応用統計学 レポート課題 1（11/17 提出）

問題 1

線分の本数 $k = 2, 5, 10, 20, 50, 100, 200$ として，交差していない線分 Yの期待値を繰り返し回数 $M = 10^3$としてモンテカルロ法により推定した結果は以下に示す表 1，図 2 のようになった．

定性的には線分の数が多いほど交差しない線分の数は減少すると考えられるが，結果からkが 10 以上の範囲では単調に減少することが分かり，今回のkの範囲では先の考察を実証できている．また，推定誤差についてもkが 10 以上の範囲では単調に減少することが分かった．これは線分の本数が多いほど系が巨視的に同じような振る舞いを示すと理解できる．補足に使用したコードを示した．

表 1. 線分の本数 k に対するYの推定値$\bar{Y}$と，推定誤差$\widehat{\sigma_Y}$

k	2	5	10	20	50	100	200
$\bar{Y}$	1.55	2.04	2.14	1.79	1.41	1.13	0.89
$\widehat{\sigma_Y}$	0.03	0.05	0.05	0.04	0.04	0.03	0.03

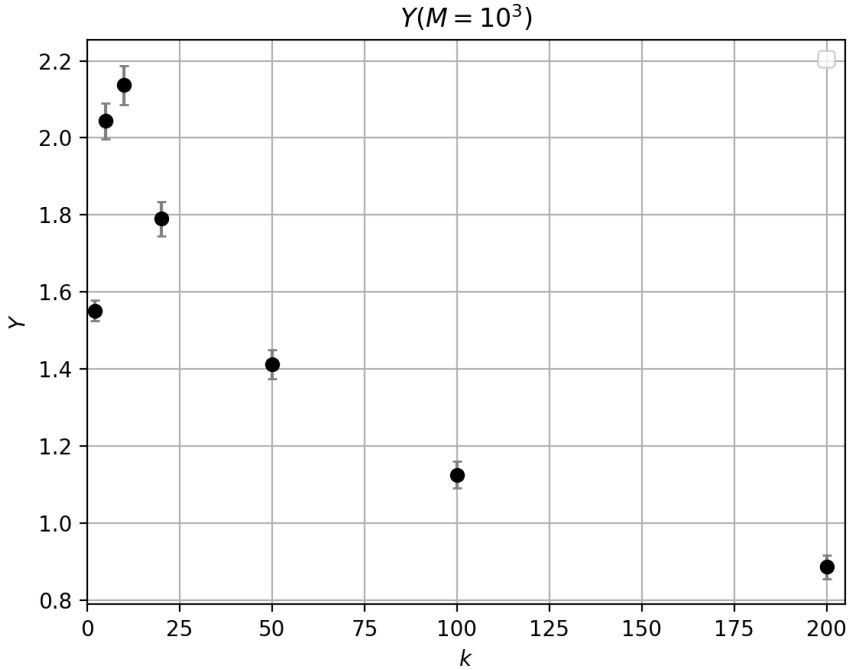

図 2. Yの期待値（繰り返し回数$M = 10^3$）

問題 2

MLB(Major League Baseball)の各チームの勝率の順位に対して，打撃，投球，守備についての指標から一つずつ選んで示したものが表 3 である（データは Baseball Reference[1]による）．選んだ指標は，OPS（打撃）：出塁率＋長打率，WHIP（投球）：1 投球回当りの走者数，守備率 Fld%（守備）：守備機会のうち失策しなかった割合）である（このうち WHIP のみ数字が小さいほど良い指標であることに注意する）．これらの指標を用いた理由は，各指標が MLB の選手の評価において重要視されているからである．他の指標については，例えば防御率のような指標は失点（自責点）を表しており，勝率との相関が強すぎると考え用いなかった．

