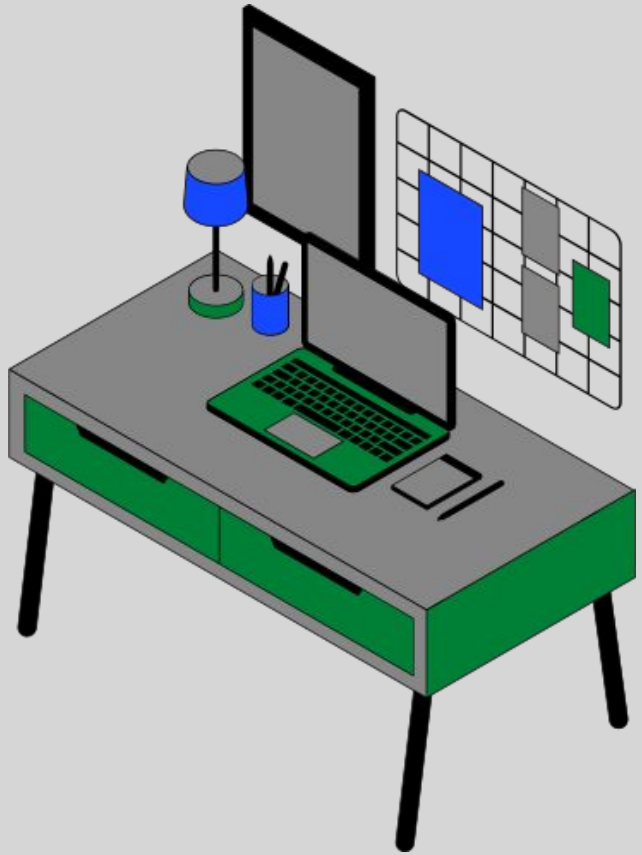
데이터 시각화 교과서

Chapter 9. 여러 분포 상태의 결합 시각화



여러개의 분포 상태를 다룰 때

-반응 변수: 우리가 도표를 통해 분포 상태를 보여주려하는 변수

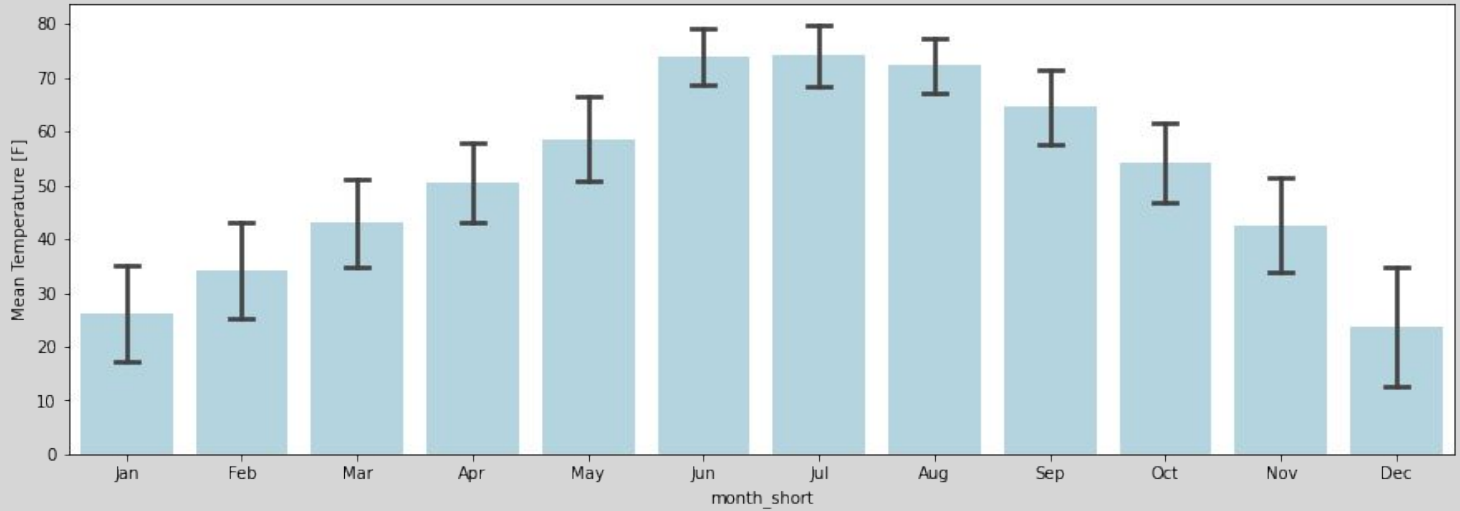
-그룹화 변수: 반응 변수의 분포가 뚜렷이 구분되는 데이터의 부분집합을 정의하는 변수

