxesSubplot at 0x1d52052f080>



class별 승객 수를 나타낼 수 있다.

Out[47]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d51ee94ac8>



요일별 팁을 준 횟수를 알 수 있다.

2.8. Joint Plot

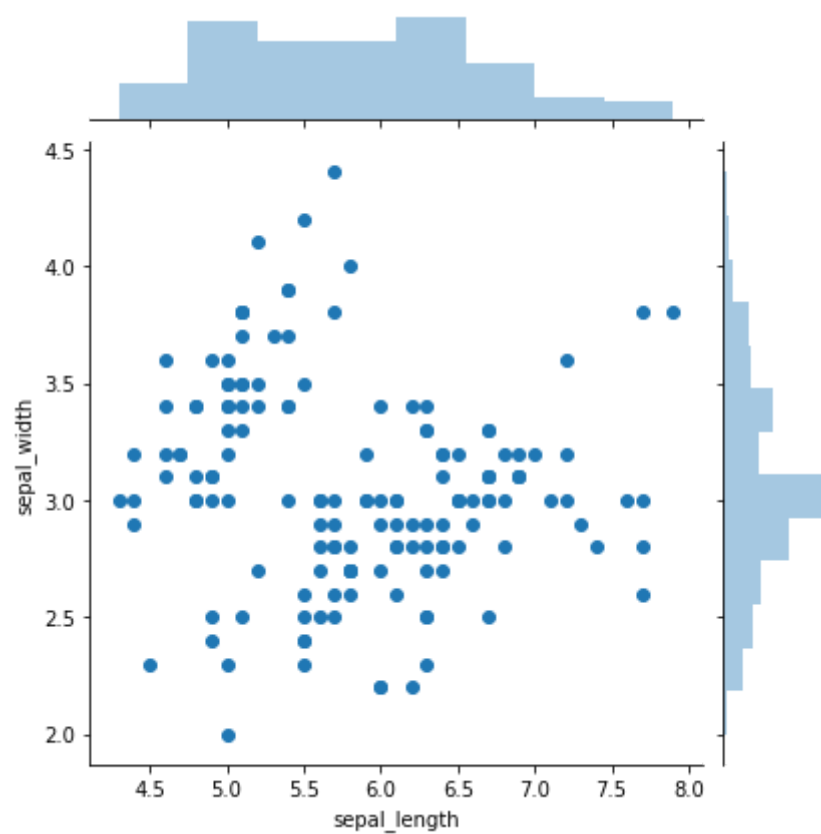
스캐터 플롯뿐 아니라 차트의 가장자리(margin)에 각 변수의 히스토그램도 그린다.

사용 방법:

`sns.jointplot(x, y, data, kind)`

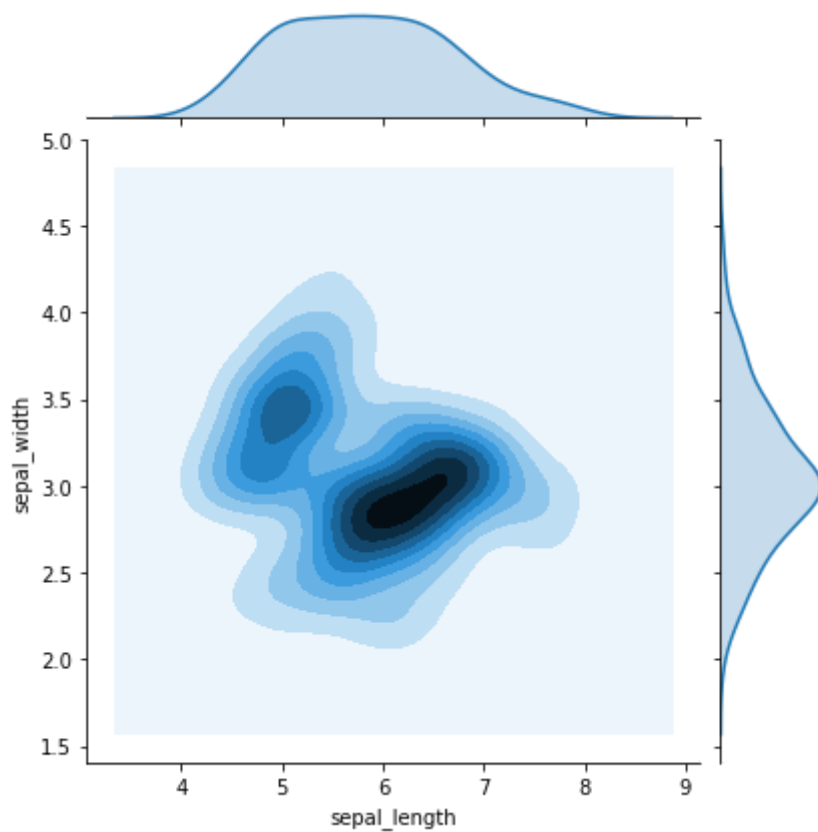
Out[49]:

