1. Description

▶ ***k-nearest-neighbor,KNN*** 은 새로운 데이터가 주어졌을 때, 기존 데이터 가운데 가장 가까운 k 개의 이웃의 정보로 새로운 데이터를 예측하는 방법론이다. 새로운 데이터가 들어왔을 때 기존 데이터 사이의 거리를 재서 이웃들을 뽑는 절차로, 모델을 별도로 구축하지 않는다고 하여 lazy model 이라고도 한다. 하이퍼 파라미터는 탐색할 이웃 수(k), 거리 측정 방법 두 가지가 있다. K 가 작을 경우 데이터의 지역적 특성을 지나치게 반영하고, 반대로 매우 클 경우 모델이 과하게 정규화되는 경향이 있다.

▶ k-nearest-neighbor 추정에 있어서 모델이 안정하면 biased 하고 덜 안정하더라도 조금 biased 한 모델 추정 방법이 있다. 큰 학습데이터를 가지고 있을 때, k-nearest-neighbor averaging 방법으로 적절한 최적의 조건부 기댓값을 찾는다. 이때, 고차원의 데이터에서는 문제가 생기는데 이를 ***"The curse of dimensionality, 차원의 저주"*** 라고 부른다.

▶ 데이터의 차원이 증가할수록 해당 공간의 크기, 부피가 기하급수적으로 증가하기 때문에 동일한 개수의 데이터의 밀도는 차원이 증가할수록 희박(sparse)해진다. 이번 과제에서는 이를 보여주는 여러가지 표현 중 하나를 다루려고 한다. 원점이 중심인 p-dimensional unit ball 에서 균등하게 분포 되어있는 N 개의 데이터에 대해 생각해보자.

1) 원점과의 거리에 대한 N 개 값들의 median 의 식을 살펴본다.
2) 데이터 수(n), 차원의 수(p)를 input 으로 넣었을 때 위의 조건을 만족시키는 N 개의 데이터가 output 인 함수를 만들어보자.
3) 만들어진 함수로 거리의 최솟값을 구하고, simulation 을 돌려 최솟값의 median 을 구해 1)의 식과 비교해보자.

2. Implement

1) Derive the equation d(*p*,*N*) = $\left(1-\left(\frac{1}{2}\right)^{1/N}\right)^{1/p}$

전체 N 개의 데이터와 원점의 거리의 median 을 r 이라고 하자. Median 의 정의에 의해서 모든 점에 대해 원점과의 거리가 d 보다클 확률은 1/2 이다. x_i , $(i = 1, \cdots N)$를 해당점의 좌표라고 하면, 각각의 점은 독립이므로, 다음과 같이 나타낼 수 있다.

$$\frac{1}{2} = \prod_{i=1}^{N} P(\|x_i\| > r) \quad \cdots\cdots\cdots \quad ①$$

x_i는 unit ball 에서 uniform 분포를 따르기 때문에

$$P(\|x_i\| > r) = 1 - P(\|x_i\| \le r) = 1 - \frac{Kr^p}{K} = 1 - r^p \quad \cdots\cdots\cdots \quad ②$$

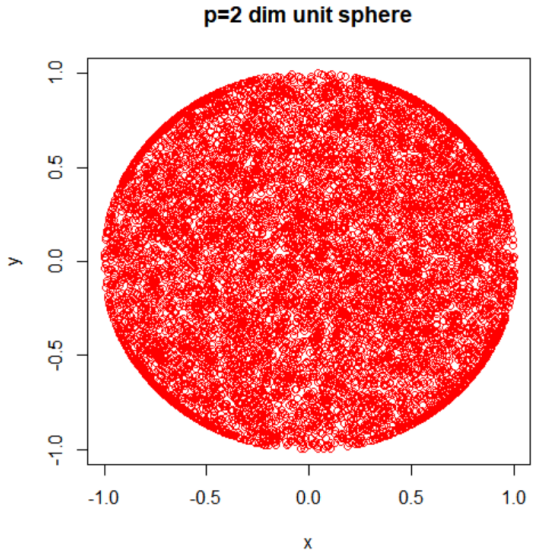
①, ②에 의해,

$$\frac{1}{2} = \prod_{i=1}^{N} P(\|x_i\| > r) = \prod_{i=1}^{N}(1 - r^p) = (1 - r^p)^N \text{ 이므로}$$

$$\text{r = d(}p,N\text{) = } \left(1-\left(\frac{1}{2}\right)^{1/N}\right)^{1/p} \text{ 이다.}$$

2) p-dim unit sphere 에서 random points 를 generate 하는 함수를 만들어보자.

먼저, 단순하게 생각해서 unif(0,1)을 따르는 p 차원의 n 개의 점을 생성했다. 각 points 와 원점과의 거리를 구한 후, 거리가 1 이상이면 단위벡터로 되게 만들어주었다.



위 사진은 만들어진 함수를 이용하여 p=2 차원에 대한 관측치 n = 10000 개를 생성하였고, 그에 따른 scatter plot 의 그림이다. 그래프를 보면 가장자리에 점이 몰려있는 것을 볼 수 있다. 전체 10000 개 중 원점과의 거리가 1 이상인 점이 2062 개로 약 20%를 차지하므로 원 내부의 점이 균등하게 생성되지 않았다.

