예제 1. 주어진 final.csv 데이터는 202x년도 2학기 통계학실험(000)의 수강생들의 기말고사 성적을 나타낸 것이다. 다음 질문에 답해보자

(단, 수강생은 총 100명이며, 기말고사는 100점 만점이다).

(1)000분반의 학생들의 기말고사 성적은 어떤 분포를 보이고 있는지를 히스토그램과 QQ-plot(분위수 대조도)를 그려보고 확인해보자.

(2)000분반의 성적부여 방침은 상대평가로, 기말고사에서 상위 30% 내에 들면 A- 이상을 받는 것이다. 그러나 ETL에는 기말고사 성적의 평균이 50점, 표준편차가 20점이라고만 공지가 되었다. 학생의 입장에서는 자신의 점수만 알고 다른 학생들의 점수를 알 수 없는데, 그렇다면 자신이 받을 학점을 어떻게 예측할 수 있을까? 기말고사 성적의 분포가 정규분포라고 가정하고, A- 이상을 받을 수 있는 최소 성적을 구해보자(소수점은 모두 올림하여 구한다. 예를 들면 50.3점, 50.7점 모두 51점으로 계산).

(3)000분반의 성적부여 방침이 절대평가로 변경되어, 기말고사에서 70점 이상을 받으면 A- 이상을 받을 수 있다. 다시 기말고사 성적이 동일한 정규분포라고 가정하고, A-이상을 받는 수강생이 약 몇 명인지 구해보자(소수점은 모두 올림하여 구한다. 예를 들면 32.4명, 32.8명 모두 33명으로 계산).

(4)000분반의 기말고사 성적의 산출 방식이 변경되어, 기존에 받았던 점수를 다음과 같이 변환하려고 한다:

$$(\text{새 점수}) = 0.9 * (\text{기존 점수}) + 30$$

이 때, 새롭게 산출된 점수의 분포를 알아보기 위해 히스토그램과 QQ-plot을 그려보자. 점수가 변환되어도 여전히 정규분포와 비슷한가? 혹은 새로운 모양의 분포인가?

*참고:
(1)의 경우 우선 다음과 같이 데이터를 불러오자:

```
final_pd = pd.read_csv("final.csv")
final_pd = final_pd.loc[:, "score"]
```

(final_pd는 작성자가 임의로 설정한 변수명이므로 자유롭게 바꿔 사용해도 무방)

예제 2. 202x년도 2학기 통계학실험(000)을 수강하는 1학년 학생인 통실이는 지난 수업에서 중심극한정리를 배우고 나서, 서로 다른 분포들의 정규분포로의 수렴 정도가 궁금해졌다. 인터넷 검색에서 로지스틱(Logistic) 분포와 균등(Uniform) 분포를 알고 난 후, 다음의 실험을 해 보려고 하였다. 질문에 답해보자.

(1)x 값의 범위가 $[-5, 5]$ 일 때 로지스틱 분포와 균등 분포를 강의노트처럼 np.linspace를 사용해서 그려보자. 단, 로지스틱 분포와 균등분포는 각각

```
scipy.stats.logistic.pdf(x, loc = 0, scale = 1)
scipy.stats.uniform.pdf(y, loc = -((np.pi)**2)/3, scale = 2*((np.pi)**2)/3)
```

를 이용한다.

(2)통실이는 중심극한정리를 실제로 확인해보기 위해 강의노트 4.3과 비슷하게 실험을 해보려고 한다. 표본의 갯수를 $n = 1, 3, 10$ 으로 바꾸어가며 각 분포마다 히스토그램 세 개를 그려보자. 즉 로지스틱 분포, 균등 분포 각각 3개씩 그려서 총 6개의 표본평균의 분포를 그려보자. 난수생성 및 표본평균 생성은 강의노트 4.3과 마찬가지로 1000회를 실시한다.

(3)(2)에서 서로 다른 두 분포가 표본 갯수가 1에서 10으로 커짐에 따라 어떻게 달라지는지 비교해보자. n이 커질수록 두 분포의 모양은 어떻게 변하는가?

(4)(2)에서 두 분포의 표본 갯수 $n = 10$으로 생성한 1000개의 표본평균들의 QQ-plot을 각각 그려보자. 로지스틱 분포와 균등 분포 각각의 표본평균의 분포는 정규분포와 유사하다고 할 수 있는가?

*참고: (2)의 경우에는 plt.subplot을 이용하면 여러가지의 그림을 한 번에 편리하게 그릴 수 있다. 다음의 코드를 이용하자:

```
plt.figure(figsize = (15, 6))
n_list = [1, 3, 10]

for i in range(6):
    plt.subplot(2,3, (i+1))
    if i < 3:
        plt.hist(로지스틱 n = n_list[i%3] 일때)
    else:
        plt.hist(균등 n = n_list[i%3] 일때)

plt.show()
```

(4)의 경우에는 현재 데이터가 list로 저장되어 있으므로 QQ-plot을 그리기 위해서는 다음과 같은 코드를 이용하여 데이터를 변환하자:

```
logistic_10 = pd.Series(logistic_list[2])
uniform_10 = pd.Series(uniform_list[2])
```

예제 3. 주어진 ames.txt 자료는 Iowa의 도시 Ames의 2006년부터 2010년 사이의 부동산 거래내역 자료이다. 5년 동안 이 지역에서 발생한 총 2930건의 부동산 거래내역이 모두 기록되어 있다. 본 예제에서는 집의 크기를 나타내는 변수인 Gr.Liv.Area를 모집단으로 사용하도록 한다. 다음의 질문에 답해보자.

(1)Gr.Liv.Area 데이터의 히스토그램과 QQ-plot을 그려보자. 정규분포와 가깝다고 할 수 있는가?

(2)Gr.Liv.Area 데이터가 정규분포를 따른다고 가정하자. 우선 `np.mean` 과 `np.std` 함수를 이용하여 데이터의 평균과 분산을 구하고, 데이터가 해당 평균과 분산을 가지는 정규분포를 따른다고 가정하자. 이 때, 집 크기의 상위 5%의 집의 크기를 구하고, 실제 상위 약 5%의 집 크기와 비교해보자.
(참고: 만약 실제로 정규분포를 따른다고 하더라도, 여기서 가정한 것처럼 데이터의 평균과 표준편차는 당연히 실제 분포의 평균과 표준편차는 아니다. 이 문제에 관해서는 차후에 추정과 검정을 배우면서 보다 자세히 공부를 하게 된다)

*참고: 실제 상위 5%의 집 크기를 구하려면 데이터를 정렬한 후, 상위 5%에 해당하는 데이터의 index를 구한 후 데이터 배열에 접근해야 한다. python 내장 정렬함수 `sorted` 및 배열의 크기(혹은 길이)를 반환해주는 내장함수 `len` 를 이용해서 실제 상위 5%의 집의 데이터를 구해보자.