# 機器學習與實作 需求+分析

0652016許銘哲

0652075黃子衡

#### 需求及目的

- 接到95%的球
  - 1. Rule Base為主
  - 2. 以Rule Base產生記錄檔輔助機器學習(P1為SVM, P2為KNN)
  - 3. 能化解奇怪的球形
- 接到球的位置是在板子中央正負5處
  - 1. 須提前預知球的位置(導入馬可夫鏈輔助)
- 攻擊:能改變球的速度與方向(切球)
  - 1. 接到球的瞬間左右搖晃
  - 2. 攻擊方式待試

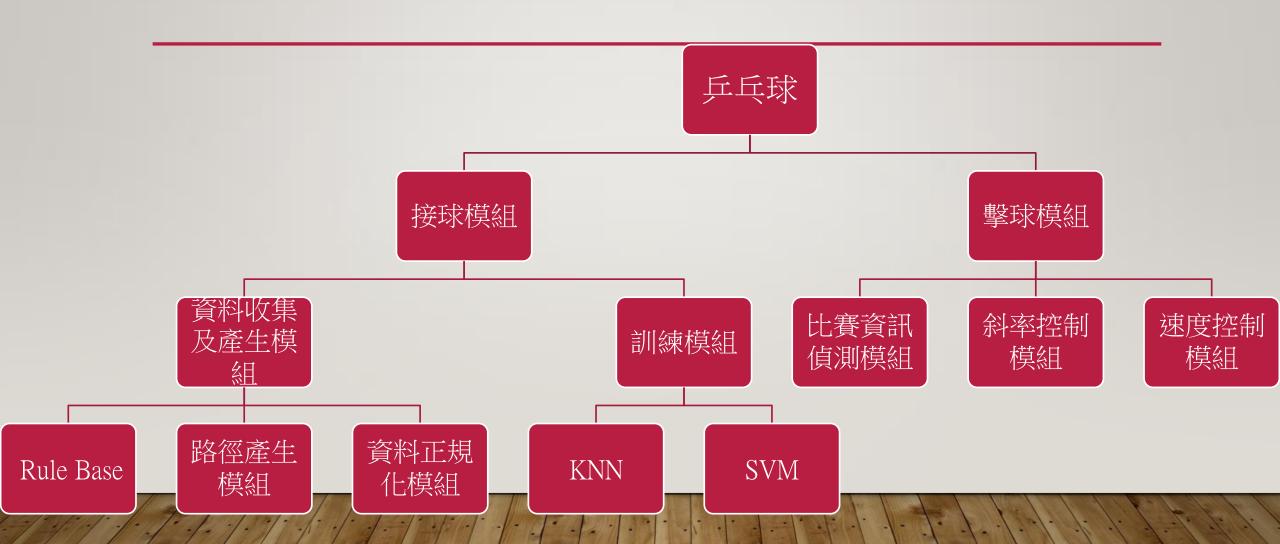
# 分析接到95%的球

- 資料分析(產生大量及有效的樣本)
  - 1. Rule Base產生一部分樣本
    - ① 以亂數方式決定發球的位置及方向
  - 2. 產生球的路徑(可視化)
    - ① 擷取接球位置及碰撞點(求斜率)
    - ② 接到球時標計碰撞點並推導出其方程式或遊戲內路徑
  - 3. 更新非已知資料
    - ① 用人工找出路徑資料的空缺點並修正