다음으로, x_i는 iid 한 unit sphere(p)의 변수라고 하자. $\|x_i\|$ 원점과의 거리를 나타내고, $x_i/\|x_i\|$는 방향벡터로, 방향을 의미한다. 따라서, 원점과의 거리 즉 반지름과 방향을 랜덤하게 뽑아 점을 생성해보았다.

- $d_i = d(x_i, p)$라 하면 p($d_i < r$) = r^p이므로, p($d_i < r$)(u 라 하자)를 unif(0, 1)로 가정하면 반지름을 $u^{1/p}$로 나타낼 수 있다.
- 논문 'Choosing A Point From The Surface Of A Sphere'에 나와있는 방법에서

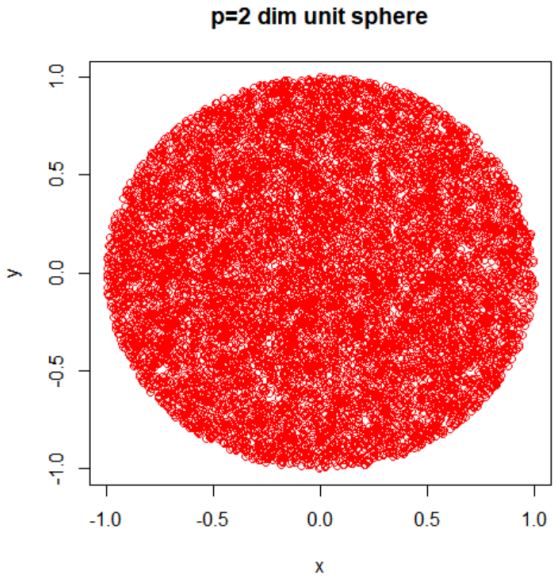
  X_1, X_2, X_3, independent 한 standard normal variates 를 생성하고

  $S = {X_1}^p + {X_2}^p + {X_3}^p$ 라고 할 때, 다음과 같이 방향벡터를 정의하였다.

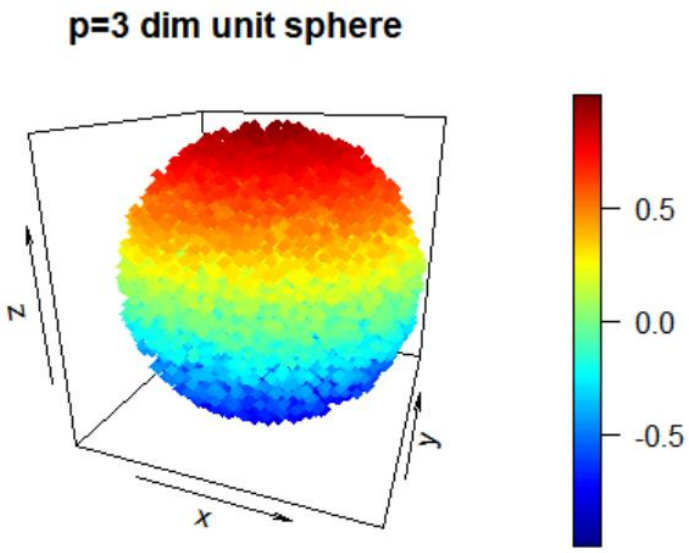
  $$(X_1/s^{1/p},\ X_2/s^{1/p},\ X_3/s^{1/p})$$

  이와 같은 방법으로, multivariate normal 한 점을 생성하여 Euclidean norm 으로 나누어 주었다.

p=2 차원에 대한 관측치 n = 10000 개를 생성하여 그에 따른 scatter plot 을 그려보았다. 그래프를 보면 첫 방법의 분포와 다르게 원 모든 점에서 균일하게 점이 생성된 것을 볼 수 있다.



역시, p=3 차원에서도 10000 개의 데이터를 이용한 그래프를 보면, 표면에서 점이 과도하게 생성되지 않았고 구의 모양이 나오는 것을 보고, 함수가 맞게 생성되었다고 생각하였다.



3) (2)에서 만든 함수로 simulation 한 값과 (1)에서의 값이 근사하는지 알아보자.