表 3. MLB 各チームの勝率と打撃，投球，守備についての指標

	Tm	W-L%	OPS	WHIP	Fld%
1	San Francisco Giants	0.66	0.769	1.148	0.986
2	Los Angeles Dodgers	0.654	0.759	1.097	0.985
3	Tampa Bay Rays	0.617	0.75	1.168	0.986
4	Houston Astros	0.586	0.783	1.232	0.988
5	Milwaukee Brewers	0.586	0.713	1.179	0.984
6	Chicago White Sox	0.574	0.758	1.204	0.982
7	Boston Red Sox	0.568	0.777	1.378	0.981
8	New York Yankees	0.568	0.729	1.209	0.983
9	Toronto Blue Jays	0.562	0.797	1.231	0.984
10	Seattle Mariners	0.556	0.688	1.278	0.986
11	St. Louis Cardinals	0.556	0.725	1.3	0.986
12	Atlanta Braves	0.547	0.754	1.243	0.988
13	Oakland Athletics	0.531	0.723	1.257	0.987
14	Cincinnati Reds	0.512	0.759	1.357	0.984
15	Philadelphia Phillies	0.506	0.726	1.29	0.984
16	Cleveland Indians	0.494	0.71	1.281	0.985
17	San Diego Padres	0.488	0.722	1.254	0.986
18	Detroit Tigers	0.475	0.707	1.367	0.986
19	Los Angeles Angels	0.475	0.717	1.382	0.985
20	New York Mets	0.475	0.705	1.23	0.983
21	Colorado Rockies	0.46	0.731	1.386	0.987
22	Kansas City Royals	0.457	0.702	1.387	0.985
23	Minnesota Twins	0.451	0.738	1.322	0.982
24	Chicago Cubs	0.438	0.719	1.403	0.985
25	Miami Marlins	0.414	0.671	1.28	0.979
26	Washington Nationals	0.401	0.754	1.371	0.983
27	Pittsburgh Pirates	0.377	0.673	1.437	0.988
28	Texas Rangers	0.37	0.67	1.344	0.986
29	Arizona Diamondbacks	0.321	0.692	1.436	0.983
30	Baltimore Orioles	0.321	0.705	1.484	0.987

このデータを標準化し，バイプロットを描くと図 4 のようになった．ここで，同じ勝率のチームには表の出現番号順に番号が振られていることに注意する．これを見ると，順位が高いチームほど左側に位置する傾向が見られ，主に第一主成分が順位を左右していることが分かる．バイプロットを見ると，9（Toronto Blue Jays）が上位のチームと似たような位置にあるが，これは Blue Jays と同じ地区に属する 5 チームのうち 4 チームが勝率の上位 9 位以内に入る強豪チームであり，これらのチームと対戦が多かったことが，勝率が低いことの一因であると考えられる．