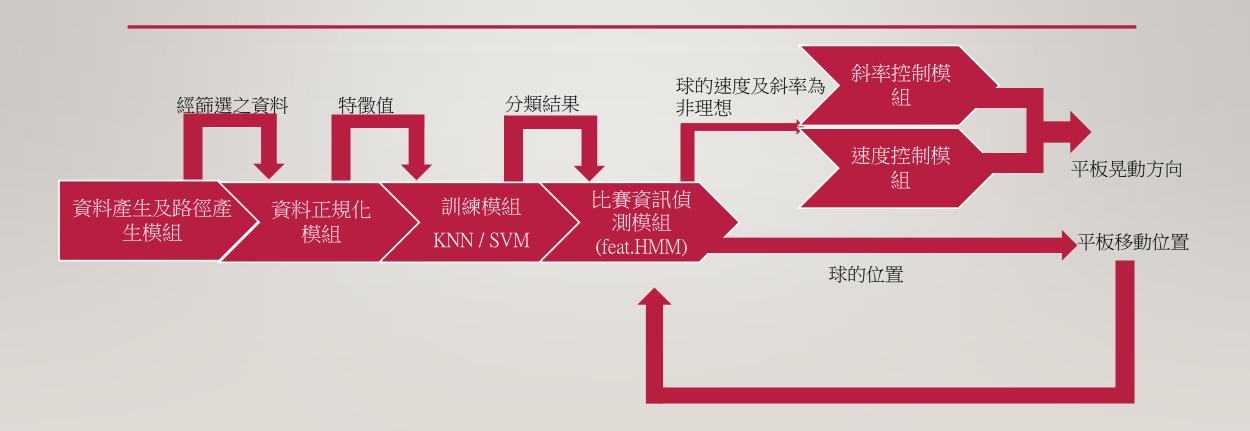
## 分析接到95%的球

- 接球模式
  - 1. 將過快的球(球速為12)減速
    - ① 逆著球的方向切球
  - 2. 將角度過大的球(斜率大於50度或斜率小於40度)修正至50度~40度之間
    - ① 斜率過大時順著球的方向切球
    - ② 斜率過小時逆著球的方向切球