<seaborn.axisgrid.JointGrid at 0x1d5205ccda0>



Out[50]:

<seaborn.axisgrid.JointGrid at 0x1d5206d0390>



2.9. Pair Plot

3차원 이상의 데이터라면 seaborn 패키지의 pairplot 명령을 사용하자.

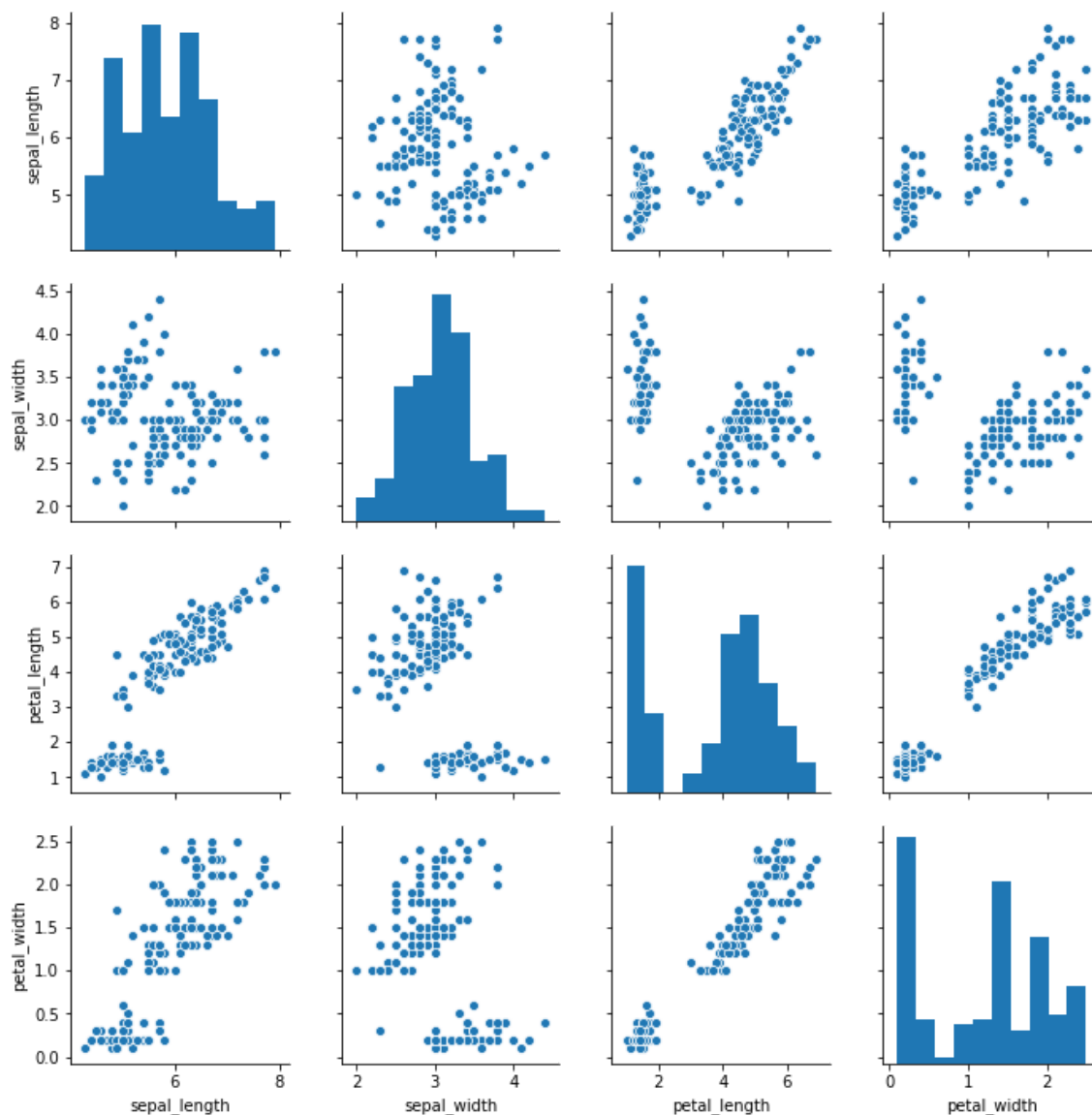
pairplot은 데이터프레임을 인수로 받아 그리드 형태로 각 데이터 열의 조합에 대해 스캐터 플롯을 그려준다. 같은 데이터가 만나는 대각선 영역에는 해당 데이터의 히스토그램을 생성한다.

