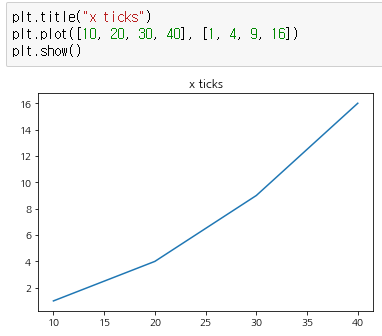
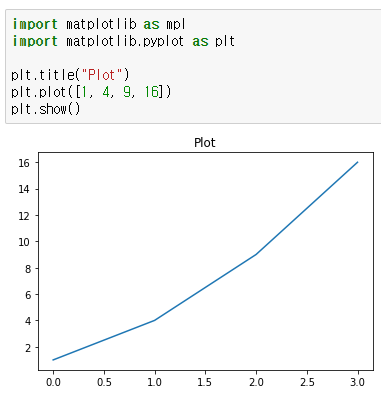
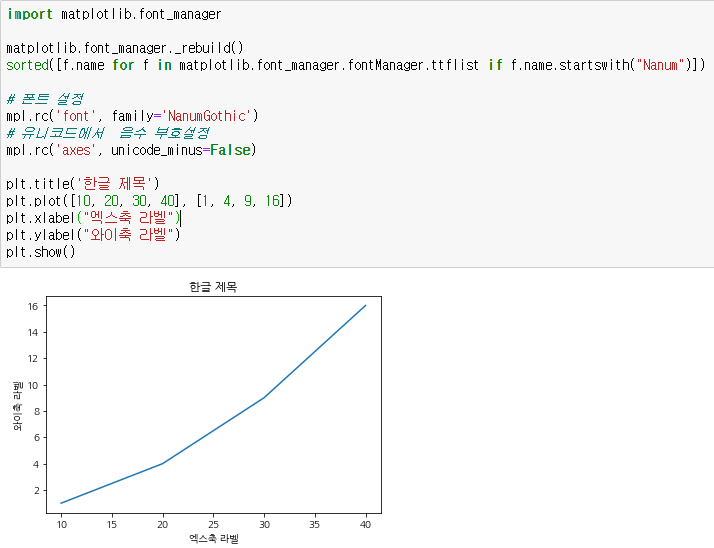
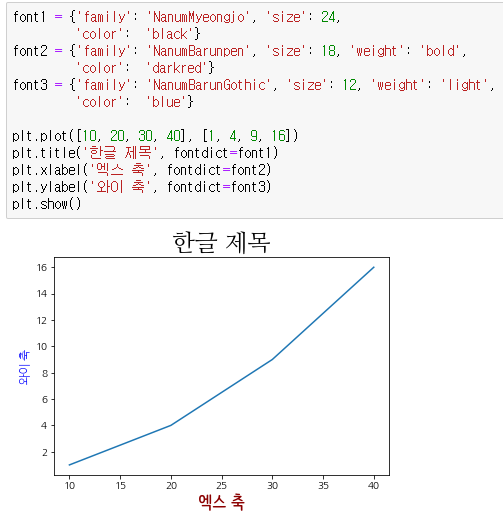
**05.01 시각화 패키지 맷플롯리브 소개**

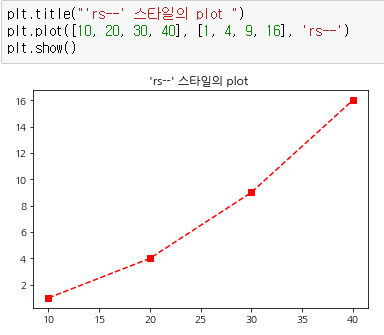
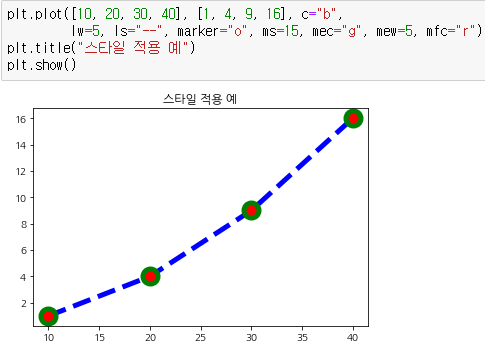
y축의 값에 따라 x값이 자동으로 정해진다. x값과 y값이 서로 순서에 맞게 좌표를 형성한다.



