

Midterm Project : M.A.F.I.A. in the morning

1. 시민과 마피아만 존재하는 게임

1-1. 구현 방법

- 게임 참가인원의 직업을 string형으로 갖는 List를 생성
- 낮
 - 마피아는 마피아를 제외한 시민 중 한 명을 각각 투표
 - 시민은 모든 인원 중 한 명을 투표
 - 투표는 완전한 랜덤으로 이루어지고 그 중 최빈값을 찾아 맨 앞의 값을 반환
 - 참가인원 리스트에는 마피아가 먼저 할당 되어 있고 마피아 끼리는 팀을 이룰 수 있기 때문에 마피아의 의견을 먼저 따른다는 가정
- 밤
 - 마피아는 각자 시민 중 한 명을 투표
 - 투표는 완전한 랜덤으로 이루어지고 그 중 최빈값을 찾아 맨 앞의 값을 반환
- 각 게임은 100번 진행 후 시민의 승리 수를 %로 반환
- 정해진 게임 인원별 승률 데이터와 마피아/참가자 비율에 대한 승률 데이터를 구현

```
메모 포함[최1]: Function 게임{
  if 마피아가 존재하고 시민수보다 적을 때{
    낮
      투표
    밤
      마피아 살인
  }
  if 마피아가 없을 때
    ret 마피아 승
  else
    ret 시민 승
}
```

1-2. 실험

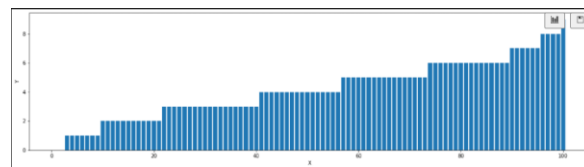
승률 데이터는 실험 인원수에 대한 시민의 승률로 나타낸다. (마피아 승률 = 100% - 시민 승률)

- 아래는 전체인원 10명 중 마피아가 2명 일 때의 데이터를 나타낸다.

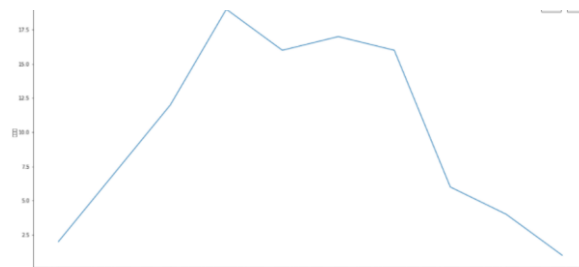
Citizens to Win(%)	
1	1.0
2	7.0
3	6.0
4	4.0
5	4.0
...	...
96	3.0
97	5.0
98	5.0
99	5.0
100	5.0

시민확률 Data Frame

{ 횡수, 시민 승률 }



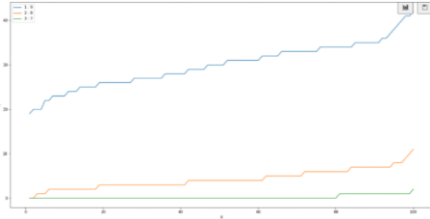
100회의 시민 승률 데이터를 정렬하여 표현(x:횡수, y:승률)



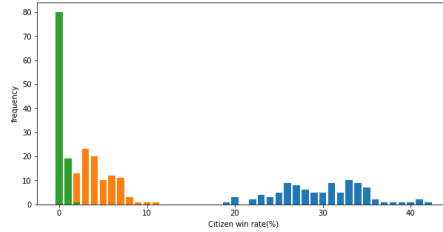
위의 bar chart를 빈도수로 표현(x:승률, y:빈도수)

메모 포함[최2]: 3~5%사이의 승률이 가장 빈도수가 높음
표본 데이터가 많을수록 빈도수 그래프가 정규분포그래프와 유사한 결과를 얻을 수 있다.

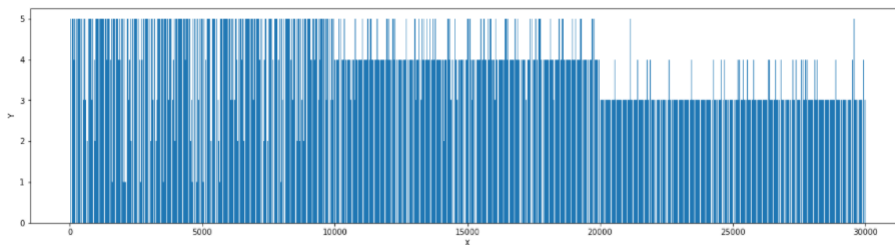
2) 전체인원 10명 중 마피아가 1명(파랑), 2명(주황), 3명(초록) 일 때



각각의 시민의 승률 데이터를 정렬한 Plot Chart
(x:횟수, y:승률)



왼쪽의 그래프를 빈도수로 정리(x:승률, y:빈도수)



