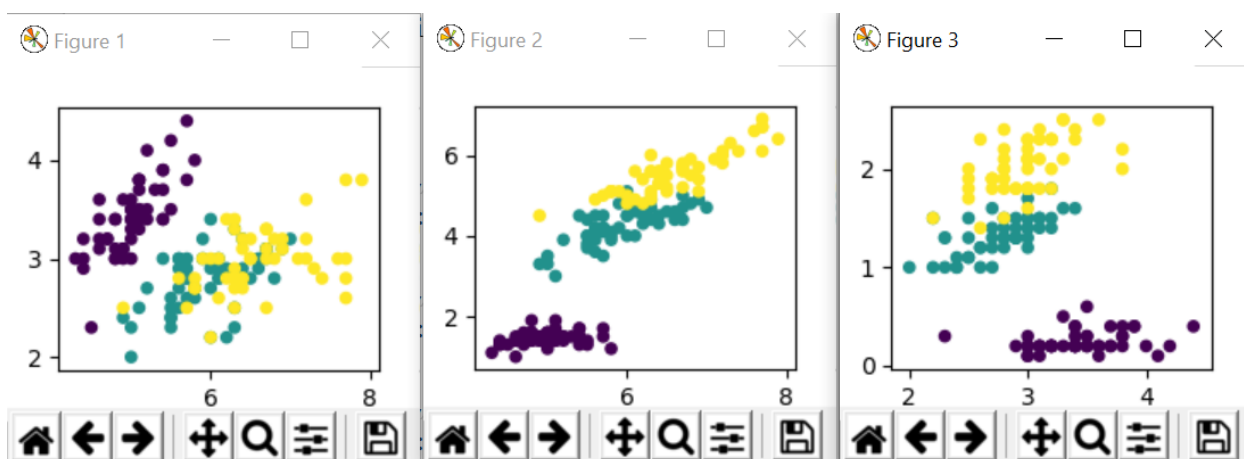


به نام خدا

سوال ۳)

(a) در این سوال ابتدا از کتابخانه sklearn به کمک دستور `datasets.load_iris()` بارگذاری می کنیم . سپس ستون های مناسب را طبق صورت سوال انتخاب و به کمک کتابخانه matplotlib ، با دستور scatter رسم می کنیم.



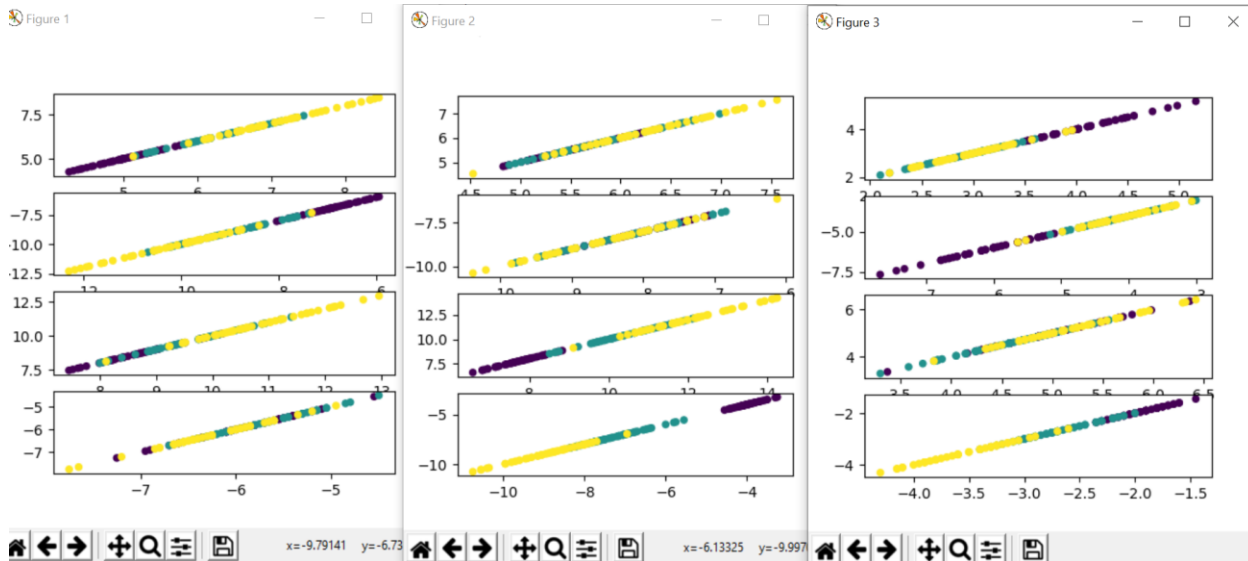
(b) در کد این سوال ابتدا ستون های گفته شده در صورت سوال را انتخاب می کنیم و در X ذخیره می کنیم:

```
X = [iris.data[:, [0, 1]] , iris.data[:, [0, 2]],  
iris.data[:, [1, 3]]]
```

