**고급 소프트웨어 실습 5주차 과제**

20171666 이예은

1. 프로그램의 동작

먼저 실습 1-1번의 결과부터 보여준다. 이후 실습 1-2의 난수 개수 n을 입력받는다.

그 이후에는 과제 1번을 수행하기 위해 생성할 난수 개수 n을 입력받는다. 이후 λ를 하나씩 입력 받고 그에 따라 생성된 난수의 평균과 분산을 출력해준다. 이를 3번 반복한다.

그 다음에는 시간 체크를 위해 생성할 난수 개수 n을 입력받는다. 걸린 시간을 출력하고 프로그램은 종료된다. 이 때, program2\_2\_b에는 현재 secant 방법으로 시간을 구하게 하고 있으므로 만약 newton 방법을 테스트 하고 싶다면 해당 함수에서 newton을 불러오는 주석 부분을 해제하고 secant를 부르는 부분을 주석처리 하면 된다.

1. 지금 개발중인 컴퓨팅 시스템의 성능 평사를 위한 시스템 모델링 작업을 수행하려 한다. 이 시스템을 구성하는 한 컴포넌트가 작동 중 고장이 나는 상황을 통 계적으로 모델링하여 보자. 시스템 가동 시점부터 이 컴포넌트가 처음 고장이 날 때까지의 시간 X (≥ 0)를 강의 시간에 설명한 지수 분포로 모사하려 한다.
2. 지수 분포의 인자 λ (> 0)에 대해 확률 변수 X의 기대 값이 E[X] = 1/λ이고 분산이 Var[X] = 1/λ^2이라는 사실을 고려하여, 세 가지 서로 다른 λ 값을 설정하라.

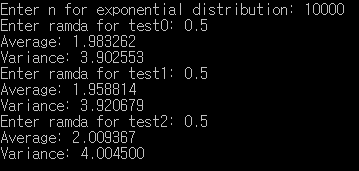
* 0.5, 1, 1.5로 설정하였다.

1. 자신이 정한 λ에 해당하는 지수 분포 각각에 대하여, inversion 방법을 사용하여 자신이 정한 충분히 큰 개수만큼 이 지수 분포를 따르는 난수를 생성하라.

* 10000개의 난수를 생성하여 평균과 분산 값을 구했다.

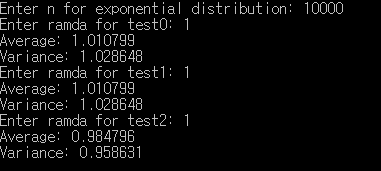
1. 자신이 생성한 난수들을 사용하여 평균값과 분산 값을 구한 후 이론적인 값들과 얼마나 일치하는지 분석하라.

λ가 0.5일 때 총 3번 난수를 생성한 결과는 다음과 같다.



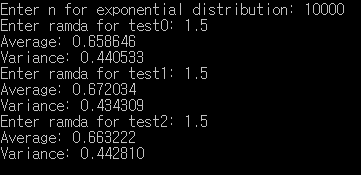
0.5일 때 이론적인 평균은 1/0.5 = 2, 분산은 1/(0.5\*0.5) = 4. 난수로 구해본 평균과 분산은 대체적으로 2와 4에 가까운 값을 보인다는 것을 확인할 수 있다.

λ가 1일 때 총 3번 난수를 생성한 결과는 다음과 같다.



이론적인 평균은 1/1 = 1, 분산은 1/1^2 = 1 이므로 이론적인 값과 비슷한 값을 보인다는 것을 확인할 수 있다.

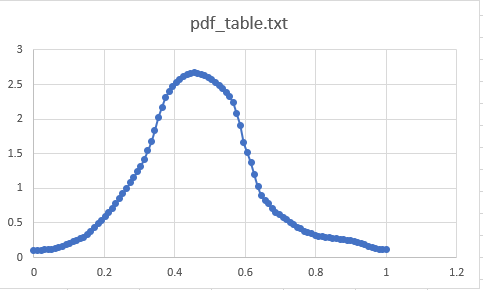
λ가 1.5일 때 총 3번 난수를 생성한 결과는 다음과 같다

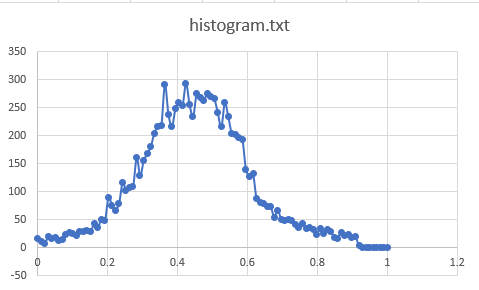


이론적인 평균은 1/1.5 = 0.666… , 분산은 1/(1.5^2) = 0.444… 이므로 이론적인 값과 비슷한 값을 보인다는 것을 확인할 수 있다.

1. 여러분이 실습 시간에 작성한 코드 (프로그램 2-2)를 잘 가다듬어 문제없이 그리고 효율적으로 수행되도록 수정하라 (프로그램 2-2(a))
2. 주어진 확률 밀도 함수에 대하여 자신의 코드가 올바르게 난수를 생성하는 지 통계적으로 확인하라. 이를 위하여 아래에서 설명하는 내용의 코드 (프로그램 2-3)을 작성한 후 이를 활용하라.

코드가 올바르게 난수를 생성하는지를 확인하기 위해 pdf\_table 파일과 각 구간에 난수가 얼마나 만들어졌는지를 저장한 histogram 파일을 확인해 그래프를 그려보면 다음과 같다.





약간의 오차는 있지만 대체적으로 비슷한 모양의 그래프를 그린다. 그러므로 성공적으로 난수를 생성한 것을 확인할 수 있다.

1. 비선형 방정식의 근을 구하는 방법 구현 시 앞에서 사용한 Bisection 방법을 Secant 방법으로 대치한 난수 생성 프로그램을 작성하라 (프로그램 2-2(b)). 물론 자신의 프로그램이 제대로 작동하는지 실험적으로 확인하라.

Newton-Raphson 방법은 bisection 방법을 일정 횟수동안 반복하여 초기 값을 설정하도록 하였다. Secant 방법의 경우는 newton 방법과 달리, [0,1] 구간에 근이 있는 것이 확실하므로 bisection처럼 x1을 1, x0를 0으로 설정하였다. Newton 방법에서 적절한 bisection 횟수는 대략 3-5번 정도로 추정된다.

각각 1,3,6 번 정도를 돌렸을 때 newton에서 걸리는 시간이다. 난수는 총 10000개를 생성했다.

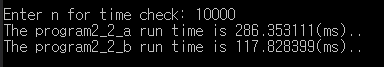






1. 충분히 큰 난수의 개수 nr에 대해 프로그램 2-2(a)와 프로그램 2-2(b)를 수행시킨 후, 각 방법이 난수를 생성하는데 걸린 시간을 비교하라

10000개의 난수에 대해 bisection에서 걸린 시간과 secant에서 걸린 시간을 우선 보면 다음과 같다.



여기서 2\_2\_a가 bisection, 2\_2\_b가 secant 방법이다. 대략 2배 이상의 시간차이가 나는 것을 확인할 수 있다.

이제 bisection과 newton에서 걸린 시간을 확인해보자.



2\_2\_a가 bisection, 2\_2\_b가 newton 방법이다. 거의 4-5배 정도의 시간 성능 차이가 나는 것을 확인할 수 있다. 또한 newton과 secant 방법도 2배가량의 성능 차이가 나는 것을 확인할 수 있다.