사용 방법:

```
sns.pairplot( data )
```

Out[52]:

<seaborn.axisgrid.PairGrid at 0x1d520801a90>



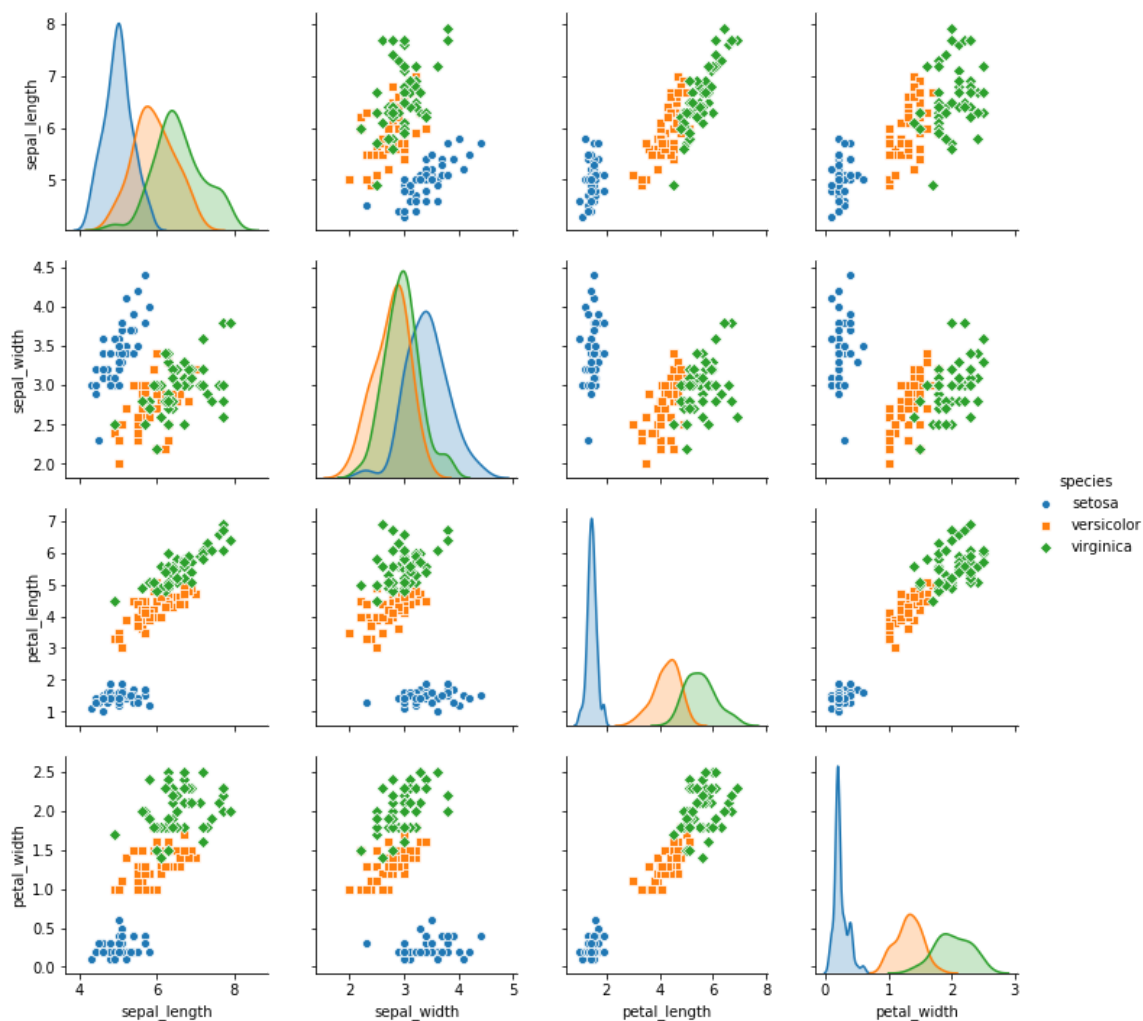
Species에 따른 구분이 안 되어 있네? 어떻게 하면 구분할 수 있을까?