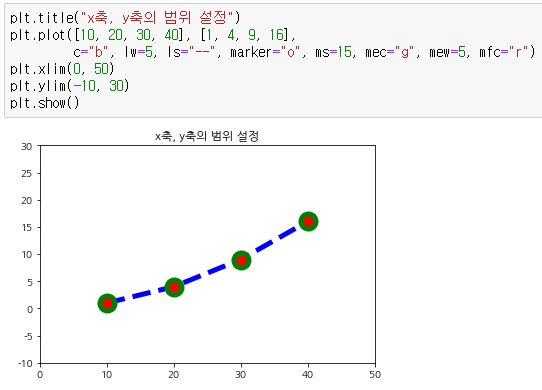
원하는 폰트로 글자들을 출력할 수 있다.



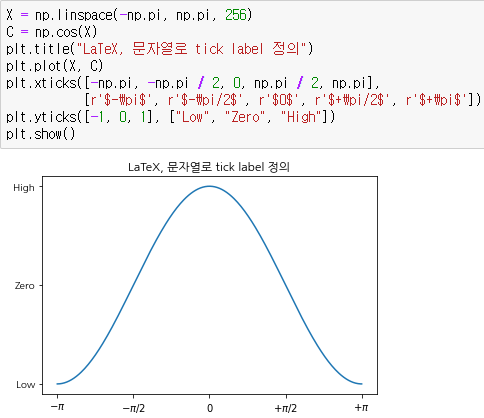
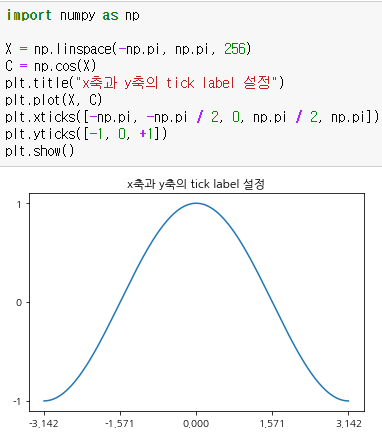
폰트변경 및 글자의 크기 조절, 색 변경까지 가능하다.



빨간 사각형과 점선으로 그래프를 표현함. 빨갛고 녹색인 원과 파란 점선으로 그래프를 표현함.

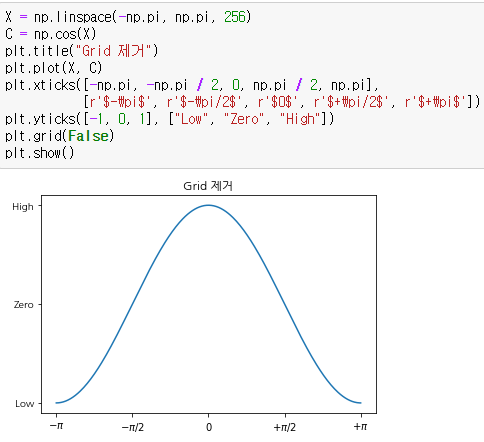
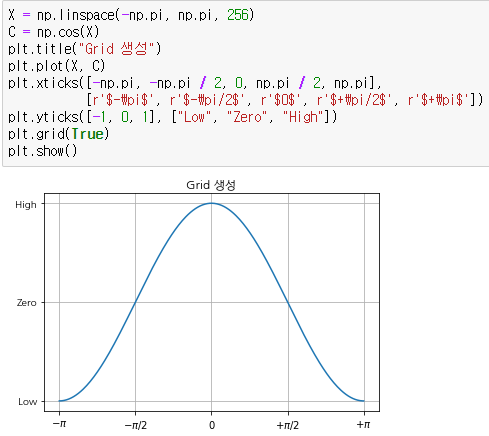


앞의 그래프에 x축, y축 범위를 새로 설정하였다.



