금융프로그래밍\_기말과제

금융공학과\_202021557\_윤태원

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

멤버 변수로 현재 주식 가격, 수익률, 변동성, 순수익,주식수로 설정했습니다. 미래 주가를 self.FV로 따로 설정해 주가 변동 함수를 두 번 시행했을 때 누적되어서 진행되는 현상 방지하고 독립적인 시행이 가능하게 했습니다. 주가이동함수를 진행했을 때, 미래 값이 list형태로 들어오기 때문에 self.FV로 따로 받아 효율적으로 진행하고자 했습니다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

주가 변동 함수인 after\_years\_stock\_value함수입니다.

year,seed,size를 인자로 받아 np.random.seed(seed)로 시드를 설정하고, z = np.random.normal(0, 1, size)로 정규분표를 size만큼 추출했습니다. 그리고 pdf에 나온 공식을 토대로 코드를 그대로 ㄱ현했습니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

n년후 주가 예상 리스트의 표본 평균과 표본 표준편차를 리턴하는 함수입니다.

22번 코드로 위 주가 변동 함수를 끌어와 .mean()과 var함수를 통해 표본 평균과 분산을 구했고, 표본 분산이기 때문에 ddof를 1로 설정해 진행했습니다.

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

표본 평균과 표본 분산, 그리고 이론과의 차이 값을 비교해보았습니다.

for구문을 활용해서 size가 10부터 10^5까지 변화함에 따라 오차가 점점 줄어드는 모습을 확인할 수 있습니다. 모분산, 모평균은 pdf파일에 있는 공식을 바탕으로 구현했습니다. 표본 평균값에서 모 평균값을 빼서 절댓값을 취해 둘 사이의 오차를 확인했습니다.

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

n년후 주가 예상 리스트를 히스토그램으로 시각화한 함수입니다.

먼저 29번째 코드로 n년 후 값 리스트 및 표본 평균을 추출했습니다.

이후 st.median()함수로 self.FV의 중앙값을 추출해 self.middle에 저장했습니다.

plt.axhline()함수를 활용해서 중앙값 및 평균 값을 수평선으로 나타냈습니다.

32번 plt.plot을 활용해서 미래 값 리스트를 점으로 그래프에 표현했습니다.

plt.title()로 제목, plt.xlabel , plt.ylabel로 축 설명, legend()함수로 label로 설정했던 범례를 소개했습니다.

텍스트, 스크린샷, 도표, 번호이(가) 표시된 사진

자동 생성된 설명

10년 후 주가를 100개 추출해서 실제로 진행하면 이런 모습으로 구현이 됩니다. 점들은 예측 값 하나하나이고, 빨간 수평선이 평균값, 파란 수평선이 중앙값입니다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

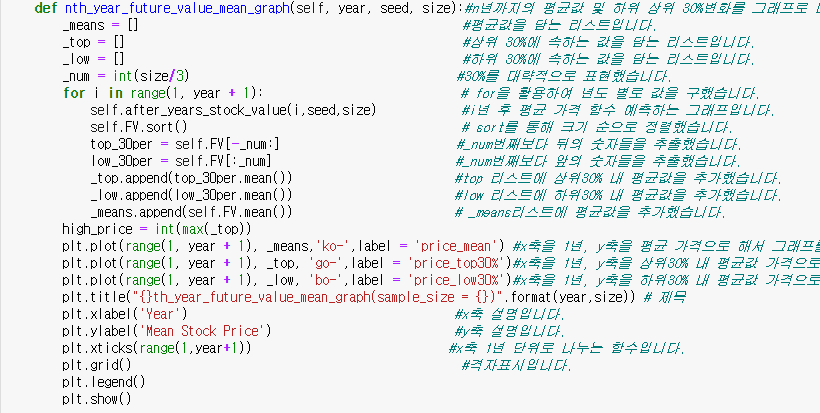
자동 생성된 설명

중앙값으로 n년까지의 미래주식가치의 변화를 그래프로 구현했습니다. \_middle함수로 리스트를 미리 설정해두고, for 구문을 활용해 1년 후의 예측주가들의 중앙값부터 year로 설정한 중앙값까지 반복적으로 추출했습니다. 그리고 46번 코드에 append를 통해 \_middle에 하나씩 추가하는 코드를 만들었습니다. 다음 plt를 활용해 그래프로 구현했습니다. 이 함수에서는 plt.ytricks를 통해 중앙값을 눈금으로 표시하고, x축은 plt.xticks를 활용해 년도 수로 눈금을 채웠습니다.

텍스트, 라인, 스크린샷, 그래프이(가) 표시된 사진

자동 생성된 설명

size는 100개로 17년 후 까지의 주가의 중앙값을 그래프로 확인해본 모습입니다. 우상향을 하는 모습을 확인할 수 있었습니다.



다음 함수는 표본평균값, 상위 약 30%의 표본 평균값, 하위 30%의 표본 평균값으로 n년까지의 미래주식가치의 변화를 그래프로 구현했습니다. 중앙값과 같은 방식으로 \_top,\_low,\_means로 각각 값을 구현했습니다. 그리고 30%는 size에 3을 나누고 int로 내림을 적용해 대략적인 30%를 구현했습니다. 다음으로 for구문을 활용하였고 이전 중앙값과 다른 부분은 sort()함수를 활용해 FV값을 크기순서대로 나열했습니다. 이후 top값은 -num번째 값보다 뒤의 숫자들을 추출하고 low는 num값보다 앞의 숫자들을 추출하였습니다. 그리고 각각의 평균 값을 구해주었습니다. 그리고 3개의 함수를 표현했기 때문에 초록,검정,파랑색으로 차이를 주었습니다.

텍스트, 그래프, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

위 함수를 size는 100개 30년으로 설정해 실행한 모습입니다. 상위 30퍼의 평균 값과 하위 30퍼의 평균 값 그리고 전체 평균 값을 비교하는 그래프의 모습입니다.

텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

다음은 per함수입니다. 순이익/발행주식수를 통해 eps(주당 순이익)값을 구하고 이를 주가에 나워서 per을 구했습니다. 그리고 print를 통해 PER값을 출력했습니다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

아래는 임의의 값으로 PER값을 구해보았습니다.