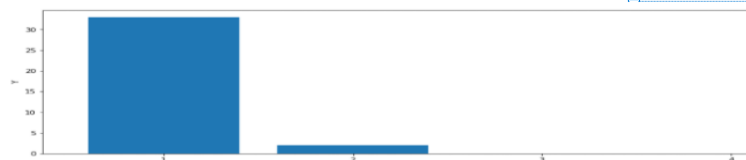
위 상황에 대한 게임 종료일 나열 (상황 1,2,3에 대해 10,000단위 구분)

메모 포함[최3]: 게임 인원이 10 명일 때, 마피아비율이 20%이상일 때부터 시민의 우승 확률이 대폭 감소하고 30%이상일 때 부터는 0에 가깝다

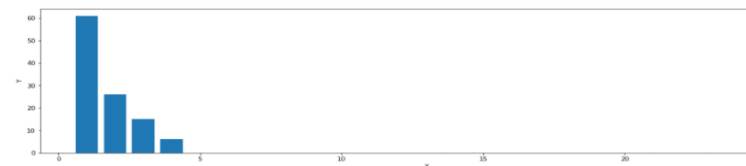
메모 포함[최4]: 아래의 bar chart는 마피아가 1명인 게임부터 차례대로 게임 종료일을 나열 한 것이다. 각 게임마다 100번의 게임의 확률을 100번 계산하므로 마피아 1명일 때 0-10,000구간, 마피아 2명일 때, 10,000-20,000구간, 마피아 3명일 때, 20,000~30,000구간이다.

마피아의 비율이 올라 갈수록 게임 종료일의 평균이 낮게 나오는 것을 확인 할 수 있다.

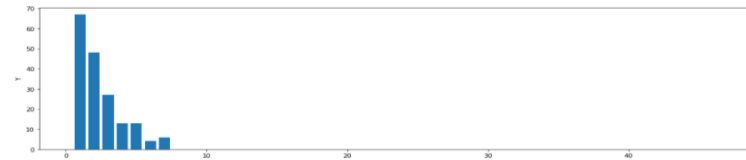
3) 전체 인원을 정한 후 마피아를 한 명씩 늘려 확인한 데이터



Case1: 전체인원 10명, 마피아가 1 명에서 4명 까지 일 때의 시민 승률(x:마피아 수, y:시민승률)



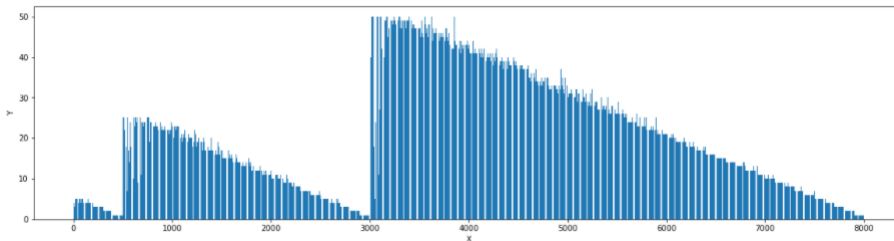
Case2: 전체인원 50명, 마피아가 1 명에서 24명 까지 일 때의 시민 승률(x:마피아 수, y:시민승률)



Case3: 전체인원 100명, 마피아가 1 명에서 49명 까지 일 때의 시민 승률(x:마피아 수, y:시민승률)

메모 포함[최5]: 아래 plot chart들은 마피아의 수를 1 명부터 전체 게임인원 절반 미만 일 때 까지 증가하며 시민 승률을 계산한 그래프이다.

각각의 chart를 같은 scale로 나타내었는데 게임 인원이 늘어날수록 같은 마피아의 비율 조건에서 더 낮은 승률이 나오는 것을 확인 가능하다.



위 상황에 대한 게임 종료일 나열, x축 기준 500, 2500, 5000구분(x:횟수, y: 게임 종료일)

1-3 분석

- 모든 확률을 Random하게 부여해 각 상황마다의 편차가 큼
- 마피아와 시민의 비율을 보통 1:3 이하로 설정하지만 위 상황에서는 1:10만 되어도 시민의 승리확률이 대폭 하락함
 - 실제 게임에서는 시민이 마피아를 추리하여 맞출 수 있는 확률은 턱이 지날수록 계속 상승함
 - 회의를 통해 마피아를 찾을 수 있는 단서를 무시하고 모든 상황을 랜덤으로 부여하였기 때문에 이러한 결과가 나온 것이라 예측
- 원활한 진행을 위해 게임인원이 10명일 때 마피아 1명, 50명일 때 2명, 100명일 때 3명이 적합하다는 그래프가 산출됨
 - 참가자가 많아질수록 마피아가 죽을 확률이 감소함
 - 시민의 승률은 인원이 같은 비율로 변동되었을 때에 같은 비율을 유지하지 않음
 - 참가자가 많아질수록 마피아 비율에 따른 시민의 승률은 더욱 적다
 - 참가자가 많아져도 그 비율만큼 마피아의 수를 늘려주는 것은 적합하지 않다고 볼 수 있음
- 게임 인원과 게임 소요시간은 비례함
 - 또한 마피아의 수와 반비례함
 - 게임인원이 많아지고 마피아의 수가 줄어들수록 소요시간은 길어짐
 - 마피아의 수가 많아지고 게임 인원이 줄어들수록 소요시간은 짧아짐
- 시민의 우승 확률에 대한 빈도수를 chart로 나타내면 중위값 부근에서 대부분의 값들이 분포하는 것을 알 수 있다.