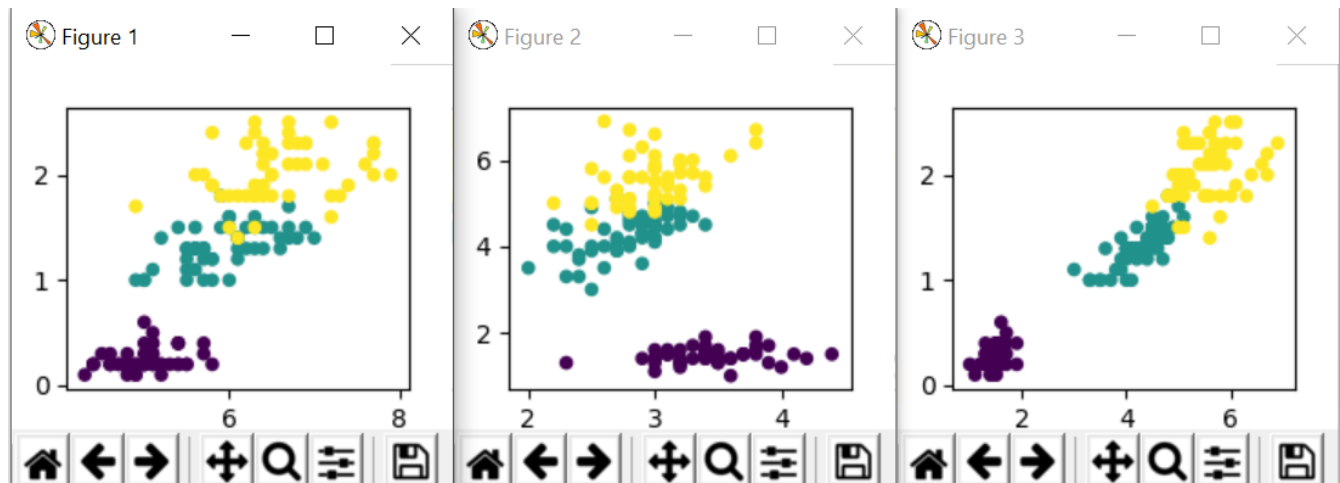
حال به کمک کتابخانه numpy این داده ها را در ماتریس های انتقال ضرب می کنیم و برای هر یک از داده های به صورت جداگانه در یک پلات نمایش می دهیم

```
for i in range(0,3) :  
    fig , myplt = plt.subplots(4)  
    for j in range(0,4):  
        dot = np.dot(X[i],t_matrix[j])  
        myplt[j].scatter(dot,dot,c=Y ,s=20)  
plt.show()
```

در نهایت حاصل اجرای کد در این مرحله به صورت زیر خواهد بود، figure ها به ترتیب مربوط به ستون های (۲،۱) و (۱،۳) و (۲،۴) می باشد و نمودارها به ترتیب از بالا مربوط به T_3, T_2, T_1 می باشد.