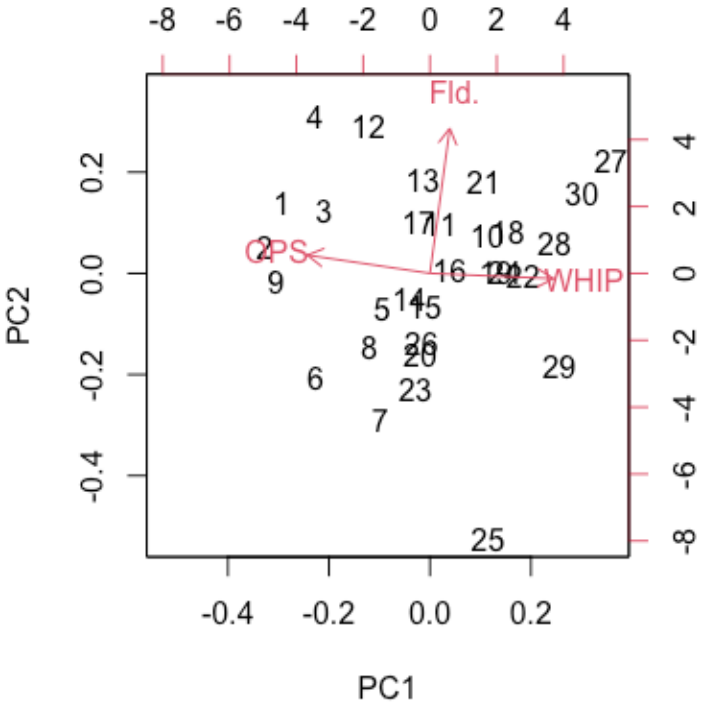

図 4. MLB のデータに対するバイプロット（変量空間）

また，回転行列と寄与率を求めるとそれぞれ表 5，表 6 のようになった．回転行列を見ると，第一主成分は OPS と WHIP に強く相関していることが分かる．第一主成分の行列要素から，OPS と WHIP の大きさがほとんど同じであり，いずれも同程度に勝率に影響すると考えられる．また，第二主成分は守備率と強く相関していることが分かり，バイプロットからも守備率はあまり勝率に影響しないことが分かる．これは守備率のデータから分かるように MLB 全体の守備のレベルが高く，投球に比べ守備が失点に影響しにくいからだと考えられる．また表 6 をみると，第一主成分の寄与率がそれほど大きくないことが分かる．この一因としては，リーグや地区により所属するチームの強さが違っており，個々のチームの強さのみでは勝率が決まらないことが考えられる（しかし，MLB では他地区のチームや他のリーグとの交流戦を多く実施しており，勝率と各指標の関係を調べることには意味があると考えられる．実際にバイプロットからは第一主成分が順位を左右している傾向が見られる）．

表 5. 回転行列

回転行列	PC1	PC2	PC3
OPS	-0.7000622	0.12444132	-0.7031552
WHIP	0.7050513	-0.0356528	-0.7082596
Fld.	0.1132062	0.99158622	0.06277836

表 6. 寄与率

PC1	PC1	PC2	PC3
寄与率	0.4767	0.332	0.1913
累積寄与率	0.4767	0.8087	1

データの出典

[1] Baseball Reference, “2021 Major League Baseball Team Statistics”, https://www.baseball-reference.com/leagues/majors/2021.shtml, (参照 2021-11-17).

問題 3

授業の形式についてですが，板書の手書きのため内容が頭に入ってきやすく，とても良いと思います．板書も 2 枚表示されているため，少し遅れても安心して授業を受けることができています．

補足

問題 1 で用いたコードを以下に示す．

```
import numpy as np
import random
import matplotlib.pyplot as plt

def line(p,p1,p2): #p1, p2 を通る直線
  x,y = p[0],p[1]
  x1,y1 = p1[0],p1[1]
  x2,y2 = p2[0],p2[1]
  return (y2-y1)*x-(x2-x1)*y-x1*y2+x2*y1

def y(k):
  n = 1000
  cexps = []
  for i in range(n):
    edges = [] #端点を保存
    count = np.zeros(k) #交差している線分を 1，非交差を 0
    for j in range(k):
      edge_j = [[random.random(),random.random()],[random.random(),random.random()]] #端点を生成
```

```
            p1_j,p2_j = edge_j[0],edge_j[1]
            for l in range(j):
                edge_l = edges[l]
                p1_l,p2_l = edge_l[0],edge_l[1]
                if line(p1_j,p1_l,p2_l)*line(p2_j,p1_l,p2_l)<=0 and line(p1_l,p1_j,p2_j)*line(p2_l,p1_j,p2_j)<=0: #交差判定
                    if line(p1_j,p1_l,p2_l)*line(p2_j,p1_l,p2_l)==0 and line(p1_l,p1_j,p2_j)*line(p2_l,p1_j,p2_j)==0:
                        if max(p1_j[0],p2_j[0])<min(p1_l[0],p2_l[0]) or max(p1_l[0],p2_l[0])<min(p1_j[0],p2_j[0]):
                            continue
                        else:
                            count[j]=1
                            count[l]=1
                    else:
                        count[j]=1
                        count[l]=1
                else:
                    continue
            edges.append(edge_j) #端点を追加
        cexps.append(k-np.sum(count)) #非交差の線分の数
    return [np.average(cexps),np.sqrt(np.var(cexps)/n)]

ydata = [y(2), y(5), y(10), y(20), y(50), y(100),y(200)]
yval = [ydata[k][0] for k in range(len(ydata))]
yerr = [ydata[k][1] for k in range(len(ydata))]
x = [2,5,10,20,50,100,200]

print(yval,yerr) #Y についての統計量を表示

fig,ax = plt.subplots() #Y(k)のグラフ
ax.errorbar(x,yval,yerr=yerr,capsize=2,fmt="o",color="black",ecolor="grey")
ax.set_xlim(0,205)
ax.set_xlabel("$k$")
ax.set_ylabel("$Y$")
ax.grid()
ax.legend()
title = "$Y (M={10^3})$"
plt.title(title)
plt.show()
```