메모 포함[최6]: 게임 종료일은 전체 인원수가 증가하여도 마피아의 비율에 따라 linear하게 감소하는 것을 확인 가능하다.

전체 10명일 때 0-500구간,
전체 50명일 때, 500-3,000구간,
전체 100명일 때, 3,000~8,000구간이다.

2. 시민, 마피아와 경찰이 존재하는 게임

2-1 구현 방법

- 1번(마피아 시민만 존재)을 기반으로 경찰을 추가.
- 경찰은 마피아를 찾음
 - 편의를 위해 구현은 낮에 하였으나 살인이 끝난 후 새벽에 마피아를 검거한다는 설정
- 랜덤하게 참가자중 한 명을 찾고 마피아인지 확인
 - 마피아가 맞다면 인덱스를 반환
- 낮 투표 중 경찰의 신뢰도를 랜덤하게 반환
 - 신뢰도는 참가 인원 중 몇 명을 신뢰하게 만들었는지로 확인
 - 과반수 이상 일 경우 기존 투표결과를 경찰이 찾은 마피아로 변경

2-2 실험

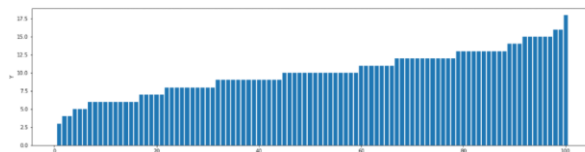
승률 데이터는 실험 인원수에 대한 시민+경찰의 승률로 나타낸다.

- 1) 전체 인원 10명 중 마피아2, 시민 7, 경찰 1명일 때의 시민의 승률

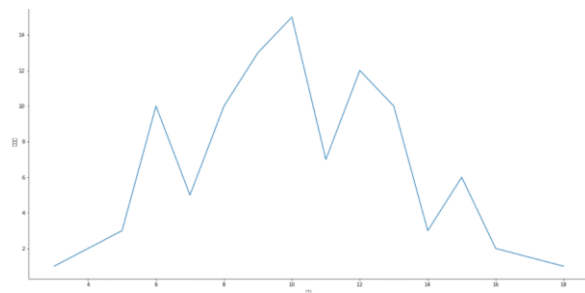
Citizens to Win(%)	
1	7.0
2	15.0
3	11.0
4	11.0
5	15.0
...	...
96	12.0
97	6.0
98	14.0
99	10.0
100	12.0

시민확률 Data Frame

{ 횟수, 시민 승률 }



100회의 시민 승률 데이터를 정렬하여 표현(x:횟수, y:승률)



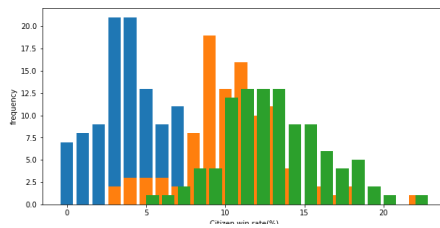
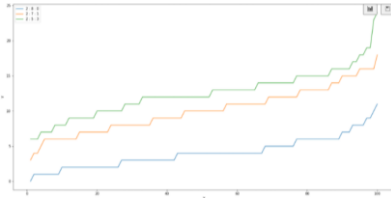
위의 bar chart를 빈도수로 표현(x:승률, y:빈도수)

```
메모 포함[최7]: Function 마피아 탐색{
  인원중 랜덤하게 한 명 지목
  If 경찰일 때
    재귀하여 다시 구함
  Elif 마피아일 때
    찾은 마피아를 반환
  Else 시민 일 때
    None
}
```

```
Function 마피아 검거{
  If 경찰이 존재하지 않을 경우
    함수 실행하지 않음
  If 난수 생성, 시민수의 절반 이상일 때
    Function 마피아 탐색
    탐색하여 찾은 마피아를 반환
}
```

메모 포함[최8]: 전체적으로 마피아, 시민만 존재 할 때보다 경찰 1명의 투입으로 인해 승률이 확연히 상승 하였다. 랜덤확률의 추가로 빈도수의 파형에 변동이 생긴 것을 확인 할 수 있다.

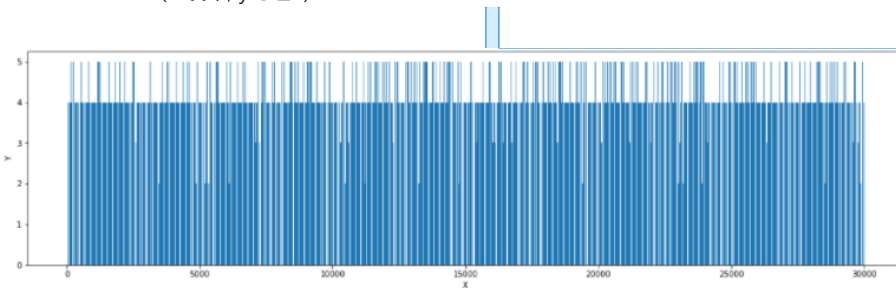
2) 전체인원 10명 중 경찰이 1명(파랑), 2명(주황), 3명(초록) 일 때



메모 포함[최9]: 경찰이 추가 될 때마다 시민의 승률이 확연히 올라가는 것을 확인 할 수 있다.