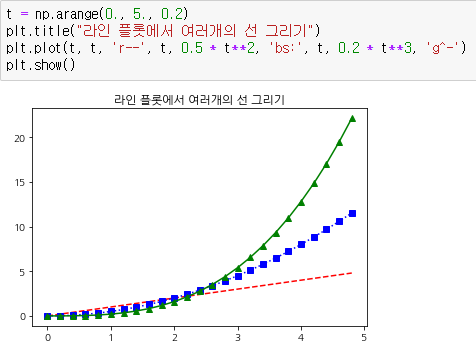
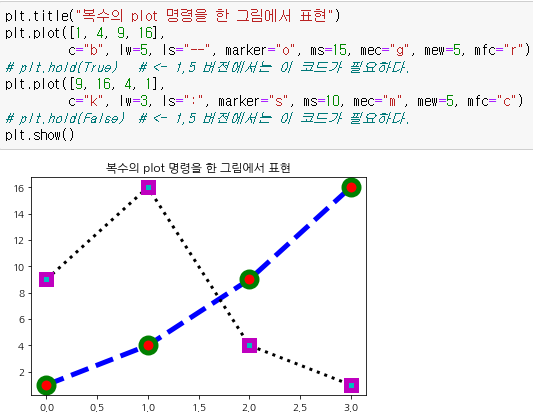
xticks 명령과 yticks 명령을 사용하여 축상에 있는 틱의 라벨들을 수동으로 설정하였다.

오른쪽의 경우는 x축에서 pi를 문자로 넣었다.

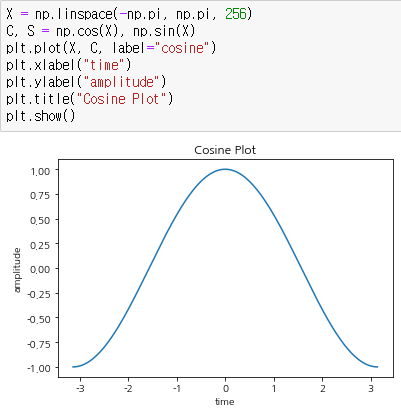
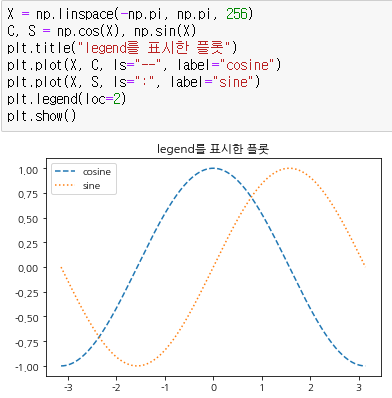


왼쪽의 경우는 틱 위치를 잘 보여주기 위한 gride line이 없고, 오른쪽의 경우는 있다.

plt.grid(False) 와 plt.grid(True) 로 구분하였다.

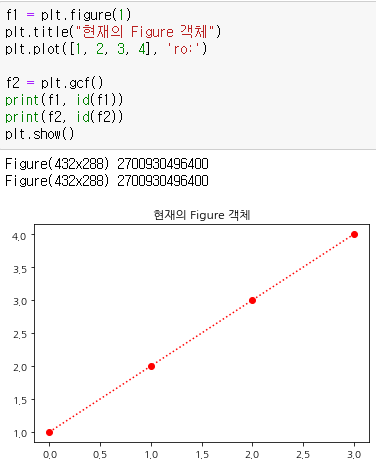


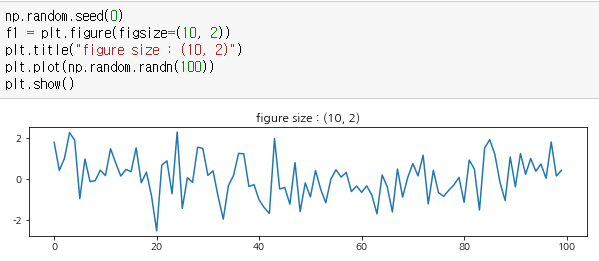
여러 개의 선을, 데이터를 살짝 변형하여 그림. 복수의 plot 명령을 하나의 그림에 겹쳐서 그림.