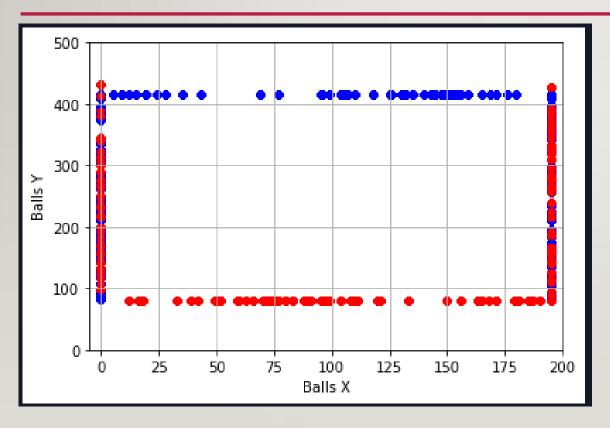
# 分析簡圖



# 架構圖

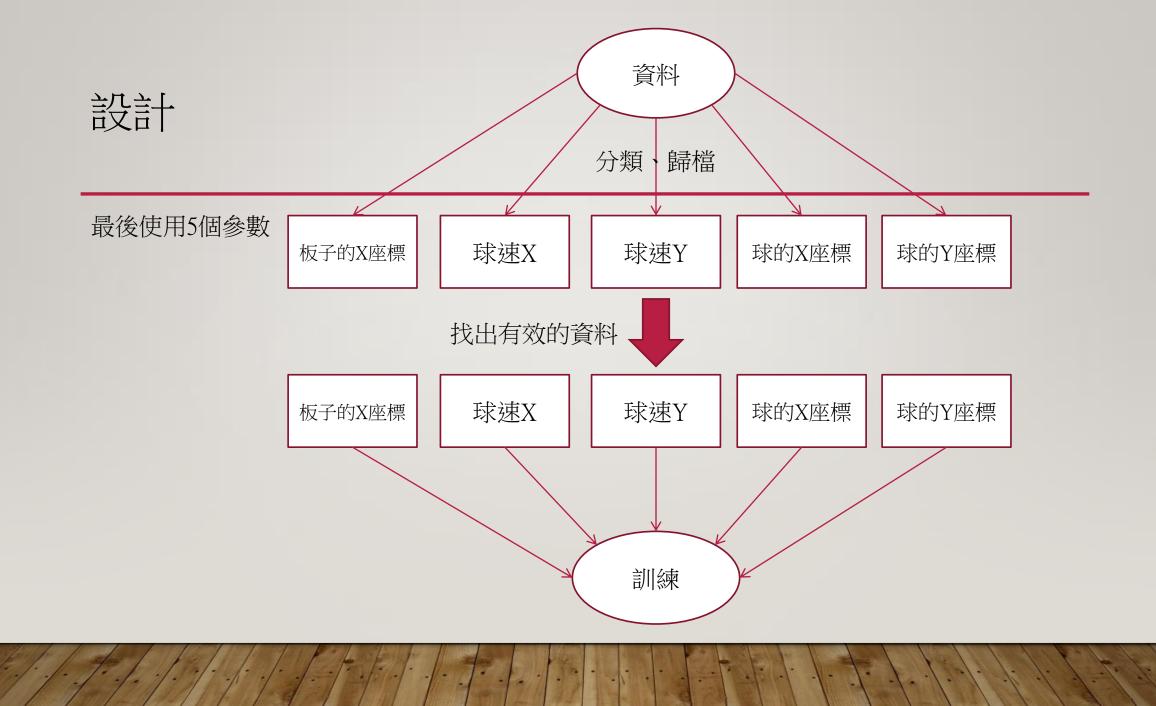


# 資料可視化



只取球的座標在Y在415和80,X在0和195, 並將他們顯示出來,是否可成為參考資料。

球向下為紅色,球向上為藍色。



# 程式(訓練KNN)

```
📙 KNN_2Player.py 🗵 📙 knn_1p.py 🗷 🔒 knn_2P.py 🗷
📙 KNN_2Player.py 🗵 📙 knn_1p.py 🛚
                                                                                                                             17 Efor f in files:
                            ##將路徑底下的檔名與路徑結合
17 Efor f in files:
                                                                                                                                       allpath = join(path, f)
          allpath = join(path, f)
                                                                                                                             19
                                                                                                                                       if isfile(allpath):
          if isfile(allpath):
                                                                                                                             20
                                                                                                                                           with open(allpath , "rb") as fl:
             with open(allpath , "rb") as fl:
                                                                                                                             21
                                                                                                                                               data list1 = pickle.load(f1)
                 data list1 = pickle.load(f1)
                                                                                                                             22
                                                                                                                                               for ml name in data listl.keys():
                 for ml name in data listl.keys():
                                                                                                                             23
                                                                                                                                                   if ml name == "record format version";
                     if ml name == "record format version":
                                                                                                                             24
                                                                                                                                                       continue
 24
                        continue
                                                                                                                             25
                                                                                                                                           target record = data listl[ml name]
 25
             target record = data listl[ml name]
                                                                                                                                           for n in range(0,len(target record["scene info"])):
                                                                                                                             26
             for n in range(0,len(target record["scene info"])):
                                                                                                                             27
                                                                                                                                               Frame.append(target record["scene info"][n]["frame"])
                 Frame.append(target record["scene info"][n]["frame"])
                                                                                                                             28
                                                                                                                                               PlatformPosition2P.append(target_record["scene_info"][n]["platform_2P"])
                 PlatformPosition1P.append(target record["scene info"][n]["platform 1P"])
                                                                                                                             29
                 BallPosition.append(target record["scene info"][n]["ball"])
                                                                                                                                               BallPosition.append(target record["scene info"][n]["ball"])
29
                                                                                                                             30
30
                 BallSpeed.append(target_record["scene_info"][n]["ball_speed"])
                                                                                                                                               BallSpeed.append(target record["scene info"][n]["ball speed"])
31
      PlatX = np.array(PlatformPosition1P) [:,0][:,np.newaxis] #[:,0]->取所有第一陣列的第一個數值(X座標) #[:,np.newaxis]->陣列變成直的
                                                                                                                             32
                                                                                                                                   PlatX = np.array(PlatformPosition1P) [:,0][:,np.newaxis] #[:,0]->取所有第一陣列的第一個數值(X座標) #[:,np.newaxis]->陣列變成直的
                                                                                                                                   PlatX next = PlatX[1:,:] #除了第一個值以外都要
      PlatX next = PlatX[1:,:] #除了第一個值以外都要
                                                                                                                                   instrust = (PlatX next-PlatX[0:len(PlatX next)])/5 #板子位移量為5
      instrust = (PlatX_next-PlatX[0:len(PlatX_next)])/5 #板子位移量為5
                                                                                                                             35
      Ballarray = np.array(BallPosition[:-1])
                                                                                                                             36
                                                                                                                                   Ballarray = np.array(BallPosition[:-1])
                                                                                                                             37