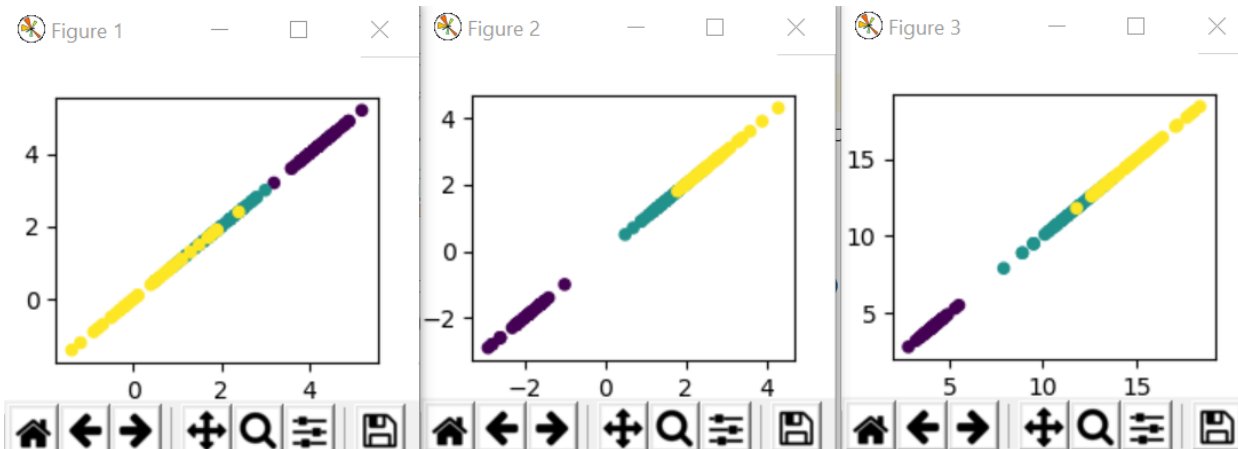
(c) برای داده اول ماتریس اول بهتر از بقیه عمل کرده ، برای داده دوم ماتریس سوم و برای داده سوم ماتریس چهارم بهترین عملکرد را داشته است
 (d) دقیقاً همانند بخش اول به رسم نمودار ها می پردازیم. نمودار ها به ترتیب ذکر شده در صورت سوال می باشد.



(e) برای هر کدام از داده ها ۴۹ ماتریس انتقال تست شد (برای هر درایه ۷ مقدار -۳ و ۰ و ۳) که بهترین آن ها به شرح زیر می باشد:

```
t_matrix = [[1,-3],[-1,1],[3,-1]]
```

و نمودار ها به شکل زیر درآمدند:



(سوال ۷)

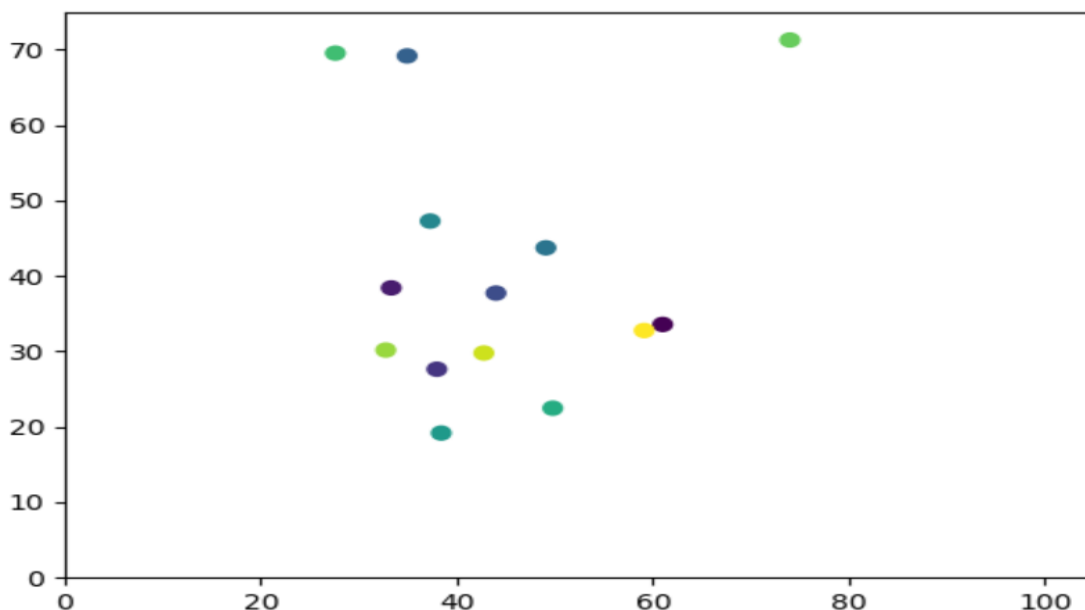
(a) در این بخش ابتدا فایل csv را به صورت خط به خط میخوانیم. سپس x و y حرکات مربوط به هر بازیکن را در آرایه player به این صورت ذخیره می کنیم.

```
players = []
for i in range(16):
    players.append([])
for d in data:
    players[int(d[1])].append([float(d[2]),float(d[3])])
```