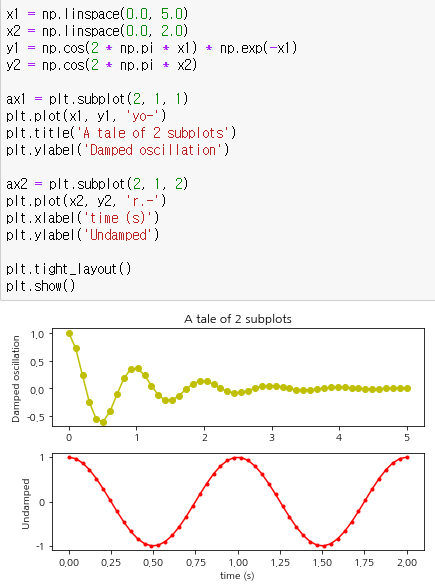
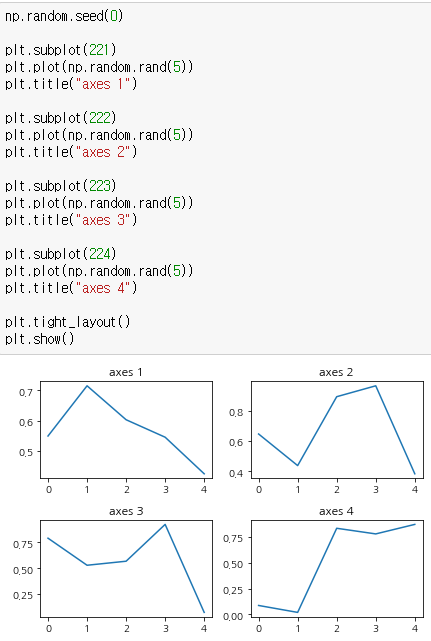
왼쪽의 경우, legend 명령으로 각 선이 무슨 자료를 표시하는지 보여준다.

오른쪽의 경우, x축과 y축이 각각 그 데이터가 의미하는 바를 표시하기 위해 xlabel 과 ylabel 명령을 사용하여 라벨을 추가하였다.



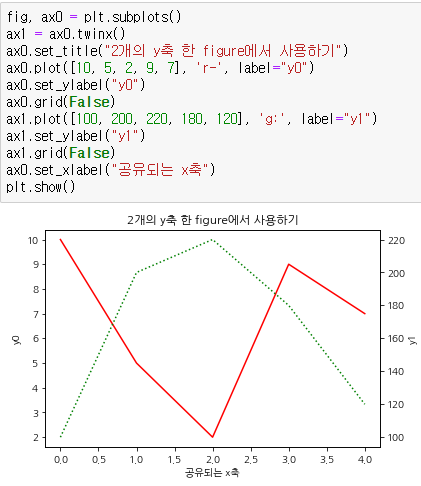
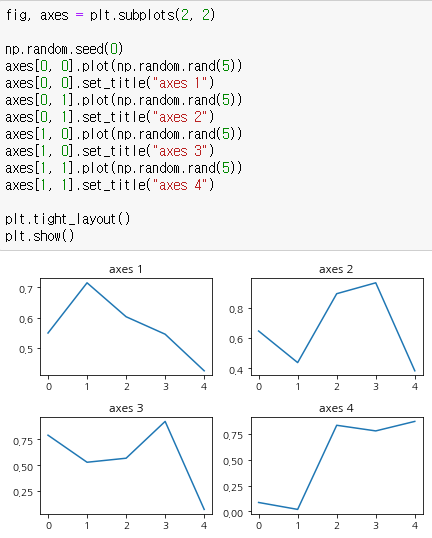


두 그림 모두 figure 명령을 사용하여 표현하였다.



왼쪽의 경우에는 tight\_layout 명령을 실행하여 플롯 간의 간격을 자동으로 맞췄다.

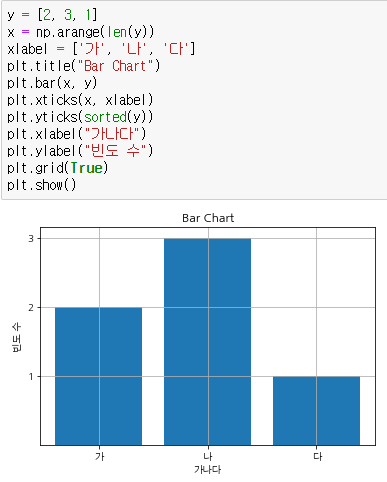
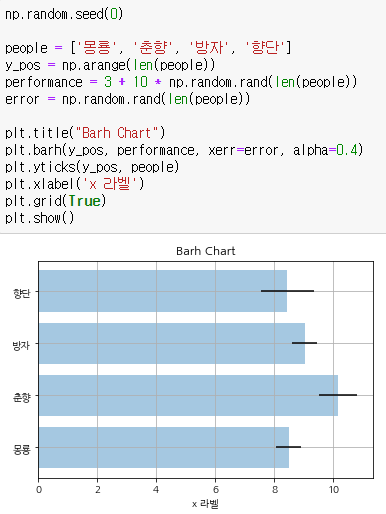
오른쪽의 경우에는 subplot 명령을 사용하여 Z 모양순서로 Axes 객체들을 생성했다.