                                                                                                                             38
      Ball Vx=np.array(BallSpeed)[:-1,0][:,np.newaxis]
                                                                                                                                   Ball_Vx=np.array(BallSpeed)[:-1,0][:,np.newaxis]
                                                                                                                             39
      Ball Vy=np.array(BallSpeed)[:-1,1][:,np.newaxis]
                                                                                                                                   Ball Vy=np.array(BallSpeed)[:-1,1][:,np.newaxis]
                                                                                                                                   x = np.hstack((Ballarray,PlatX[:-1,0][:,np.newaxis],Ball_Vx,Ball_Vy))
      x = np.hstack((Ballarray,PlatX[:-1,0][:,np.newaxis],Ball Vx,Ball Vy))
                                                                                                                             42
      np.set printoptions(threshold=np.inf)
                                                                                                                                   np.set printoptions(threshold=np.inf)
      y = instrust
                                                                                                                                  y = instrust
      y = np.array(y, dtype=int)
                                                                                                                                  y = np.array(y, dtype=int)
                                                                                                                             46
                                                                                                                             47
      np.set printoptions(threshold=np.inf)
                                                                                                                                   np.set printoptions(threshold=np.inf)
      #----- train & test data
                                                                                                                                   #----- train & test data
      from sklearn.model selection import train test split
                                                                                                                             50
                                                                                                                                   from sklearn.model selection import train test split
                                                                                                                             51
      #----- train model
                                                                                                                             52
                                                                                                                                   #----- train model
      from sklearn.neighbors import KNeighborsClassifier
                                                                                                                                   from sklearn.neighbors import KNeighborsClassifier
      from sklearn.metrics import accuracy score ,mean squared error
                                                                                                                                   from sklearn.metrics import accuracy score , mean squared error
      from sklearn.feature selection import SelectKBest, f regression, GenericUnivariateSelect
                                                                                                                                   from sklearn.feature selection import SelectKBest, f regression, GenericUnivariateSelect
                                                                                                                             56
57
                                                                                                                             57
                                                                                                                             58
      x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 1)
                                                                                                                                   x train,x test,y train,y test = train test split(x,y,test size = 0.2,random state = 1)
      knn = KNeighborsClassifier(n neighbors = 3)
                                                                                                                                   knn = KNeighborsClassifier(n neighbors = 3)
      knn.fit(x train,y train)
                                                                                                                                   knn.fit(x train,y train)
                                                                                                                             62
      yknn bef scaler = knn.predict(x test)
                                                                                                                                   yknn bef scaler = knn.predict(x test)
      r2 = accuracy_score(yknn_bef_scaler,y_test)
                                                                                                                                   r2 = accuracy score(yknn bef scaler,y test)
      mse = mean_squared_error(y_test, yknn_bef_scaler)
                                                                                                                                   mse = mean_squared_error(y_test, yknn_bef_scaler)
                                                                                                                             65
                                                                                                                             66
      filename = "C:\\Users\\aiolb\\Desktop\\MLGame-beta8.0.1\\games\\pingpong\\ml\\knn 1P 20210106.sav"
                                                                                                                             67
                                                                                                                                    filename = "C:\\Users\\aiolb\\Desktop\\MLGame-beta8.0.1\\games\\pingpong\\ml\\knn 2P 20210106.sav"
      pickle.dump(knn,open(filename, "wb"))
                                                                                                                                   pickle.dump(knn,open(filename,"wb"))
```