각각의 시민의 승률 데이터를 정렬한 Plot Chart
(x:횟수, y:승률)

왼쪽의 그래프를 빈도수로 정리(x:승률, y:빈도수)

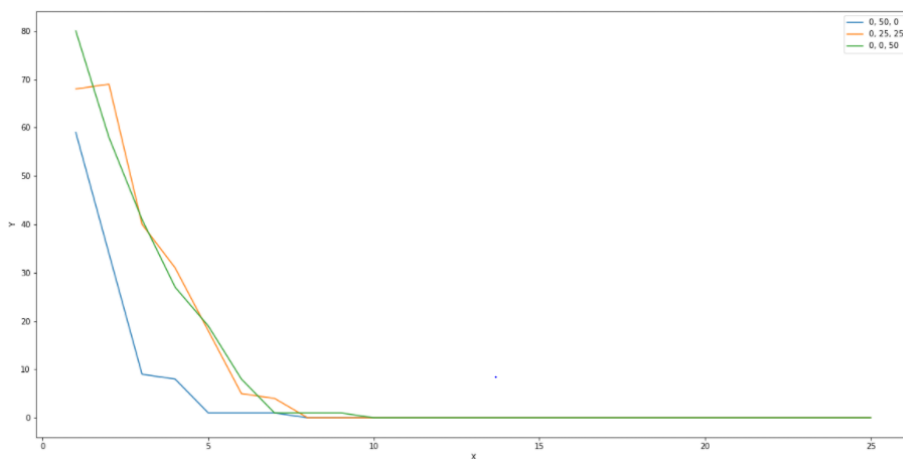


메모 포함[최10]: 경찰의 수가 게임 종료일에는 전체적으로 영향을 주지는 않지만 경찰이 늘어 날수록 편차가 커지는 것을 확인 가능하다.

경찰 1명일 때 0-10,000구간,
경찰 2명일 때, 10,000-20,000구간,
경찰 3명일 때, 20,000~30,000구간이다.

위 상황에 대한 게임 종료일 나열 (상황 1,2,3에 대해 10,000단위 구분)

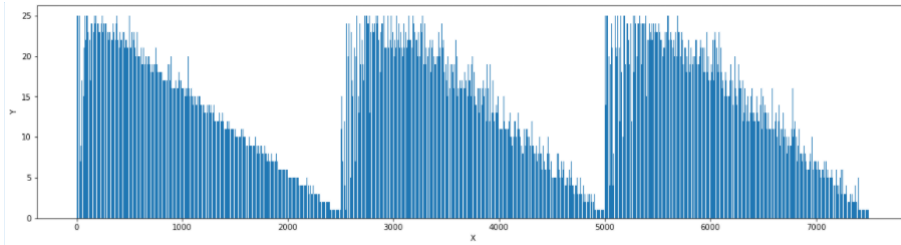
3) 전체 인원 50명 중 경찰이 0명(파랑), 25명(주황), 50명(초록) 일 때 마피아를 한 명씩 늘려 시민의 우승 확률을 확인한 그래프



메모 포함[최11]: 경찰이 존재하지 않는 경우와 존재하는 경우와는 승률에 약 10~20%가량 차이를 보인다. 또한 경찰이 존재하는 두 가지 경우는 경찰이 25명이나 차이가 남에도 불구하고 오차범위 내에서 겹치는 모습을 확인 가능하다

50명 중 경찰이 50인 경우는, 마피아를 제외한 모든 인원이 경찰인 경우.

각각의 시민의 승률 데이터를 정렬한 Plot Chart (x:횟수, y:승률)



위 상황에 대한 게임 종료일 나열, x축 기준 2500구분(x:횟수, y: 게임 종료일)

메모 포함[최12]: 순서대로 경찰이 존재하지 않는 경우, 25명인 경우, 마피아를 제외한 모든 인원이 경찰인 경우 이다. 경찰이 많이 존재 할수록 게임 종료일의 편차가 커진다.

경찰 0명일 때 0-2,500구간,
경찰 25명일 때, 2,500-5,000구간,
경찰 50명일 때, 5,000~7,500구간이다.

