
python

5. 데이터 시각화

- 데이터 시각화
  - 변수 값의 분포나 변수 사이의 관계를 확인, 모델링을 위한 가설을 도출하는 데 도움
  - matplotlib, pandas, ggplot, seaborn 등의 패키지 제공

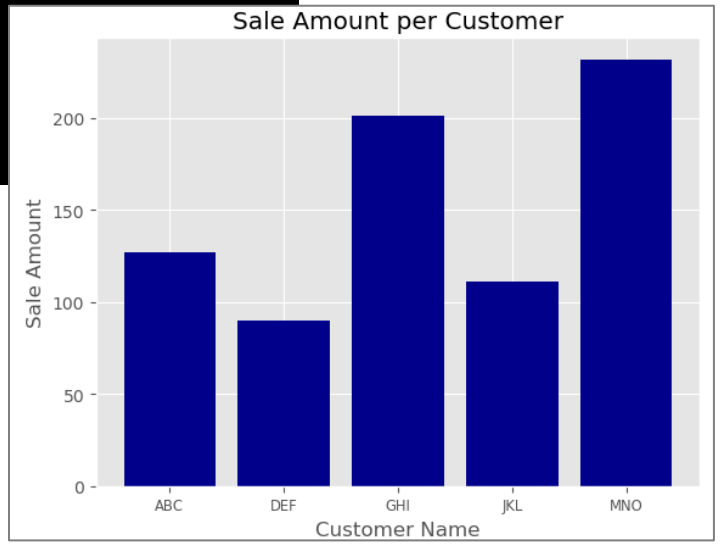
5-1. matplotlib

- 고품질의 그래프 작성
- 막대 그래프, 상자그림, 선 그래프, 산점도, 히스토그램 등의 통계 그래프 생성
- basemap, cartopy, mplot3d 등도 지원

1) 그림 생성 (figure) - fig = plt.figure()
2) 하위 그래프 추가 (subplot) - ax1 = fig.add_subplot(1, 1, 1)
3) X, Y축 레이블, 눈금 작성 - plt.xlabel('Customer Name')
4) 그래프 작성 - plt.xticks(customers_index, customers, rotation=0, fontsize='small')
5) 이미지로 저장 or 화면에 표시 - plt.savefig() or plt.show()

5-1. matplotlib – 막대 그래프


```
customers = ['ABC', 'DEF', 'GHI', 'JKL', 'MNO']
customers_index = range(len(customers))
sale_amounts = [127, 90, 201, 111, 232]

fig = plt.figure()
ax1 = fig.add_subplot(1,1,1)
ax1.bar(customers_index, sale_amounts, align='center', color='darkblue')
ax1.xaxis.set_ticks_position('bottom')
ax1.yaxis.set_ticks_position('left')
plt.xticks(customers_index, customers, rotation=0, fontsize='small')

plt.xlabel('Customer Name')
plt.ylabel('Sale Amount')
plt.title('Sale Amount per Customer')
```