## 程式(訓練SVM)

```
KNN_2Player.py ☑ 🖶 knn_1p.py 区 🔡 knn_2P.py 区 🖺 svm_1p.py 区 🔡 svm_2p.py 🗵

      Image: Boundard of the control of 
 19 ⊟for f in files:
                                               ##將路徑底下的檔名與路徑結合
                                                                                                                                                                     19 ⊟for f in files:
                                                                                                                                                                                                                    ##將路徑底下的檔名與路徑結合
               allpath = join(path, f)
                                                                                                                                                                                   allpath = join(path, f)
 21 🖨
               if isfile(allpath):
                      with open(allpath , "rb") as f1:
                                                                                                                                                                    21 🖨
                                                                                                                                                                                   if isfile(allpath):
 22
                                                                                                                                                                    22
                                                                                                                                                                                          with open(allpath , "rb") as f1:
 23
                            data list1 = pickle.load(f1)
                                                                                                                                                                    23
 24
                            for ml name in data list1.keys():
                                                                                                                                                                                                data list1 = pickle.load(f1)
 25 🖨
                                                                                                                                                                    24 🖨
                                                                                                                                                                                                for ml name in data list1.keys():
                                  if ml name == "record format version":
                                                                                                                                                                                                       if ml name == "record format version":
 26
                                                                                                                                                                    25 白
                                         continue
                                                                                                                                                                                                             continue
 27
                      target record = data list1[ml name]
                                                                                                                                                                    26
                                                                                                                                                                                          target record = data list1[ml name]
 28
                      for n in range(0,len(target record["scene info"])):
                                                                                                                                                                    27
                                                                                                                                                                                          for n in range(0,len(target record["scene info"])):
 29
                            Frame.append(target record["scene info"][n]["frame"])
                                                                                                                                                                    28 🖨
                            PlatformPosition1P.append(target record["scene_info"][n]["platform_1P"])
                                                                                                                                                                                                Frame.append(target record["scene info"][n]["frame"])
                                                                                                                                                                    29
                                                                                                                                                                                                PlatformPosition2P.append(target record["scene info"][n]["platform 2P"])
 31
                            BallPosition.append(target record["scene info"][n]["ball"])
                            BallSpeed.append(target record["scene info"][n]["ball speed"])
                                                                                                                                                                                                BallPosition.append(target record["scene info"][n]["ball"])
                                                                                                                                                                     32
                                                                                                                                                                                                BallSpeed.append(target record["scene info"][n]["ball speed"])
 34
          from sklearn.model selection import train test split
                                                                                                                                                                     34
          from sklearn.metrics import accuracy score , mean squared error
                                                                                                                                                                             from sklearn.model selection import train test split
                                                                                                                                                                             from sklearn.metrics import accuracy score , mean squared error
         PlatX = np.array(PlatformPosition2P) [:,0][:,np.newaxis] #[:,0]->取所有第一陣列的第一個數值(X座標)
          PlatX next = PlatX[1:,:] #除了第一個值以外都要
                                                                                                                                                                             PlatX = np.array(PlatformPosition2P) [:,0][:,np.newaxis] #[:,0]->取所有第一陣列的第一個數值(X座標)
          instrust = (PlatX next-PlatX[0:len(PlatX next)])/5 #板子位移量為5
                                                                                                                                                                             PlatX next = PlatX[1:,:] #除了第一個值以外都要
 42
                                                                                                                                                                             instrust = (PlatX next-PlatX[0:len(PlatX next)])/5 #板子位移量為5
 43
         Ballarray = np.array(BallPosition[:-1])
                                                                                                                                                                    42
         Ball Vx=np.array(BallSpeed)[:-1,0][:,np.newaxis]
 44
                                                                                                                                                                             Ballarray = np.array(BallPosition[:-1])
 45
                                                                                                                                                                             Ball Vx=np.array(BallSpeed)[:-1,0][:,np.newaxis]
         Ball Vy=np.array(BallSpeed)[:-1,1][:,np.newaxis]
                                                                                                                                                                    45
         x = np.hstack((Ballarray, PlatX[:-1,0][:,np.newaxis], Ball Vx, Ball Vy))
 47
                                                                                                                                                                             Ball Vy=np.array(BallSpeed)[:-1,1][:,np.newaxis]
                           #球座標X,球座標Y
                                                                                             球速X,球速Y
 48
                                                                                                                                                                             x = np.hstack((Ballarray,PlatX[:-1,0][:,np.newaxis],Ball Vx,Ball Vy))
 49
                                                                                                                                                                     48
                                                                                                                                                                                               #球座標X,球座標Y
                                                                                                                                                                                                                                   板子X
         np.set printoptions (threshold=np.inf)
                                                                                                                                                                    49
        y = instrust
 51
                                                                                                                                                                             np.set printoptions (threshold=np.inf)
        y = np.array(y, dtype=int)
                                                                                                                                                                            v = instrust
         print(len(instrust))
                                                                                                                                                                    52 y = np.array(y, dtype=int)
 54
                                                                                                                                                                    53
                                                                                                                                                                             print(len(instrust))
          svr = SVR(gamma=0.001, C = 1, epsilon = 0.1, kernel = 'rbf')
                                                                                                                                                                    54
 56
                                                                                                                                                                             svr = SVR(gamma=0.001, C = 1, epsilon = 0.1, kernel = 'rbf')
 57
         x train,x test,y train,y test = train test split(x,y,test size = 0.2,random state = 40)
                                                                                                                                                                    56
 58
                                                                                                                                                                    57
                                                                                                                                                                             svr.fit(x train,y train)
 59
         svr.fit(x train,y train)
                                                                                                                                                                             y predict = svr.predict(x test)
         y predict = svr.predict(x test)
         filename = "C:\\Users\\aiolb\\Desktop\\MLGame-beta8.0.1\\games\\pingpong\\ml\\SVM 1P 20210106.sav"
                                                                                                                                                                             filename = "C:\\Users\\aiolb\\Desktop\\MLGame-beta8.0.1\\qames\\pingpong\\ml\\SVM 2P 20210106.sav"
         pickle.dump(svr,open(filename, "wb"))
                                                                                                                                                                             pickle.dump(svr,open(filename,"wb"))
```