2-3 분석

- 경찰의 합류로 시민의 우승확률이 대체적으로 상승함
 - 경찰이 많아질수록 상승하지만 시민 전원이 경찰이어도 큰 변화를 보이지는 않음
 - 경찰에 한 턴에 꺼낼 수 있는 카드는 하나기 때문
 - 경찰의 수가 많으면 모두 죽을 확률이 더 낮아지기 때문에 게임이 후반으로 갈수록 더 높음
- 랜덤확률이 추가로 더해졌기 때문에 그래프에 변화가 생김
 - 게임이 종료되는 시점의 편차가 커짐
- 경찰의 역할에 따라 승률, 진행시간 모두 영향을 줌.
 - 검거에 성공했을 경우, 시민선택에 성공했을 경우에 시민의 우승확률을 상승시킴

3. 시민과 마피아, 그리고 메딕이 존재하는 게임

3-1 구현 방법

- 1번(마피아 시민만 존재)을 기반으로 메딕을 추가
- 메딕은 살릴 사람을 무작위로 정함
 - 밤에 마피아가 지목한 사람과 같은 사람을 살릴 경우 죽이지 않고 진행
- 실험은 세 가지 케이스로 진행
 - 밤에 랜덤으로 사람을 살리는 가장 기본적인 경우
 - 밤에 우연히 모든 사람을 살리는 경우
 - 밤에 자신만을(여러명일 경우 메딕중 한 명) 살리는 경우

3-2 실험

- 1) 마피아2, 시민 7, 메딕 1명일 때의 시민의 승률

Citizens to Win(%)	
1	7.0
2	4.0
3	8.0
4	4.0
5	6.0
...	...
96	1.0
97	5.0
98	4.0
99	4.0
100	5.0

Case1: 랜덤으로 살림

Citizens to Win(%)	
1	6.0
2	5.0
3	7.0
4	9.0
5	15.0
...	...
96	12.0
97	14.0
98	13.0
99	8.0
100	11.0

Case2: 매번 살림

Citizens to Win(%)	
1	5.0
2	4.0
3	2.0
4	7.0
5	2.0
...	...
96	5.0
97	7.0
98	5.0
99	5.0
100	6.0

Case3: 자신만을 살림

메모 포함[최13]: Case1:

```
Function 랜덤으로살림{
    난수발생
    If 발생한 난수 == 마피아 지목
        ret 낮으로 감
}
```

Case2:

```
Function 모두 살림
    ret 낮으로 감
}
```

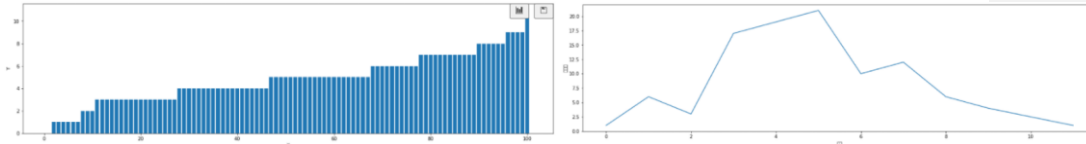
Case3:

```
Function 자신만살림{
    메딕중에서 난수 발생
    If 발생한 난수 == 마피아 지목
        ret 낮으로 감
}
```

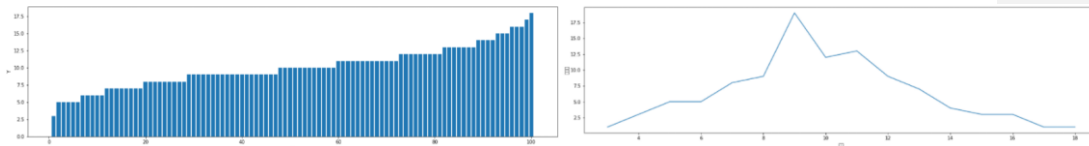
메모 포함[최14]: 데이터를 분석 해 보면

case1 < case3 < case2
순으로 시민이 우승할 확률이 높게 나오는 것을 확인
가능하다

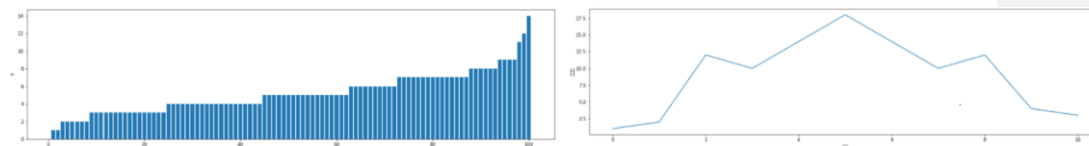
2) case 1,2,3 의 데이터를 정렬하여 표현(x:횟수, y:승률)하고 이를 빈도수로 표현(x:승률, y:빈도수)



Case1 정렬한 시민 승률(랜덤으로 1명 살림)

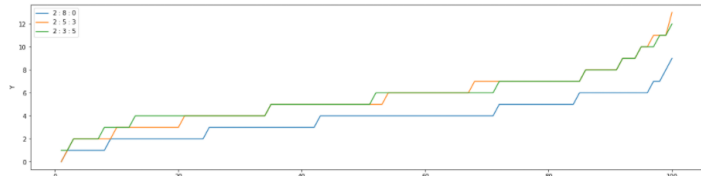


Case2 정렬한 시민 승률(매일 밤 1명 살림)

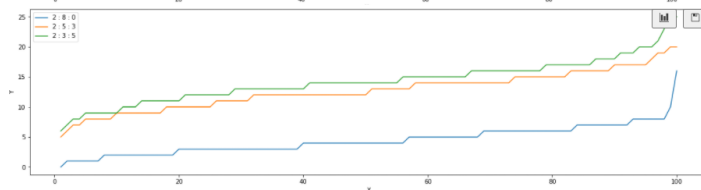


Case1 정렬한 시민 승률(메딕 자신을 살림)