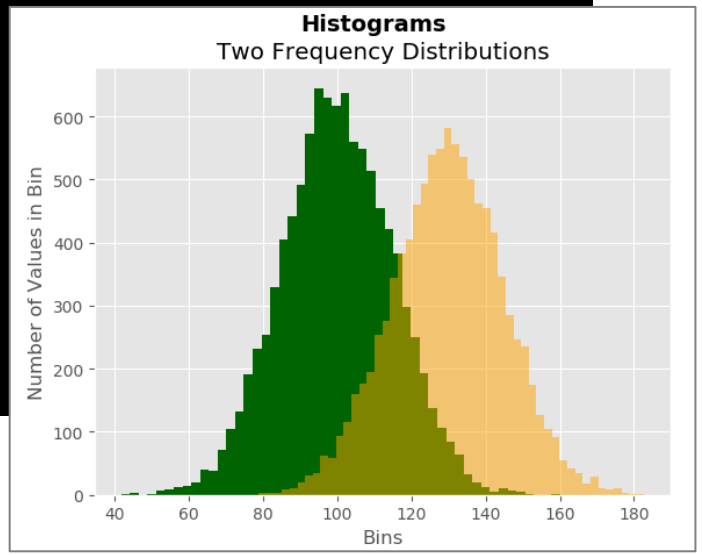


1matplotlib_basic_bar.py

5-1. matplotlib – 히스토그램

- 수치형 데이터 분포
- 빈도(도수), 빈도밀도(도수밀도), 확률, 확률밀도 등의 분포를 그릴 때 사용

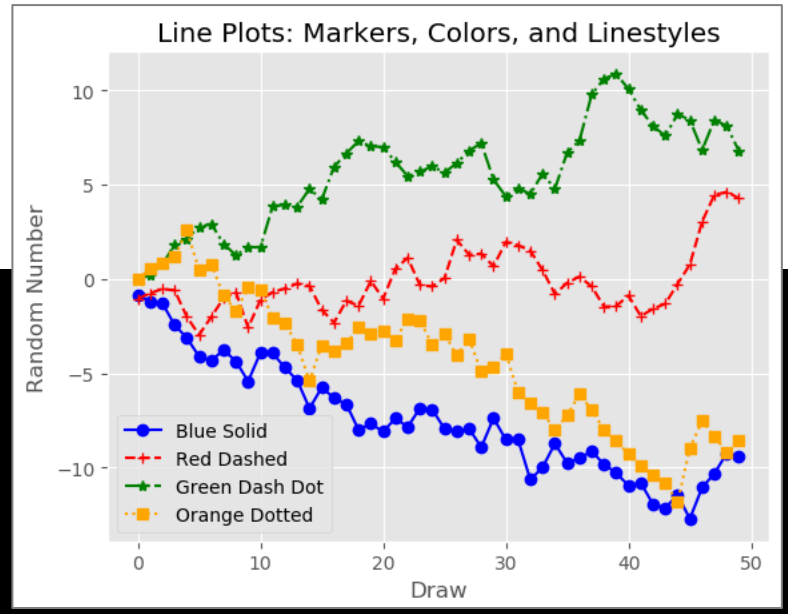
```
mu1, mu2, sigma = 100, 130, 15
x1 = mu1 + sigma*np.random.randn(10000)  ➔ 난수를 이용한 정규분포
x2 = mu2 + sigma*np.random.randn(10000)  ➔ 난수를 이용한 정규분포

fig = plt.figure()
ax1 = fig.add_subplot(1,1,1)     수치를 50개의 구간으로 표시
n, bins, patches = ax1.hist(x1, bins=50, normed=False, color='darkgreen')
n, bins, patches = ax1.hist(x2, bins=50, normed=False, color='orange', alpha=0.5)
ax1.xaxis.set_ticks_position('bottom')      빈도를 표시
ax1.yaxis.set_ticks_position('left')

plt.xlabel('Bins')
plt.ylabel('Number of Values in Bin')
fig.suptitle('Histograms', fontsize=14, fontweight='bold')
ax1.set_title('Two Frequency Distributions')
```



2matplotlib_basic_histogram.py

5-1. matplotlib – 선 그래프

- 수치의 변화를 선으로 표시
- 시간에 따른 데이터 변화 추세를 나타냄




```
plot_data1 = randn(50).cumsum() → 임의의 데이터 생성
plot_data2 = randn(50).cumsum()
plot_data3 = randn(50).cumsum()
plot_data4 = randn(50).cumsum()

fig = plt.figure()
ax1 = fig.add_subplot(1,1,1)
ax1.plot(plot_data1, marker=r'o', color=u'blue', linestyle='-', label='Blue Solid')
ax1.plot(plot_data2, marker=r'+', color=u'red', linestyle='--', label='Red Dashed')
ax1.plot(plot_data3, marker=r'*', color=u'green', linestyle='-.', label='Green Dash Dot')
ax1.plot(plot_data4, marker=r's', color=u'orange', linestyle=':', label='Orange Dotted')
ax1.xaxis.set_ticks_position('bottom')
ax1.yaxis.set_ticks_position('left')
```

```
plt.legend(loc='best')
```