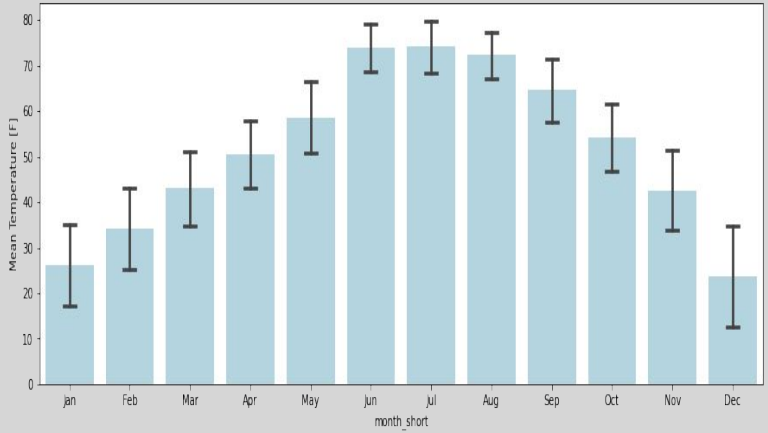
Ex. 월별 기온 분포
기온: 반응변수
월: 그룹화변수

가로축에 기준을 둔 분포 상태의 시각화

여러 분포 상태를 도표 하나에 나타내는 가장 간단한 방법

-평균/중간값을 표시하고 그 주변에 오차막대 그림





2016년 링컨시의 월별 기온 분포

위 도표의 문제점

1. 데이터에 대한 정보가 상당수 가려짐
2. 점이 무엇을 의미하는지 명시X
3. 오차 막대가 무엇을 뜻하는지 설명X
4. 대칭오차 막대는 비대칭 데이터를 잘못 전달