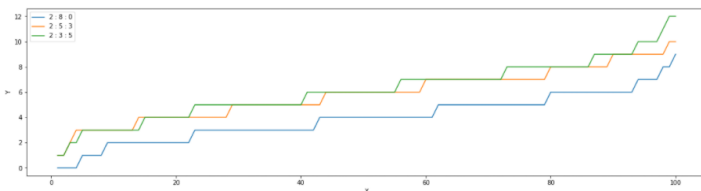
3) case 1,2,3에서 마피아가 2명 일 때, 각각 메딕의 수를 0명(파랑), 3명(주황), 5명(초록)으로 변동시키며 시민의 승률을 나타냄(x: 게임 횟수, y: 시민 승률)



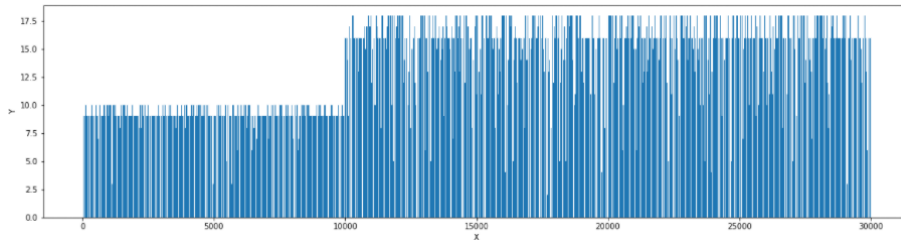
Case1: 랜덤



Case2: 매번 살림

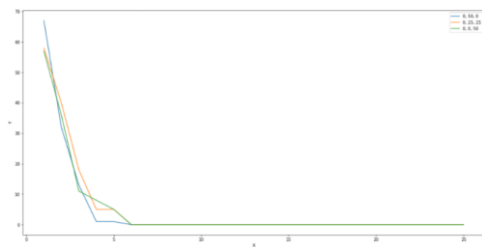


Case3: 자신만 살림

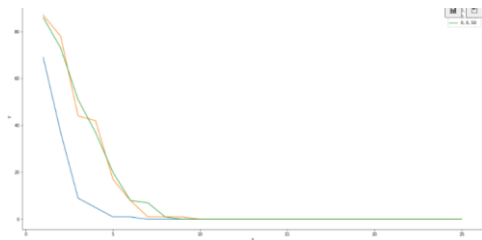


Case2: 위 상황에 대한 게임 종료일 나열 (상황 세 가지에 대해 10,000단위 구분)

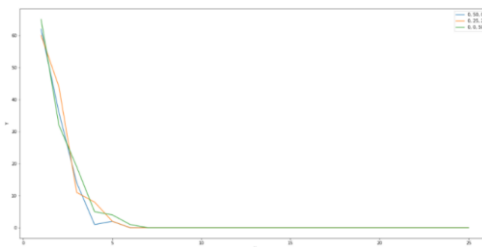
4) 전체 인원 50명 중 메딕이 0명(파랑), 25명(주황), 50명(초록) 일 때 마피아를 한 명씩 늘려 시민의 우승 확률을 확인한 그래프(x: 마피아 수, y: 시민 우승 확률)



Case1



Case2

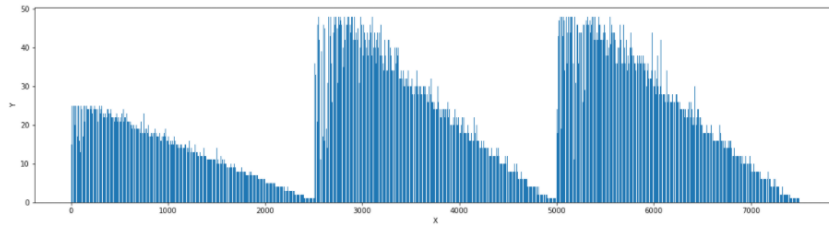


Case1

메모 포함[최15]: 메딕의 능력이 발생했을 때의 데이터를 확인하기 위해 case2의 자료만 가지고 왔다. 메딕이 사람을 살리는 빈도가 높을수록 게임 진행일이 최대 2배까지 증가되는 것을 확인 가능하다.

메모 포함[최16]: 1)에서 Data Frame으로 확인 한 것처럼

case1 < case3 < case2 순으로 시민이 우승할 확률이 높게 나오는 것을 확인 가능하지만 랜덤으로 사람을 살릴 때나, 메딕만 살릴 경우에는 그 차이가 크지 않은 것을 확인 가능하다.



Case2: 위 상황에 대한 게임 종료일 나열 (상황 세 가지에 대해 2,500단위 구분)

메모 포함[최17]: 3)에서 확인 한 것 처럼 메딕이 사람을 살리는 빈도가 높을수록 게임 진행일이 최대 2배 까지 증가되는 것을 확인 가능하다.