3matplotlib_basic_line.py

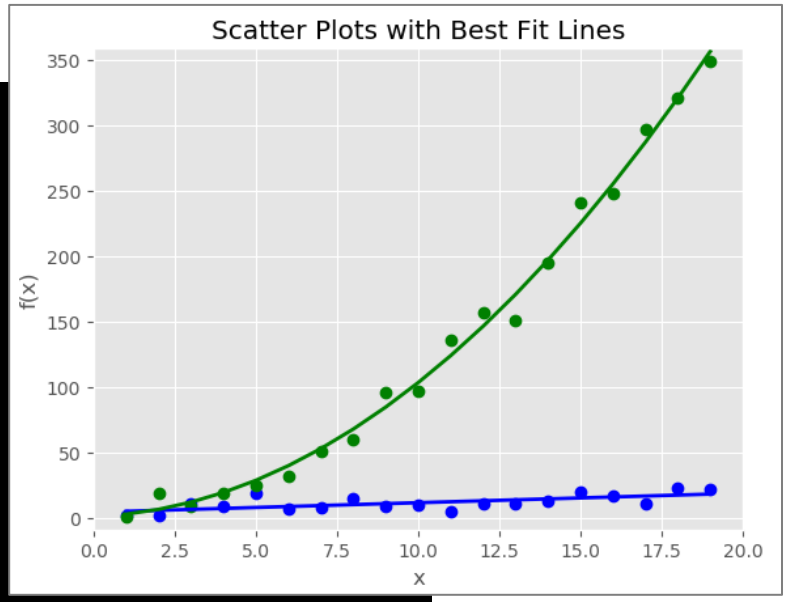
5-1. matplotlib – 산점도

- 두 변수 간의 관계를 표현

  ex) 키와 몸무게, 수요와 공급

- 두 변수가 양의 상관관계인지, 음의 상관관계인지 파악 가능
- Regression line으로 하나의 변수 값에 따른 다른 변수 값의 변화 추이를 예측 가능

  - 회귀선이란 제곱 오차의 최소값




```
x = np.arange(start=1., stop=20., step=1.)
y_linear = x + 5. * np.random.randn(19)
y_quadratic = x**2 + 10. * np.random.randn(19)

fn_linear = np.poly1d(np.polyfit(x, y_linear, deg=1))
fn_quadratic = np.poly1d(np.polyfit(x, y_quadratic, deg=2))

fig = plt.figure()
ax1 = fig.add_subplot(1,1,1)
ax1.plot(x, y_linear, 'bo', x, y_quadratic, 'go', \
            x, fn_linear(x), 'b-', x, fn_quadratic(x), 'g-', linewidth=2.)
ax1.xaxis.set_ticks_position('bottom')
ax1.yaxis.set_ticks_position('left')
```

선형 2차 다항식 생성

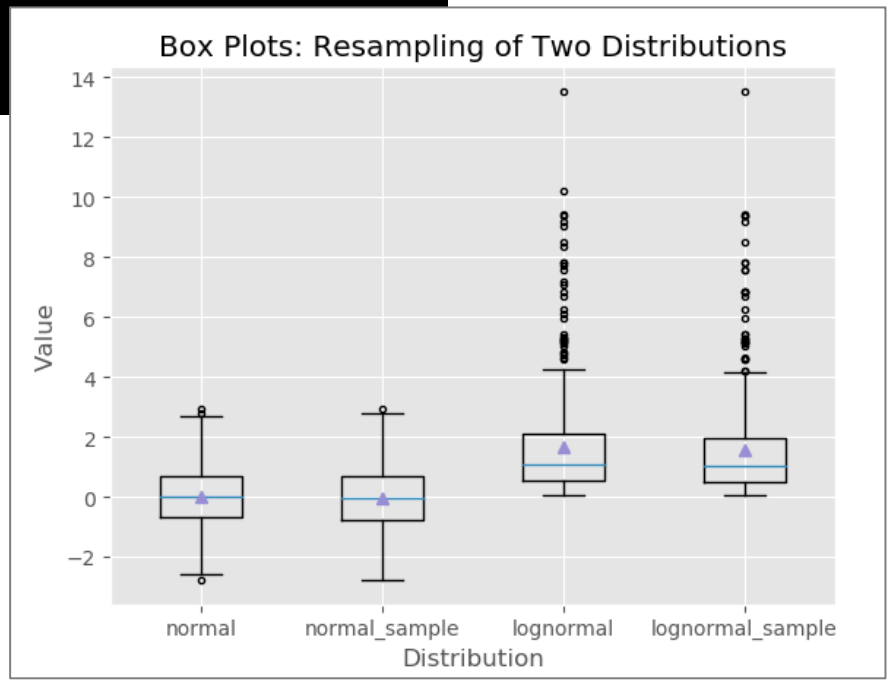
2개의 회귀선 생성

4matplotlib_basic_scatter.py

5-1. matplotlib – 박스(Candle)