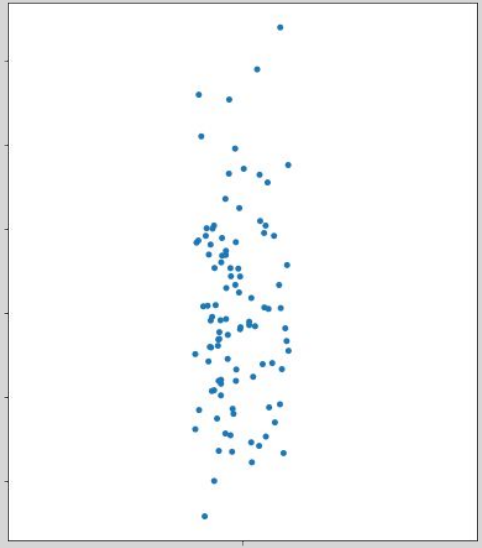
보완 방식

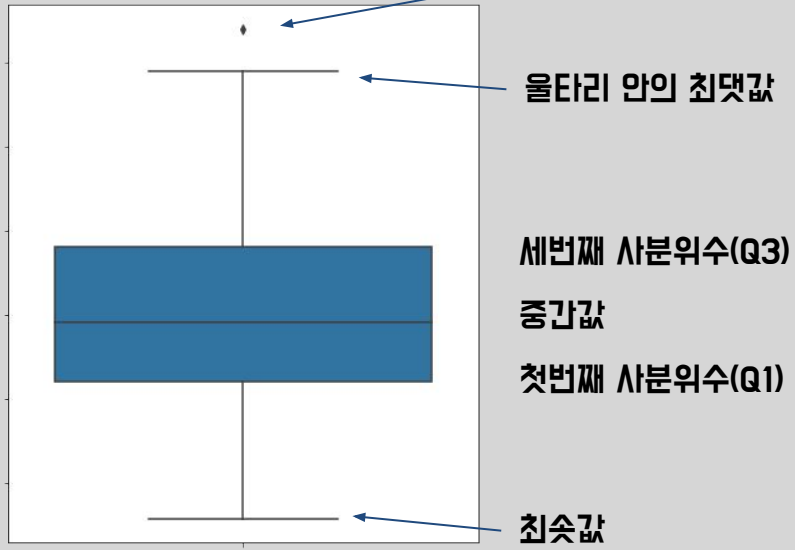
-Box Plot 활용

Box Plot

-데이터를 사분위수로 나누고 시각화

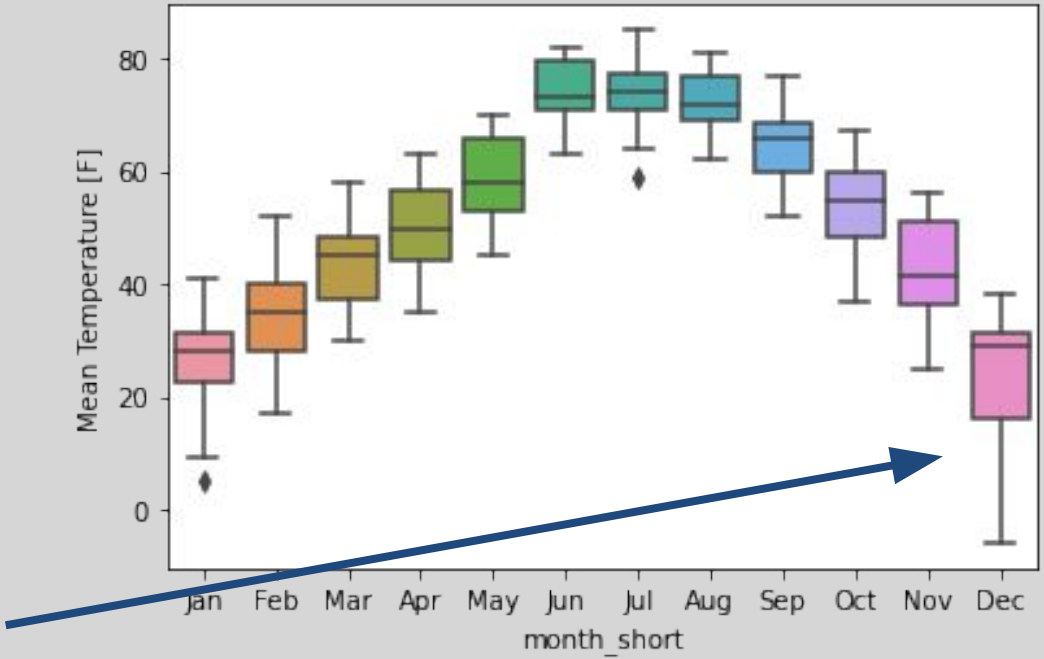
-형태가 단순, 정보 전달에 효과적





보완 방식

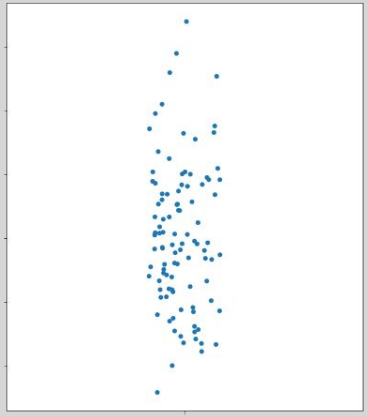
-Box Plot 활용

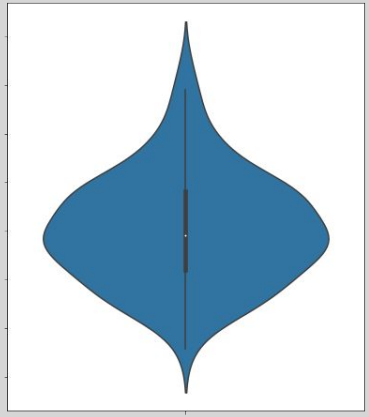