왼쪽의 경우에는 subplots 명령으로 복수의 Axes 객체들을 동시에 생성하였다.

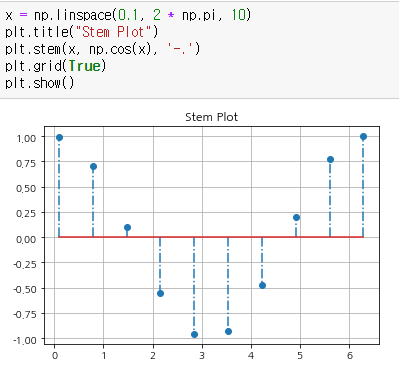
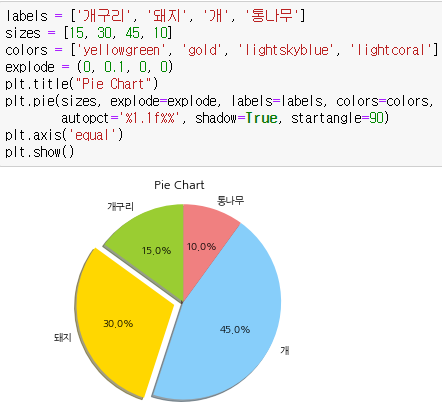
오른쪽의 경우에는 twinx 명령으로 복수의 y축을 가진 플롯을 만들었다.

x축을 공유하는 것이다.

**05.02 맷플롯리브의 여러가지 플롯**

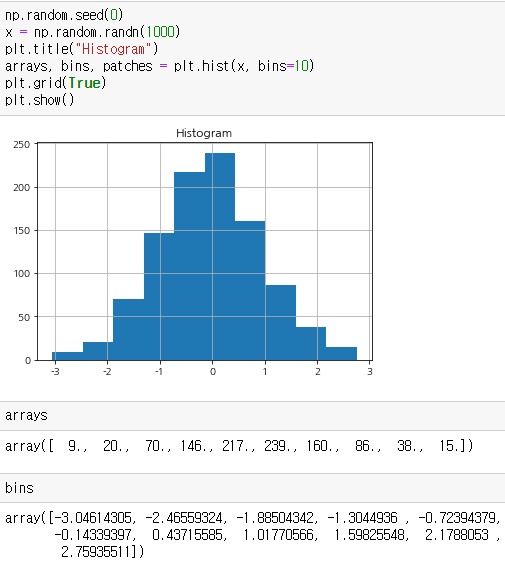
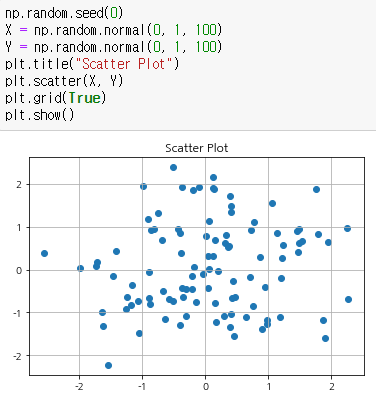
왼쪽의 경우, x데이터가 카테고리 값인 경우이며 bar 명령으로 바 차트 시각화를 할 수 있다. 이때 첫번째 인수인 left가 x축에서 바의 왼쪽 변의 위치를 나타낸다.

오른쪽의 경우, barh 명령으로 가로 방향으로 바 차트를 그렸다.