- 5가지 통계량 표시
  - 최소값, 제1사분위수, 제2사분위수(중앙값), 제3사분위수, 최대값

```
box_labels = ['normal','normal_sample','lognormal','lognormal_sample']
ax1.boxplot(box_plot_data, notch=False, sym='.', vert=True, whis=1.5, \
                showmeans=True, labels=box_labels)
ax1.xaxis.set_ticks_position('bottom')
ax1.yaxis.set_ticks_position('left')
ax1.set_title('Box Plots: Resampling of Two Distributions')
ax1.set_xlabel('Distribution')
ax1.set_ylabel('Value')
```

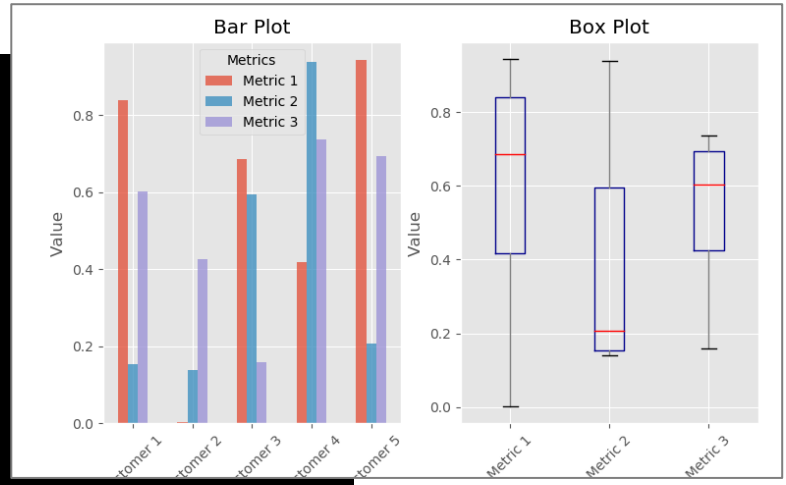


5matplotlib_basic_boxplot.py

5-2. Pandas

- 시리즈와 데이터프레임 자료형을 시각화 하기 위한 plot 함수 제공
- 기본은 선 그래프
  - 구간, 행렬, 밀도, 앤드루스, 평행좌표계, 시차, 자기상관, 부트스트랩 그래프 등 생성 가능

```
data_frame.plot(kind='bar', ax=ax1, alpha=0.75, title='Bar Plot')
plt.setp(ax1.get_xticklabels(), rotation=45, fontsize=10)
plt.setp(ax1.get_yticklabels(), rotation=0, fontsize=10)
ax1.set_xlabel('Customer')
ax1.set_ylabel('Value')
ax1.xaxis.set_ticks_position('bottom')
ax1.yaxis.set_ticks_position('left')

colors = dict(boxes='DarkBlue', whiskers='Gray', medians='Red', caps='Black')
data_frame.plot(kind='box', color=colors, sym='r.', ax=ax2, title='Box Plot')
plt.setp(ax2.get_xticklabels(), rotation=45, fontsize=10)
plt.setp(ax2.get_yticklabels(), rotation=0, fontsize=10)
```