12월 기온 비대칭성이
강하다는 사실을 알 수 있음

보완 방식

-Violin Plot 활용





Violin Plot

특징

-밀도 추정 도표를 90도 회전, 반전시킨 것

-바이올린의 너비: 그 y 값에서의 점밀도

장점

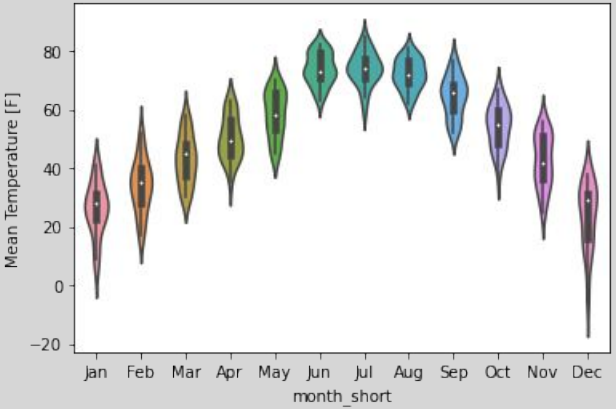
-현대에는 box plot보다 많이 사용

-어떤 경우든 box plot 대체 가능

-데이터의 미묘한 차이를 보여줌

-데이터 분포 표현 가능(쌍봉 데이터 표현 가능)

