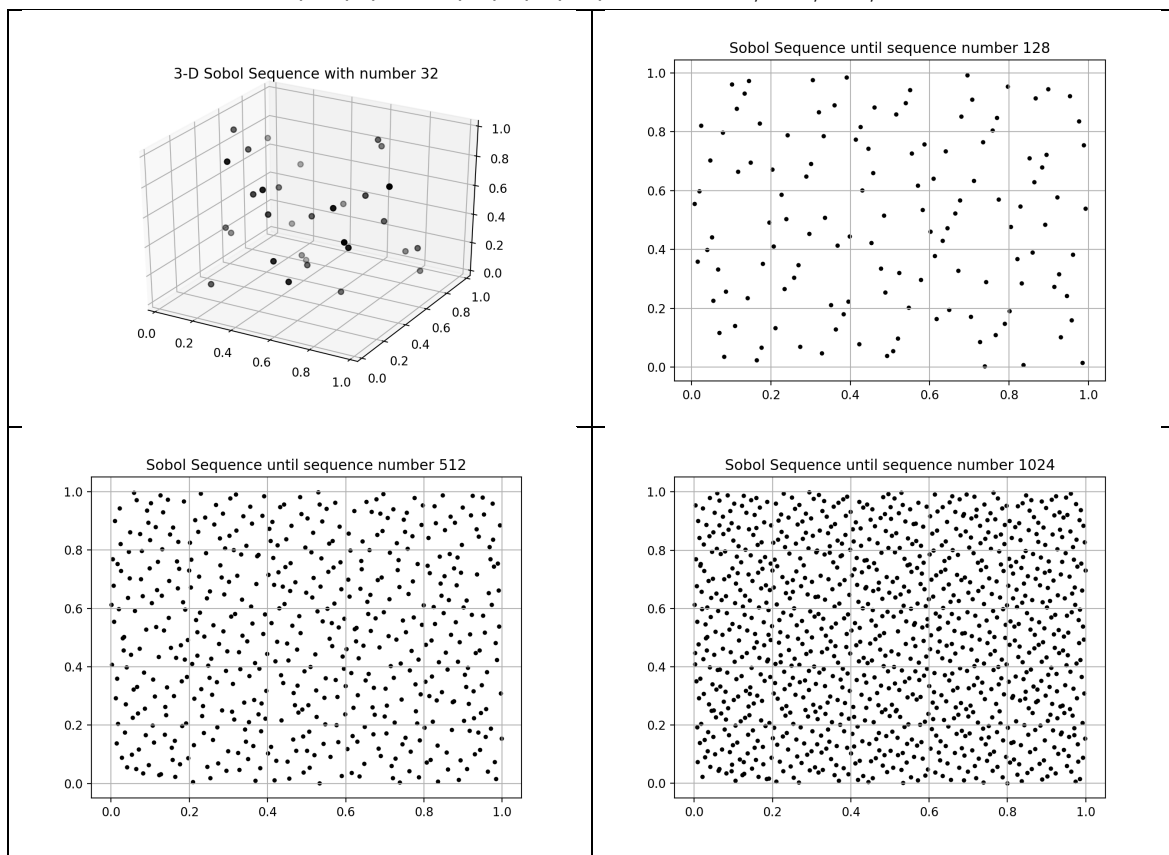


다음과 같이 Sobol sequence 를 생성하여 산점도를 그려보았다.

1. 3 차원 32 개 샘플
2. 2 차원 1024 개 샘플
 - A. 1 ~ 128 번 plot
 - B. 1 ~ 512 번 plot
 - C. 1 ~ 1024 번 plot

<결과: 좌측 상단부터 우측 하단 순으로 1, 2-A, 2-B, 2-C>



산점도를 보면, 2 차원과 3 차원 모두 균등분포를 이루는 것을 확인할 수 있다.