3-3 분석

1. 메딕이 랜덤으로 사람을 살릴 경우

- 게임 인원이 100명일 때 메딕의 수를 늘려도 시민이 이길 확률에 큰 영향을 주지 않았음
- 사람을 살렸을 경우 게임의 종료시간이 늘어남

2. 메딕이 매번 사람을 살릴 경우

- 밤에 마피아와 메딕 모두 상의를 통해 한 명을 지목 하게 됨
- 밤에 죽는 사람은 존재하지 않음
- 메딕이 존재하지 않을 때 보다 게임 진행 시간이 약 두배 증가함
- 시민만이 죽는 밤에 살인이 일어나지 않기 때문에 시민이 이길 확률이 소폭 상승함

3. 메딕이 매번 메딕을 살릴 경우(자신을 살릴 경우)

- 시민이 죽을 확률을 줄여주는 메딕의 생존 확률을 조금 줄여 줌

4. 시민과 마피아, 그리고 경찰과 메딕 모두 존재하는 게임

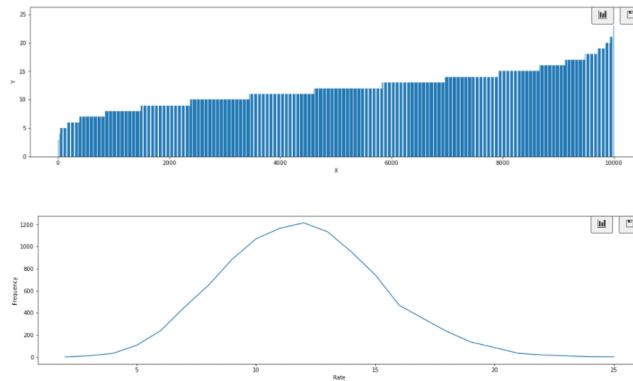
4-1. 구현 방법

- 1번(마피아 시민만 존재)을 기반으로 경찰과 메딕을 추가
- 낮에는 경찰을, 밤에는 메딕을 추가
 - 메딕은 시민을 랜덤하게 살리는 Case1의 경우로 사용

4-2. 실험

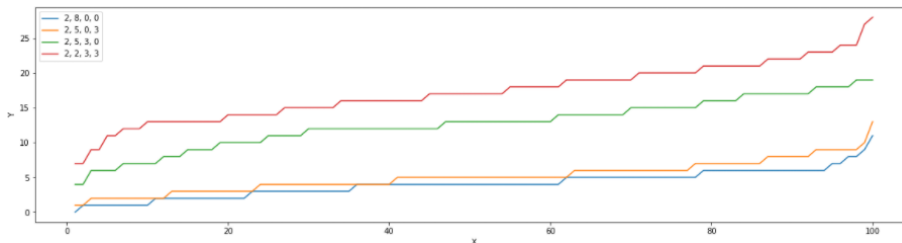
1) 마피아2, 시민 6, 경찰 1명, 메딕 1명일 때의 시민의 승률

Citizens to Win(%)	
1	15.0
2	11.0
3	7.0
4	12.0
5	14.0
...	...
96	14.0
97	12.0
98	12.0
99	7.0
100	13.0



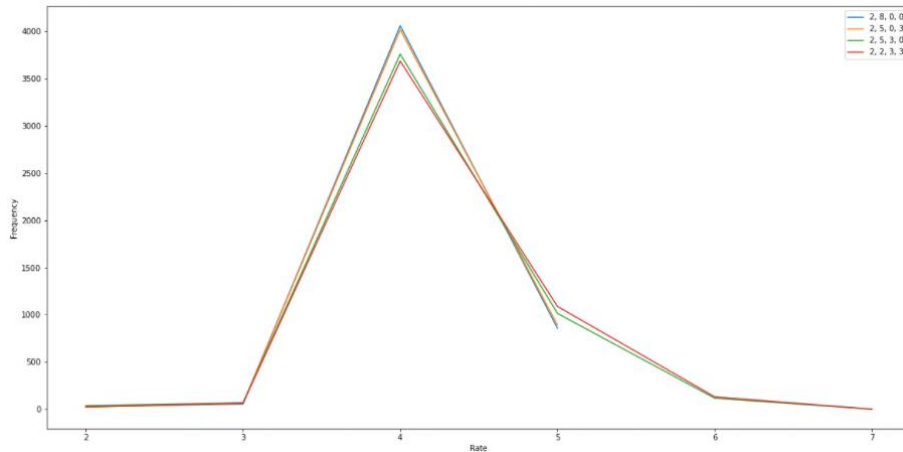
메모 포함[최18]: 위의 데이터와는 달리 bar chart와 plot chart를 생성 할 때 10,000개의 데이터를 다뤘다. 게임을 10,000번 시행 했을 때의 시민의 승률을 x축으로, 그 빈도수를 y축으로 하였을 때, 정규분포의 형태를 띄는 것을 확인 가능하다

2) 각각 {마피아2, 시민8}, {마피아2, 시민 5, 메딕3}, {마피아2, 시민5, 경찰3, 초록}, {마피아2, 시민 2, 경찰3, 메딕3} 일 때의 시민의 승률을 정렬해 나타냄.(x: 횟수, y: 시민 승률)