바이올린 도표



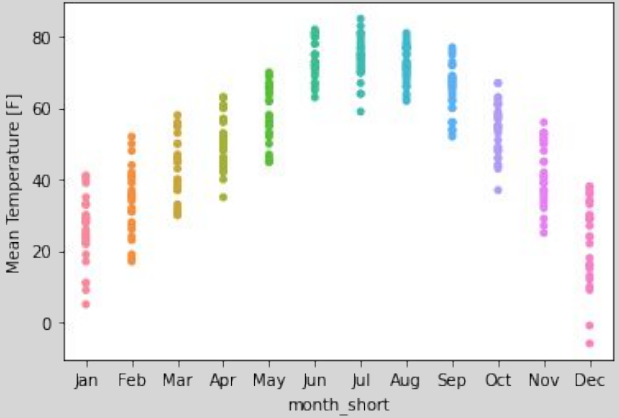
단점

-데이터가 없는 영역이 데이터가

있는듯한 형태를 띠거나, 듬성듬성한

데이터가 빽빽한 것처럼 표현될수도

스트립 차트

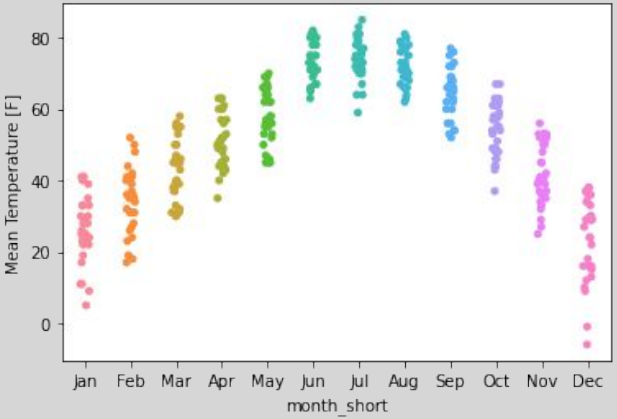


단점

-점을 너무 많이 겹쳐 찍으면 데이터의

분포를 알아보기 힘듦

스트립 차트

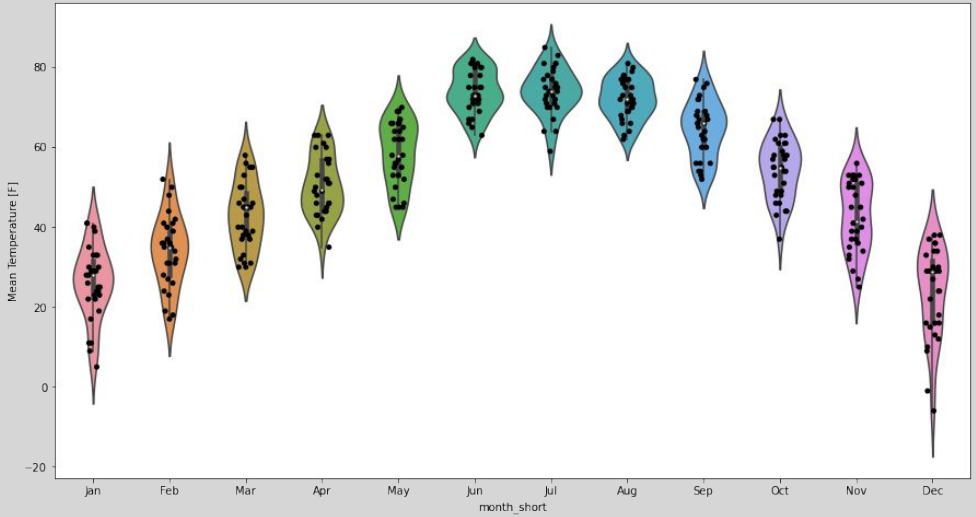


지터링 기법

각 데이터의 x축 값에 임의의 노이즈를

더해 x축 상에 데이터를 흩뿌림

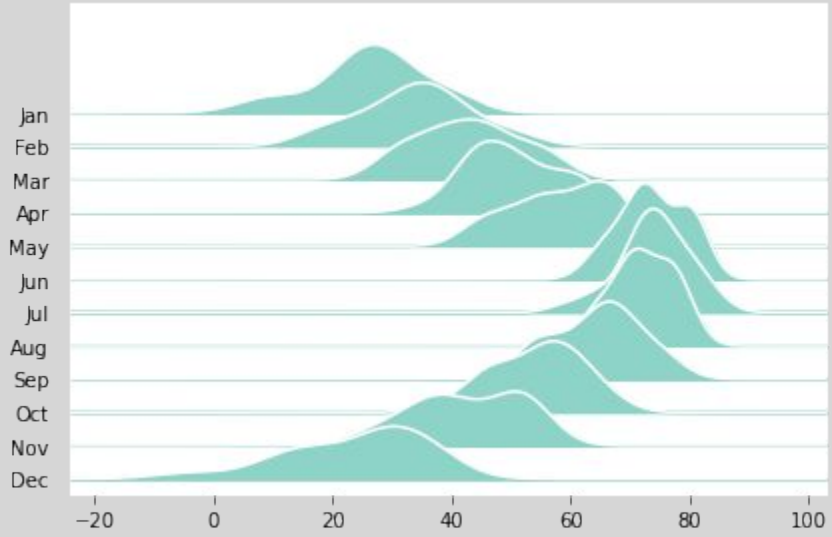
시나 도표



바이올린 도표와 지터링한 스트립 차트의 장점만 모은 결과물

⇒ 두 방식을 혼합해 점들의 각 위치와 분포 상태를 동시에 보여줌

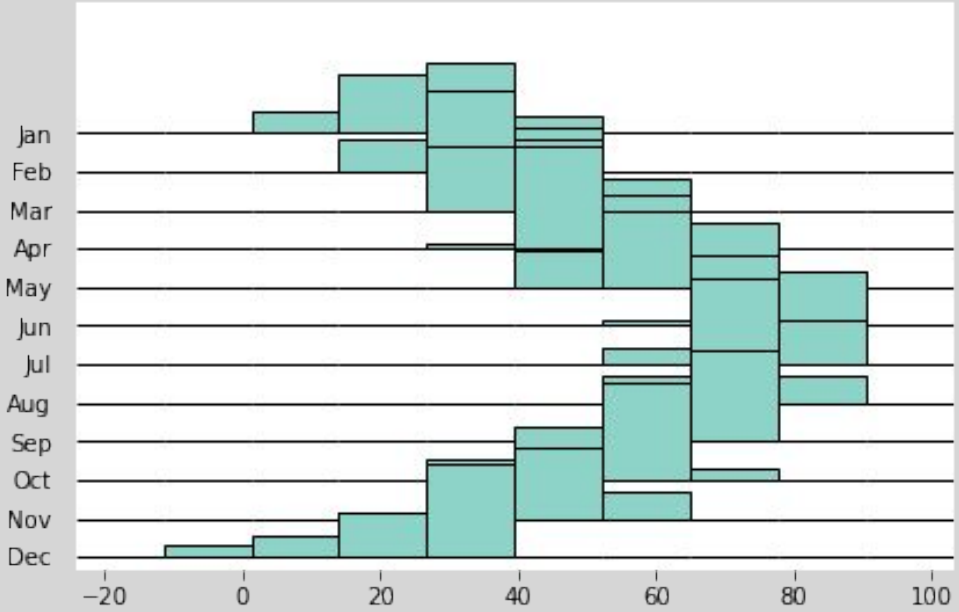
세로축에 기준을 둔 분포 상태의 시각화



융기선 도표

-히스토그램과 밀도 도표를 사용해 분포를 시각화
-시간의 흐름에 따른 분포 추세를 보여줄 때 유용
-바이올린 도표와 비슷(각 그룹 전반의 밀도 형태와 상대적인 높이를 쉽게 비교하기 위한 시각화 방식)