<3 차원 샘플 결과>

| No. | X | Y | Z |
|-----|---------|---------|---------|
| 1 | 0.5 | 0.5 | 0.5 |
| 2 | 0.25 | 0.75 | 0.25 |
| 3 | 0.75 | 0.25 | 0.75 |
| 4 | 0.375 | 0.625 | 0.125 |
| 5 | 0.875 | 0.125 | 0.625 |
| 6 | 0.125 | 0.375 | 0.375 |
| 7 | 0.625 | 0.875 | 0.875 |
| 8 | 0.3125 | 0.3125 | 0.6875 |
| 9 | 0.8125 | 0.8125 | 0.1875 |
| 10 | 0.0625 | 0.5625 | 0.9375 |
| 11 | 0.5625 | 0.0625 | 0.4375 |
| 12 | 0.1875 | 0.9375 | 0.5625 |
| 13 | 0.6875 | 0.4375 | 0.0625 |
| 14 | 0.4375 | 0.1875 | 0.8125 |
| 15 | 0.9375 | 0.6875 | 0.3125 |
| 16 | 0.46875 | 0.84375 | 0.40625 |
| 17 | 0.96875 | 0.34375 | 0.90625 |
| 18 | 0.21875 | 0.09375 | 0.15625 |
| 19 | 0.71875 | 0.59375 | 0.65625 |
| 20 | 0.09375 | 0.46875 | 0.28125 |
| 21 | 0.59375 | 0.96875 | 0.78125 |
| 22 | 0.34375 | 0.71875 | 0.03125 |
| 23 | 0.84375 | 0.21875 | 0.53125 |
| 24 | 0.15625 | 0.53125 | 0.84375 |
| 25 | 0.65625 | 0.03125 | 0.34375 |
| 26 | 0.40625 | 0.28125 | 0.59375 |
| 27 | 0.90625 | 0.78125 | 0.09375 |
| 28 | 0.28125 | 0.15625 | 0.96875 |
| 29 | 0.78125 | 0.65625 | 0.46875 |
| 30 | 0.03125 | 0.90625 | 0.71875 |
| 31 | 0.53125 | 0.40625 | 0.21875 |
| 32 | 0.26562 | 0.60938 | 0.57812 |

<Code>¹**Sobol's sequence 출력 Code(C++ main function)**

```

#include <iostream>
#include <iomanip>
#include "nr.h"

int main(void) {
    int n1=(-1);
    Vec_DP x1(3);
    Vec_DP x2(2);
    FILE* fp;

    fp = fopen("result1.csv", "w+");          // 출력할 엑셀 파일
    NR::sobseq(n1,x1);
    NR::sobseq(n1,x2);
    std::cout << fixed << setprecision(5);    // 소수점 5 번째 자리까지 출력
    std::cout << "3-D sobol sequence" << std::endl;
    for (int i = 0; i < 32; ++i) {
        NR::sobseq(3, x1);
        std::cout << setw(1) << i+1 << "\t" << x1[0] << "  " << x1[1];
        std::cout << "  " << x1[2] << std::endl;
        // 엑셀 파일 출력
        fprintf(fp, "%d, %.5f, %.5f, %.5f \n", i+1, x1[0], x1[1], x1[2]);
    }
    fclose(fp);
    fp = fopen("result2.csv", "w+");
    std::cout << "2-D sobol sequence" << std::endl;
    for (int i = 0; i < 1024; ++i) {
        NR::sobseq(2, x2);
        std::cout << i+1 << "\t" << x2[0] << "  " << x2[1] << std::endl;
        fprintf(fp, "%d, %.5f, %.5f \n", i+1, x2[0], x2[1]);
    }
    fclose(fp);
    return 0;
}

```

¹ Sequence generation 은 C++, plotting 은 Python 을 이용하였음

Plotting Code(Python)

```
import pandas as pd
import matplotlib.pyplot as plt

result1 = pd.read_csv('result1.csv', header = None, index_col = 0)
result2 = pd.read_csv('result2.csv', header = None, index_col = 0)

fig1 = plt.figure()
ax1 = fig1.add_subplot(111, projection = '3d')
ax1.scatter(result1[1], result1[2], result1[3], c = 'k', marker = 'o')
plt.title("3-D Sobol Sequence with number 32")

result = []
scatternumber = [128, 512, 1024]
for num in scatternumber:
    fig2 = plt.figure()
    ax2 = fig2.add_subplot(111)
    ax2.scatter(result2[1][:num], result2[2][:num], s = 7, c = 'k')
    plt.grid()
    plt.title("Sobol Sequence until sequence number %d" %num)
    plt.savefig("sobol_2d_%d" %num)
    result.append(fig2)
```