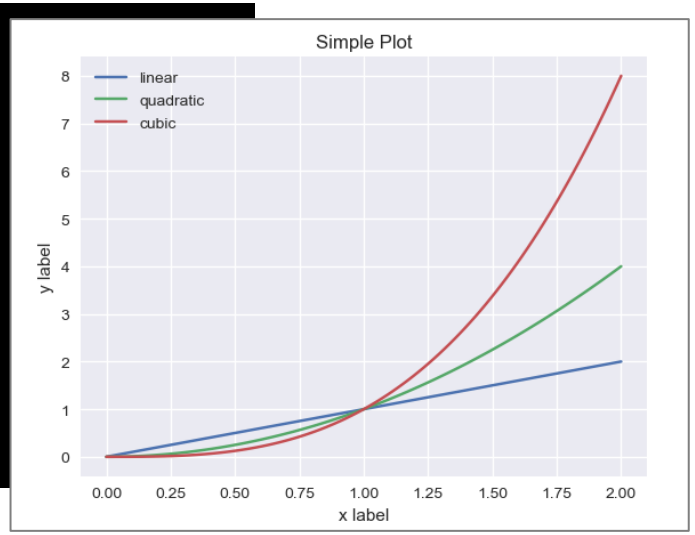


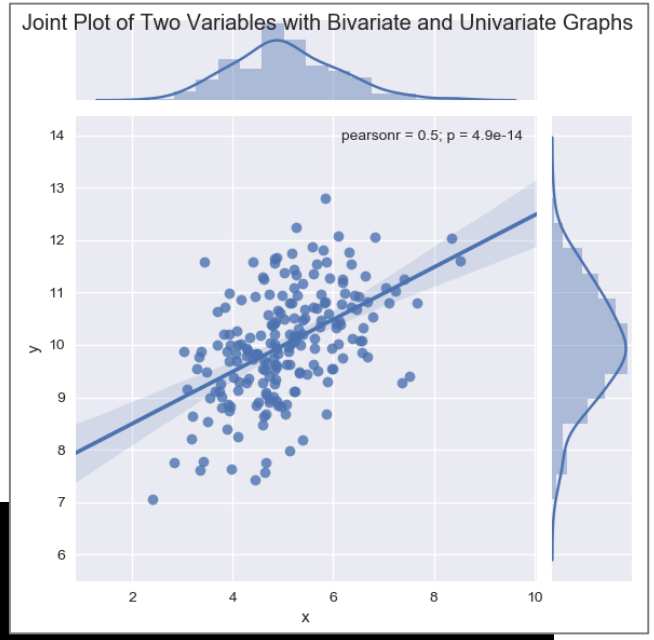
6pandas_plots.py

5-3. Seaborn

- 파이썬에서 통계 그래프와 그림을 단순하게 그려줌
- numpy, pandas 자료 구조 지원
- 히스토그램, 밀도 그래프, 막대 그래프, 상자그림, 산점도 등 통계 그래프 지원

```
x = np.linspace(0, 2, 100)
plt.plot(x, x, label='linear')
plt.plot(x, x**2, label='quadratic')
plt.plot(x, x**3, label='cubic')
plt.xlabel('x label')
plt.ylabel('y label')
plt.title("Simple Plot")
plt.legend(loc="best")
```





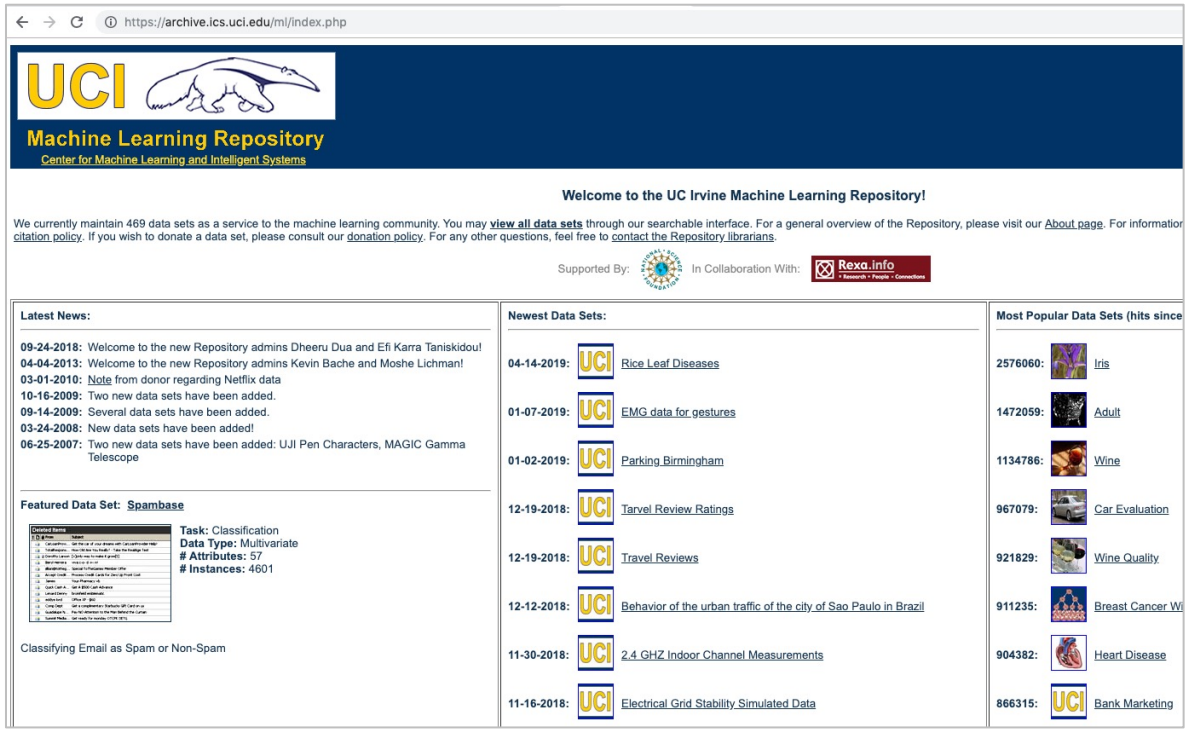

```
mean, cov = [5, 10], [(1, .5), (.5, 1)]
data = np.random.multivariate_normal(mean, cov, 200)
data_frame = pd.DataFrame(data, columns=["x", "y"])
sns.jointplot(x="x", y="y", data=data_frame, kind="reg").set_axis_labels("x", "y")
plt.suptitle("Joint Plot of Two Variables with Bivariate and Univariate Graphs")
```