융기선 도표 with 히스토그램

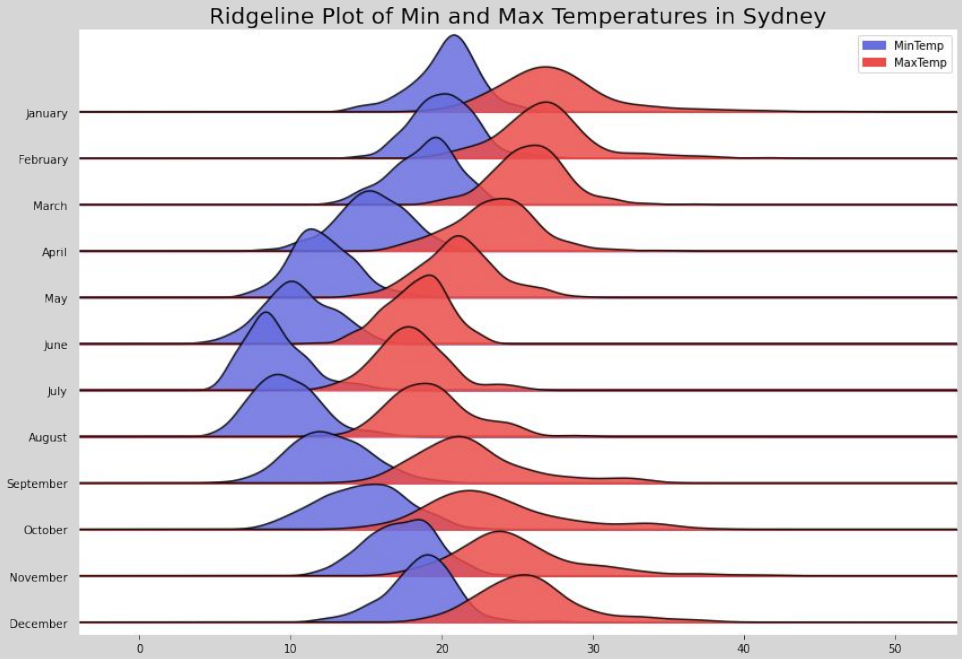


-여러 히스토그램의 막대가 한줄로 이어져 구분하기 힘듦

-누적/중첩의 문제가 생김

=> 융기선 도표에 히스토그램을 사용하는 것 권장하지 않음

융기선 도표



-시간의 흐름에 따른 두 개의 추세를 비교하기에도 좋음

-시드니의 겨울인 7,8월에 일교차가 크다는 것을 알 수 있음

#my code

```
from matplotlib import pyplot as plt
import seaborn as sns
import random
import pandas as pd
from pandas import Series, DataFrame
import numpy as np
lincoln_df=pd.read_csv('lincoln_df.csv')
fig = plt.figure(figsize=(15,5))
ax = sns.barplot(data=lincoln_df,x = 'month_short', y = 'Mean
Temperature [F]', estimator=np.mean, ci="sd", capsize=.2,
color='lightblue')
```