حال به کمک کتابخانه statistics میانگین x و y ها را به صورت زیر محاسبه می کنیم:

```
for p in players:
    x_mean.append(statistics.mean(np.array(p)[: ,0]))
    y_mean.append(statistics.mean(np.array(p)[: ,1]))
    id_player.append(counter_player)
    counter_player += 1
```

و در نهایت نمودار مکان میانگین برای بازیکنان به این صورت خواهد بود :



(b) در این سوال نحوه خواندن داده ها و ساخت ماتریس **player** همانند بخش قبل است، در ادامه در یک حلقه به کمک کتابخانه **statistics** ماتریس میانگین و کوواریانس را برای هر یک از بازیکنان به شکل زیر محاسبه می کنیم:

```
for player in players :  
    if len(player) > 1:  
  
        mean.append([statistics.mean(np.array(player)[: ,0]),statistics.  
                    s.mean(np.array(player)[: ,1])])  
        number.append(player)  
  
        cov.append(np.cov(np.array(player)[: ,0],np.array(player)[: ,1]  
                        ))
```

پس از محاسبه این پارامتر ها، به کمک توابع کتابخانه **numpy** به تولید نقاط توزیع گوسی می پردازیم و پس از تولید نقاط، به رسم **heat-map** مربوطه می پردازیم، کد این بخش به صورت زیر است :

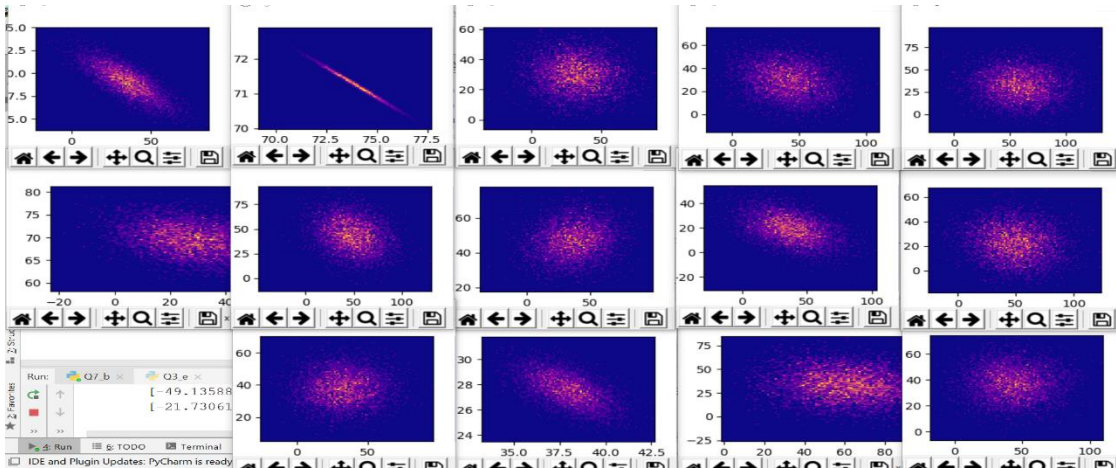
```
counter =0  
N_bins = 100
```

```

for player in range(len(mean)) :
    x, y = np.random.multivariate_normal(mean[player],
    cov[player], 5000).T
    plt.figure(counter)
    plt.hist2d(x, y, bins=N_bins, normed=False,
    cmap='plasma')
    counter += 1

```

در زیر تصویر توزیع های هر یک از بازیکنان رسم شده است:



(c) با توجه به این که در فضای دو بعدی به دنبال احتمال یک نقطه هستیم، پس احتمال برای هر نقطه ای ۰ می باشد.

(d) بنده اصلا تا بحال حتی ۱ بازی فوتبال به طور کامل ندیده ام و به هیچ وجه با نقش های فوتبال آشنایی ندارم و تنها به کمک لینک داده شده و نظر کارشناسی اعضا خانواده برای بعضی از بازیکنان نقش هایش را انتخاب کرده ام:

شماره بازیکن	نقش
۱	هافبک هجومی یا مهاجم وسط
۲	دفاع چپ
۳	؟
۴	—
۵	هافبک دفاعی

۶	بیرون در حال گرم کردن
۷	هافبک چپ
۸	دفاع چپ
۹	دفاع راست
۱۰	هافبک راست
۱۱	بیرون در حال گرم کردن
۱۲	بیرون ایستاده
۱۳	دفاع آخر
۱۴	هافبک وسط
۱۵	مهاجم نوک