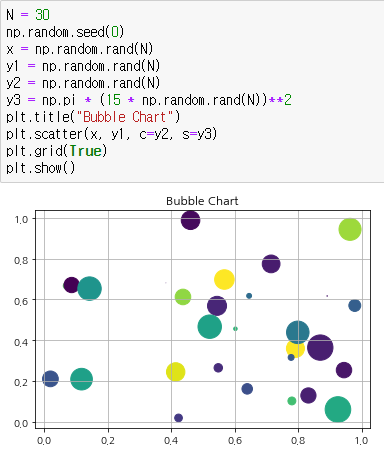
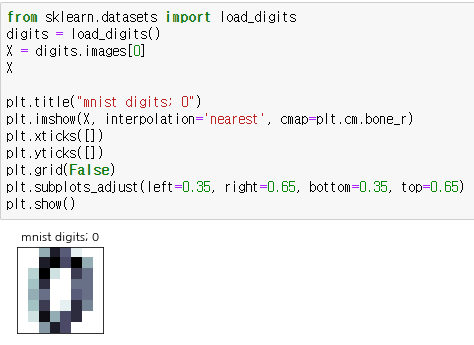
왼쪽의 경우, 바 차트와 유사하지만 폭이 없는 스템 플롯(stem plot)이다.

오른쪽의 경우, 카테고리 별 값의 상대적인 비교를 해야 할 때는 pie 명령으로 파이 차트(pie chart)를 그릴 수 있다.



왼쪽의 경우, hist 명령을 사용하여 히스토그램을 그렸다.

오른쪽의 경우, 점 하나의 위치가 데이터 하나의 x, y 값인 스캐터 플롯이다.

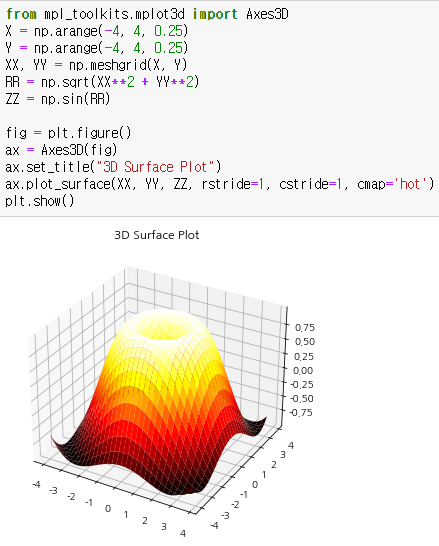
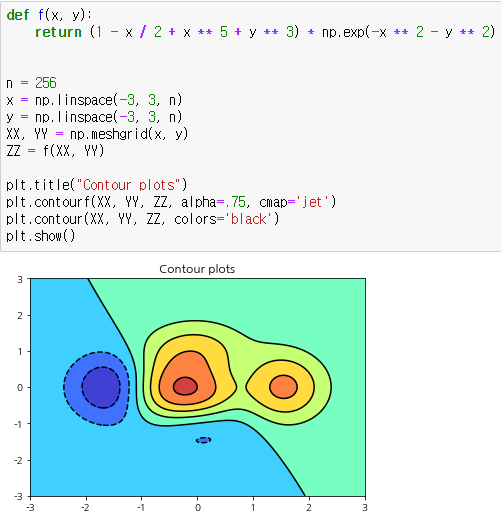


왼쪽의 경우, 점 하나의 크기 혹은 색깔을 이용하여 다른 데이터 값을 나타낸 버블 차트(bubble chart)이다. 크기는 s 인수로, 색깔은 c 인수로 지정한다.

오른쪽의 경우, 행과 열을 가진 행렬 형태의 2차원 데이터를 inshow 명령을 써서 2차원 자료의 크기를 색깔로 표시했다.



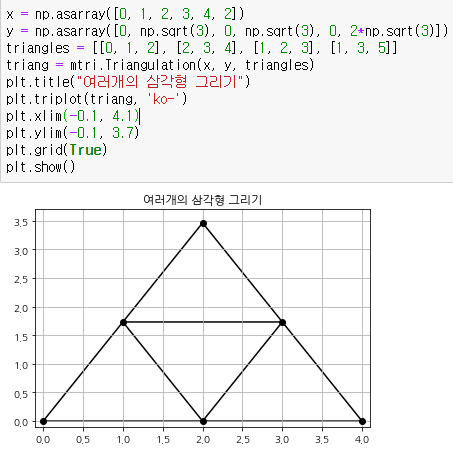
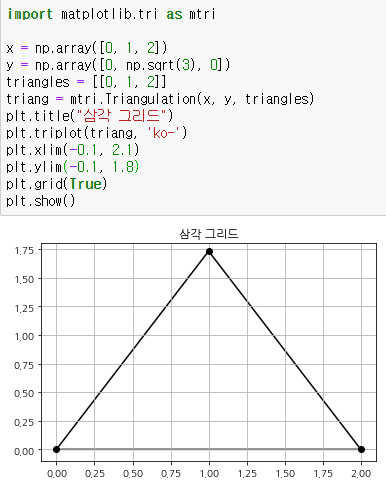
데이터 수치를 색으로 바꾸는 함수는 칼라맵(color map)이라 하며 칼라맵은 츠메 인수로 지정한다. 또한 imshow 명령은 자료의 시각화를 돕기 위해 다양한 2차원 인터폴레이션을 지원한다.