① 차원 p 를 변화시켜 관측치 N=100 로 1000 번의 simulation

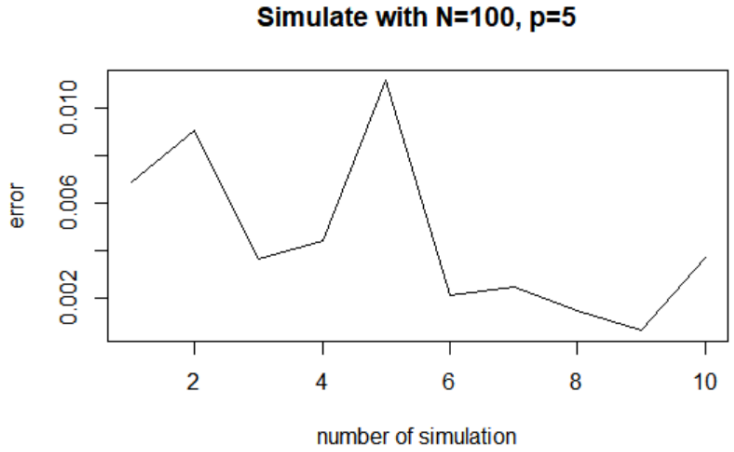
p	(1)의 값	(2)의 값	error
2	0.1904	0.1899	0.0005
3	0.2883	0.2862	0.0021
4	0.3697	0.3664	0.0033
5	0.4364	0.4361	0.0003
⋮	⋮	⋮	⋮
26	0.8258	0.8254	0.0005
27	0.8317	0.8326	-0.0009
28	0.8372	0.8350	0.0022
29	0.8424	0.8423	0.0001
30	0.8472	0.8487	-0.0015

차원 p 를 2 부터 30 까지 1 씩 증가하여 simulation 을 해보았다. (1)의 식에서의 값과 시뮬레이션 결과의 차의 평균은 약 0.0015 로 근사한다고 할 수 있다.

② p=5 차원일 때, simulation 수를 변화

1) 관측치 N=100 개 일 때

sim.n	(1)의 값	(2)의 값	error
100	0.3697	0.3628	0.0069
200	0.3697	0.3787	-0.0090
300	0.3697	0.3661	0.0036
400	0.3697	0.3653	0.0044
500	0.3697	0.3586	0.0111
600	0.3697	0.3676	0.0021
700	0.3697	0.3672	0.0025
800	0.3697	0.3683	0.0014
900	0.3697	0.3703	-0.0006
1000	0.3697	0.3734	-0.0037

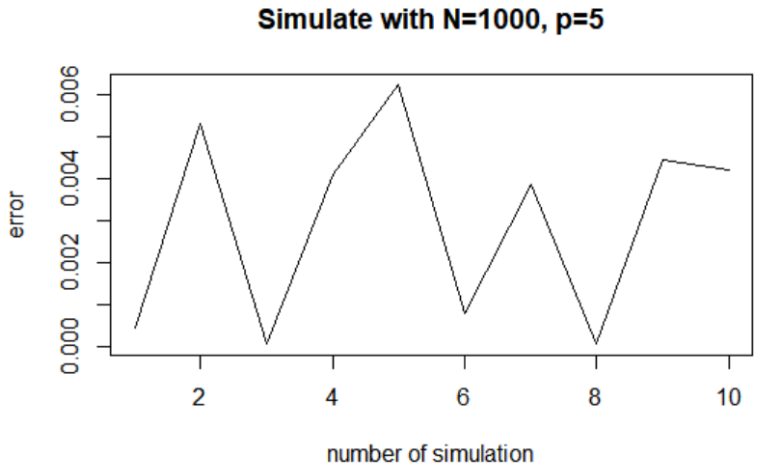


5 차원의 데이터 100 개를 이용하여 simulation 수를 100 부터 1000 까지 100 씩 증가하였다. 10 개의 error 에 대한 절대값의 평균은 0.0045 로 근사한다고 말할 수 있다.

2) 관측치 N=1000 개 일 때

sim	(1)의 값	(2)의 값	error
100	0.2334	0.2330	0.0004
200	0.2334	0.2281	0.0053
300	0.2334	0.2335	-0.0001
400	0.2334	0.2375	-0.0041

500	0.2334	0.2397	-0.0062
600	0.2334	0.2326	0.0008
700	0.2334	0.2296	0.0039
800	0.2334	0.2335	-0.0001
900	0.2334	0.2379	-0.0045
1000	0.2334	0.2292	0.0042



5 차원의 데이터 1000 개를 이용하여 simulation 수를 100 부터 1000 까지 100 씩 증가하였다. 10 개의 error 에 대한 절대값의 평균은 0.0029 로 근사한다고 말할 수 있다.

3. Discussion

▶ $d(p,N) = \left(1-\left(\frac{1}{2}\right)^{1/N}\right)^{1/p}$ 에서 N = 500, p = 10 일 때, $d(p,N) \approx 0.52$ 로 절반 이상의 점들이 원점과의 거리가 0.5 이상 떨어져 있다. 이는 구 표면에 가까운 점들이 많다는 것을 의미한다. 즉 차원이 높을수록 평균 데이터 거리보다 임의의 두 점 간 거리가 멀어질 확률이 커진다는 의미고, k-nearest-neighbor 추정에서 가깝다는 의미가 없어지므로 차원축소가 필요하다. 이것이 "차원의 저주"이다.

▶ 원점이 중심인 p-dimensional unit ball 에서 균등하게 분포 되어있는 N 개의 데이터를 생성하는 함수를 만들었고 그 함수를 이용하여 simulation 을 해보았다. Simulation 수를 증가시키면 당연히 error 도 감소할 것이라고 생각했지만, 많이 반복시키지 않아도 오차가 작아 감소하는 경향을 나타내지는 않았다. 차원과 시뮬레이션 수를 같게 하였을 때는 generate 된 points 가 많을수록 error 가 작았다.