hue 파라미터 이용하면 됨!

seaborn은 matplotlib의 확장 패키지이기 때문에 marker의 종류가 같음

Out[53]:

<seaborn.axisgrid.PairGrid at 0x1d521fa4be0>



2.10. Reg Plot & LM Plot

변수들 간의 선형 관계를 확인할 때에 사용한다.

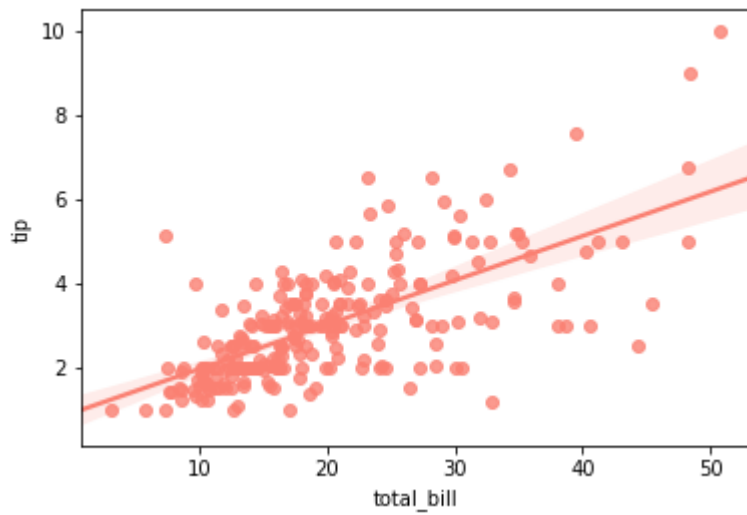
사용 방법:

1. sns.regplot(x축 데이터 , y축 데이터 , data)
2. sns.lmplot(x축 데이터 , y축 데이터 , data)

실제로 regplot보다 lmplot이 더 많이 쓰이는데 그 이유는 lmplot에서만 hue 파라미터가 적용되기 때문!

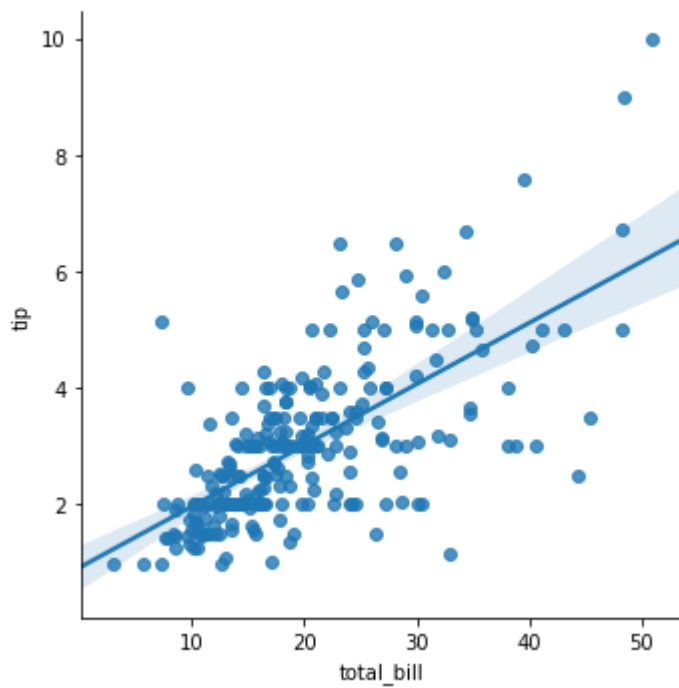
Out[55]:

<matplotlib.axes._subplots.AxesSubplot at 0x1d5229ebeb8>



Out[56]:

<seaborn.axisgrid.FacetGrid at 0x1d522d25e10>



Out[57]:

<seaborn.axisgrid.FacetGrid at 0x1d522fb36d8>