왼쪽의 경우, 컨투어 플롯이라고 한다. 입력 변수가 x, y 두 개이고 출력 변수가 z 하나인 경우 3차원 자료가 되는데, 3차원 자료 시각화 방법은 등고선을 사용하는 방법이다.

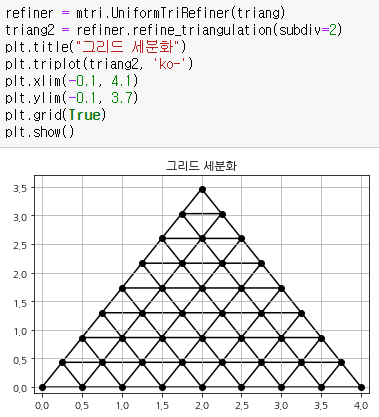
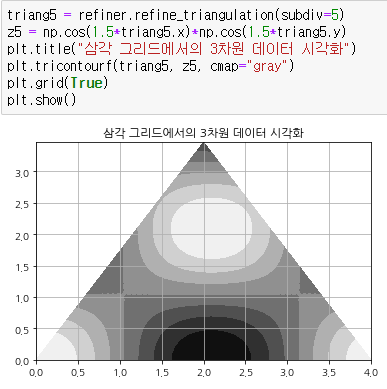
오른쪽의 경우, 3D 서피스 플롯이라고 한다. 3차원 플롯은 등고선 플롯과 달리 3차원 전용 axes를 생성하여 입체적으로 표시한다.

**05.03 맷플롯리브의 삼각 그리드 사용법**



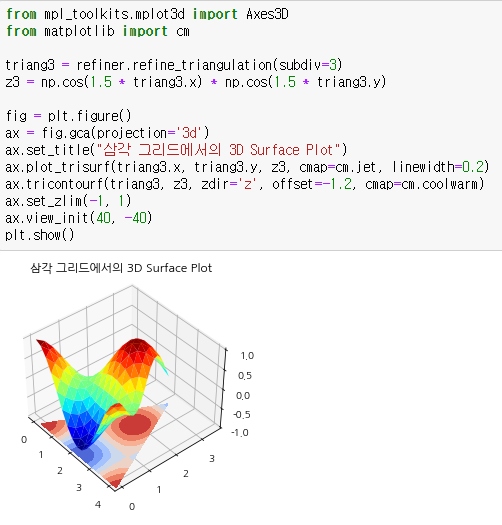
왼쪽의 경우, 세 점에 대한 각 x의 좌표, y의 좌표 그리고 triangles 까지 총 세 개의 인수를 받아 삼각형을 생성했다.

오른쪽의 경우, 왼쪽과 같은 방식으로 여러 개의 삼각형을 그렸다.



왼쪽의 경우, 위에서의 단순히 인자들의 여러 값으로 삼각형 여러 개를 생성한 게 아닌, 세분화 메서드를 사용해 몇 개 정도와는 비교가 안 될 정도로 세분화하여 생성했다.

오른쪽의 경우, 앞서 만들었던 그리드상에 tricontour, tricontourf, plot\_trisur, plot\_trisurf 등의 명령을 사용하여 2차원 등고선 플롯이나 3차원 표면 플롯을 그릴 수 있다.



앞서 말했던 명령들을 통해 3차원 표면 플롯을 만들었다.

총평: 다양한 명령들을 사용하여 코딩을 통해 여러 그래프와 도형을 생성하고 이에 대한 통계, 이미지 등 여러 자료들을 만들 수 있다는 게 기존의 코딩과는 색다른 즐거움인 것 같습니다.

**20181592 김재민**