## 程式(遊玩)

```
KNN 2Player.py
KNN 1Player.py
                                                                                        23
                                                                                        24
24
           global wait frame
                                                                                                  global wait frame
                                                                                        25
                                                                                                  while True:
25
           while True:
                                                                                        26
26 由
             if scene info["status"] != "GAME ALIVE":
                                                          #比出勝負
                  return "RESET" #游戲重製
                                                                                        27
27
             if not self.ball served: #如果未發球
                                                                                        28
28
29
                 self.ball served = True
                                                                                        29
                  #print("ball pos:",scene info["ball"])
31
                  if(ball served random==1):
32
                      return "SERVE TO RIGHT" #往右發球
                                                                                        33
33
34
                      return "SERVE_TO_LEFT" #往左發球
                                                                                        34
                                                                                        35
35
36
              else:
                                                                                        36
                                                                                                     else:
                                                                                        37
37
                 ball position history.append(scene info["ball"])
                                                                                        38
38
                 BallPosition=np.asarray(ball position history[-1])
39
                  PlatX = np.asarray(scene info["platform 1P"][-2])
                                                                                        39
                 Ball Vx=np.asarray(scene info["ball speed"][-2])
                                                                                        40
40
41
                  Ball Vy=np.asarray(scene info["ball speed"][-1])
                                                                                        41
42
                 data x = np.hstack((BallPosition, PlatX, Ball Vx, Ball Vy))
                                                                                        42
                                                                                        43
43
                 input data x = data x[np.newaxis, :]
                  move = model.predict(input data x)
                                                                                        44
44
                                                                                        45
45
                 print("1P=",move)
46
                                                                                        46
                                                                                        47
47
                 if (move <0):
                                                                                        48
48
                      return "MOVE LEFT"
                                                                                        49
49
                  elif(move >0):
                                                                                        50
50
                      return "MOVE RIGHT"
51
                                                                                        51
                                                                                                         else:
                  else:
                                                                                        52
52
                      return "NONE"
                                                                                        53
53
                  return "NONE"
                                                                                        54
54
                                                                                        55
55
56
                                                                                        56
                                                                                        57
57
                                                                                                 def reset(self):
          def reset(self):
                                                                                        58
58
59
                                                                                        59
             Reset the status
                                                                                        60
60
                                                                                        61
61
             global ball served random
                                                                                        62
62
             self.ball served = False
63
             ball served random=random.randrange(1,3)
                                                                                        63
                                                                                        64
64
```

```
if scene info["status"] != "GAME ALIVE":
                                           #比出勝負
    return "RESET" #遊戲重製
if not self.ball served:
                         #如果未發球
    self.ball served = True
    #print("ball pos:",scene info["ball"])
    if(ball served random==1):
       return "SERVE TO RIGHT" #往右發球
       return "SERVE TO LEFT" #往左發球
   ball position history.append(scene info["ball"])
   BallPosition=np.asarray(ball position history[-1])
   PlatX = np.asarray(scene info["platform 2P"][-2])
   Ball Vx=np.asarray(scene info["ball speed"][-2])
   Ball Vy=np.asarray(scene info["ball speed"][-1])
   data x = np.hstack((BallPosition,PlatX,Ball Vx,Ball Vy))
   input data x = data x[np.newaxis, :]
   move = model.predict(input data x)
   print("2P=",move)
    if (move <0):
        return "MOVE LEFT"
    elif(move>0):
        return "MOVE RIGHT"
       return "NONE"
    return "NONE"
Reset the status
global ball served random
self.ball served = False
ball served random=random.randrange(1,3)
```