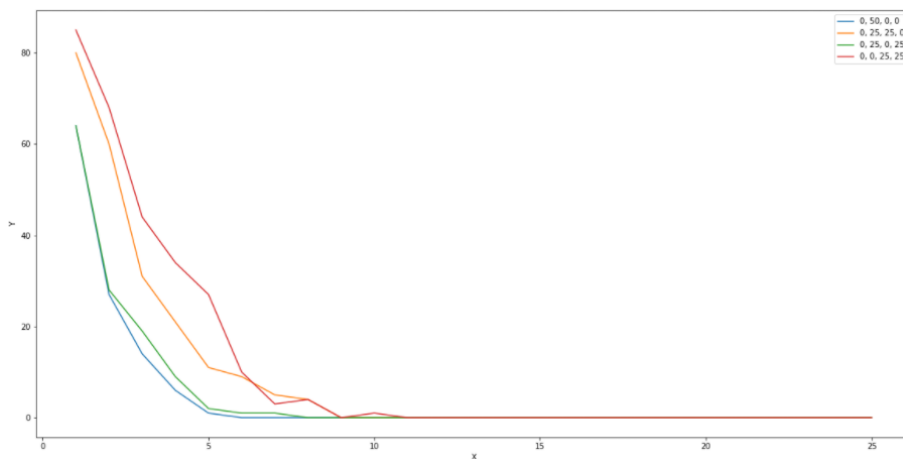
메모 포함[최19]: 프로젝트의 1,2,3,4번의 상황에 대해 시각적으로 비교해 나타낸 그래프이다. 시민의 승률에는 메딕보다 경찰이 더 큰 영향을 주는 것을 확인 가능하다.

3) 위의 네 가지 상황에 대해 게임 종료일을 빈도수로 나타냄(x: 게임 종료일, y:빈도수)

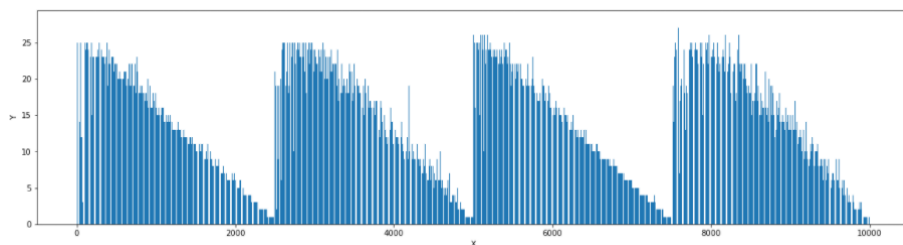


메모 포함[최20]: 위 4가지 상황의 게임 종료일을 빈도수로 나타냈다. 앞서 확인 한 것 처럼 메딕이 존재하는 경우에만 그래프가 더 높게 분포하는 것을 확인 가능하다.(메딕이 게임 진행일을 늘려줌)

4) 전체 게임인원 50명, 경찰과 메딕의 비율을 조정 (x:마피아 수, y:시민의 승률)



5) 위 상황에 대한 게임 종료일 나열(상황1,2,3,4에 대해 2,500단위 구분)



메모 포함[최21]: 2,500을 기준으로 2번째, 4번째 데이터가 경찰이 존재하는 경우이다. 경찰은 마피아를 검거함으로써 게임을 끝낼 수 있기 때문에 경찰이 존재할 경우의 편차가 비교적 큰 것을 확인 가능하다.

마피아, 시민 존재 : 0-2,500
경찰 존재 : 2,500-5,000
메딕 존재 : 5,000-7,500
모두 존재 : 7,500-10,000

4-3. 분석

1 : 마피아, 시민, 경찰이 존재 할 때

- 경찰의 수와 검거율에 따라 시민의 우승 확률에 영향을 줌
- 검거율이 높으면 확정적으로 마피아를 잡을 수 있기 때문에 시민의 우승확률에 큰 영향을 줌
- 마피아를 체포하고 시민의 동의를 받는다는 랜덤확률을 두 가지나 추가했기 때문에 확률의 편차가 크다

2 : 마피아, 시민, 메딕이 존재 할 때

- 메딕의 수와 적중률에 따라 게임 진행시간에 영향을 줌
- 밤에 치료할 확률이 높을수록 게임을 오래 끌고 가게 됨. 최대 2배까지 진행시간을 늘려준다
- 메딕이 본인만 치료한다는 가정에도 낮투표에는 똑같은 방법으로 죽을 수 있음
- 모든 상황이 랜덤일 때, 경찰이 존재 할 때보다 시민의 우승확률에 현저히 적은 기여를 한다

3 : 모든 직업이 다 존재 할 때

- 세 가지 상황 중 동일 조건에서 시민의 우승 확률이 제일 높음
- 메딕의 역할이 클 수록 게임이 오래 진행됨
- 경찰의 역할이 클 수록 시민의 우승 확률이 상승
- 메딕의 역할도 승률에 영향을 준다