7seaborn_plots.py

6. 통계 및 모델링

- 와인 품질 데이터셋
  - 레드 와인(1,599개)과 화이트 와인(4,898개)의 품질 평가 점수
  - 11개의 입력 데이터를 가지고 품질 평가 점수를 출력 (0~10)
  - https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv
  - https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv

- UCI Machine Learning Repository



6. 통계 및 모델링

- 와인 품질 데이터셋

In [1]:
```
import pandas as pd
df = pd.read_csv("winequality-white.csv", sep=";", encoding="utf-8")
df
```

Out[1]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	7.0	0.270	0.36	20.70	0.045	45.0	170.0	1.00100	3.00	0.45	8.800000	6
1	6.3	0.300	0.34	1.60	0.049	14.0	132.0	0.99400	3.30	0.49	9.500000	6
2	8.1	0.280	0.40	6.90	0.050	30.0	97.0	0.99510	3.26	0.44	10.100000	6
3	7.2	0.230	0.32	8.50	0.058	47.0	186.0	0.99560	3.19	0.40	9.900000	6
4	7.2	0.230	0.32	8.50	0.058	47.0	186.0	0.99560	3.19	0.40	9.900000	6
5	8.1	0.280	0.40	6.90	0.050	30.0	97.0	0.99510	3.26	0.44	10.100000	6
6	6.2	0.320	0.16	7.00	0.045	30.0	136.0	0.99490	3.18	0.47	9.600000	6
7	7.0	0.270	0.36	20.70	0.045	45.0	170.0	1.00100	3.00	0.45	8.800000	6

산성도
휘발성 산도
시트르산
잔류당
염화물
유리 이산화황
총 이산화황
밀도
pH
황산염
알코올
품질(0: 나쁨 ~ 10: 좋음)

6-1. 와인 품질 데이터셋

- 기술통계

```
# Read the data set into a pandas DataFrame
wine = pd.read_csv('winequality-both.csv', sep=',', header=0)
wine.columns = wine.columns.str.replace(' ', '_')
print(wine.head())
```

```
# Display descriptive statistics for all variables
print(wine.describe()) → 요약통계 출력 (개수, 평균, 표준편차, 최소값, 중앙 값 등)

# Identify unique values
print(sorted(wine.quality.unique()))

# Calculate value frequencies
print(wine.quality.value_counts())
```

6-1. 와인 품질 데이터셋

- 그룹핑, 히스토그램, t 검정

```
# Display descriptive statistics for quality by wine type
print(wine.groupby('type')[['alcohol']].describe().unstack('type'))

# Calculate specific quantiles
print(wine.groupby('type')[['quality']].quantile([0.25, 0.75]).unstack('type'))
```

wine_quality.py