#### 驗證

#### 初賽:

1P: 0652075, 0652016

2P: 0652068, 0652001, 0652013

比賽結果:

EASY: 0:3 2P win

第一回: 速度21 2P win

第二回: 速度21 2P win

第三回: 速度23 2P win

\_

NORMAL: 0:3 2P win

第一回: 速度21 2P win

第二回: 速度21 2P win

第三回: 速度21 2P win

\_

HARD: 3:2 1P win

第一回: 速度7 2P win

第二回: 速度8 1P win

第三回: 速度12 1P win 第四回: 速度14 2P win

第五回: 速度14 1P win

#### 複賽:

1P: C3 組員: 0652050

2P: A3 組員: 0652075 0652016

比賽結果:

EASY: 0:3 2P WIN

第一回合: 速度 22 2P WIN 第二回合: 速度 15 2P WIN 第三回合: 速度 22 2P WIN

NORMAL: 0:3 2P WIN

第一回合: 速度 12 2P WIN 第二回合: 速度 13 2P WIN 第三回合: 速度 12 2P WIN

HARD: 2:3 2P WIN

第一回合: 速度 13 1P WIN 第二回合: 速度 8 2P WIN 第三回合: 速度 11 1P WIN 第四回合: 速度 10 2P WIN 第五回合: 速度 15 2P WIN 1P: A3 組員: 0652075 0652016

2P: C4 組員: 109112128 109112118

比賽結果:

EASY: 0:3 2P WIN

第一回合: 速度20 2P WIN 第二回合: 速度25 2P WIN 第三回合: 速度20 2P WIN

NORMAL: 3:0 1P WIN

第一回合:速度191PWIN 第二回合:速度211PWIN 第三回合:速度191PWIN

HARD: 1:3 2P WIN

第一回合: 速度15 1P WIN 第二回合: 速度16 2P WIN 第三回合: 速度10 2P WIN 第四回合: 速度20 2P WIN