#box plot

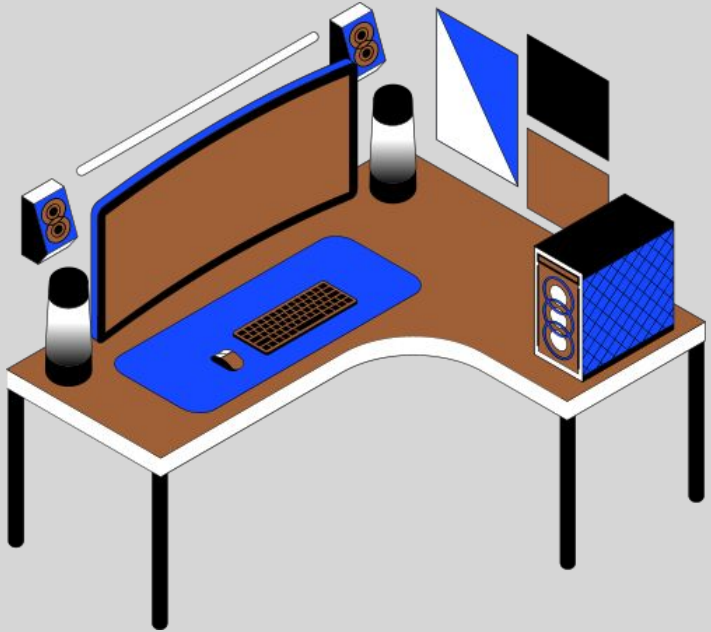
```
df = pd.read_csv('df (1).csv')
fig = plt.figure(figsize=(15,8))
for i in range(1,3):
 globals()['area{}'.format(i)]=fig.add_subplot(1,2,i)
a1 = sns.stripplot(data=df, y = "y",ax=area1)
a1.set(xticklabels=[],yticklabels=[],xlabel=None,ylabel=None)
a2 = sns.boxplot(y = "y", data = df,ax=area2)
a2.set(xticklabels=[],yticklabels=[],xlabel=None,ylabel=None)
plt.show()
```

#violin plot

```
fig = plt.figure(figsize=(15,8))
for i in range(1,3):
 globals()['area{}'.format(i)]=fig.add_subplot(1,2,i)
a1 = sns.stripplot(data=df, y = "y",ax=area1)
a1.set(xticklabels=[],yticklabels=[],xlabel=None,ylabel=None)
a2 = sns.violinplot(data=df, y = "y",ax=area2)
a2.set(xticklabels=[],yticklabels=[],xlabel=None,ylabel=None)
plt.show()
x = sns.violinplot(x = 'month_short', y = 'Mean Temperature
[F]', data = lincoln_df)
plt.show()
x = sns.stripplot(x = 'month_short', y = 'Mean Temperature
[F]', data = lincoln_df,jitter=False)
x = sns.stripplot(x = 'month_short', y = 'Mean Temperature
[F]', data = lincoln_df,jitter=True)
fig = plt.figure(figsize=(15,8))
sns.stripplot(x = 'month_short', y = 'Mean Temperature [F]',
data = lincoln_df,jitter=True,color='black')
sns.violinplot(x = 'month_short', y = 'Mean Temperature [F]',
data = lincoln_df)
plt.show()
```

#ridgeline plot

```
from seaborn import palettes
sns.set_palette('Set3')
grouped = lincoln_df.groupby("month_short", sort=False)
joypy.joyplot(grouped, column="Mean Temperature [F]",linecolor='w')
plt.show()
from seaborn import palettes
sns.set_palette('Set3')
grouped = lincoln_df.groupby("month_short", sort=False)
joypy.joyplot(grouped, column="Mean Temperature [F]",hist=True)
plt.show()
ax, fig = joypy.joyplot(
   data=sydney[['MinTemp', 'MaxTemp', 'Month']],
   by='Month',
   column=['MinTemp', 'MaxTemp'],
   color=['#686de0', '#eb4d4b'],
   legend=True,
   alpha=0.85,
   figsize=(12, 8)
)
plt.title('Ridgeline Plot of Min and Max Temperatures in Sydney', fontsize=20)
plt.show()
```



Do you have any questions?

Thank you!