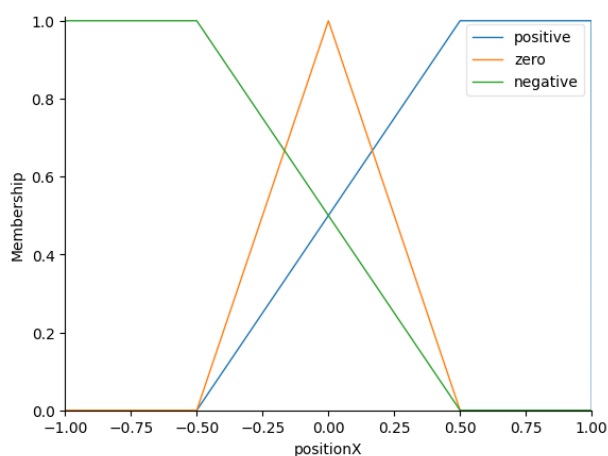


## سوال اول

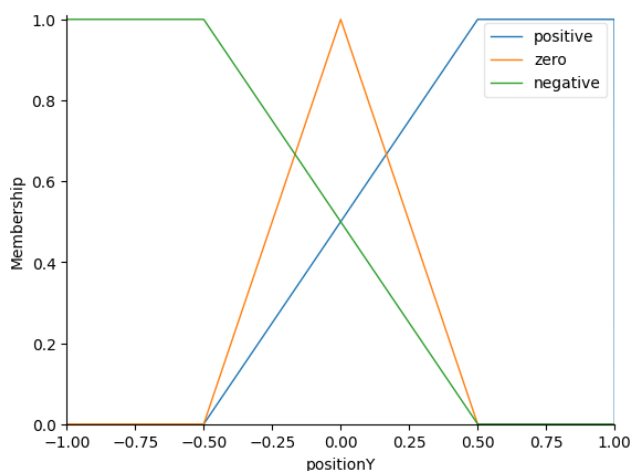
در ابتدا متغیرهای زبانی را توضیح میدهم ما در این مسئله ۳ تا observation space داریم پوزیشن  $x, y$  سرعت که هر کدام ترم های ان را به صورت زیر تعریف کردم:

با توجه به اینکه موقعی برنده میشویم که سرعت  $\bullet$  و پوزیشن  $y \bullet$  و پوزیشن  $x$  برابر با ۱ باشه برای هر کدام از از مشاهدات یک ترم zeros نیز تعریف کردم:

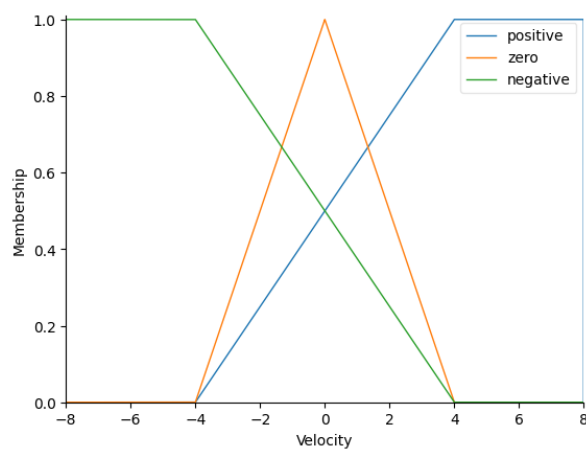
نمودار متغیر زبانی پوزیشن  $x$  برابر با شکل زیر هست:



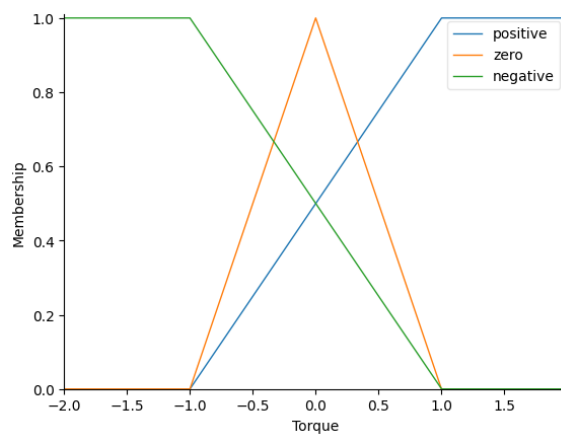
نمودار متغیر زبانی  $y$  نیز به صورت زیر است:



برای سرعت نیز به صورت زیر میشه:



و برای Torque نیز به صورت زیر میشه:



ما برای هر کدوم از مشاهدات ۳ تا ترم تعریف کردیم و پس اگر بخوایم همه قوانین موجود در نظر بگیریم ۳ به توان ۳ یعنی ۲۷ تا قانون میتونیم تعریف کنیم.

```

rule1 = ctrl.Rule(positionX['positive'] & positionY['positive'] & Velocity['positive'], Torque['negative'])
rule2 = ctrl.Rule(positionX['positive'] & positionY['positive'] & Velocity['zero'], Torque['negative'])
rule3 = ctrl.Rule(positionX['positive'] & positionY['positive'] & Velocity['negative'], Torque['negative'])
rule4 = ctrl.Rule(positionX['positive'] & positionY['zero'] & Velocity['positive'], Torque['negative'])
rule5 = ctrl.Rule(positionX['positive'] & positionY['zero'] & Velocity['zero'], Torque['negative'])
rule6 = ctrl.Rule(positionX['positive'] & positionY['zero'] & Velocity['negative'], Torque['negative'])
rule7 = ctrl.Rule(positionX['positive'] & positionY['negative'] & Velocity['positive'], Torque['negative'])
rule8 = ctrl.Rule(positionX['positive'] & positionY['negative'] & Velocity['zero'], Torque['negative'])
rule9 = ctrl.Rule(positionX['positive'] & positionY['negative'] & Velocity['negative'], Torque['negative'])

rule10 = ctrl.Rule(positionX['zero'] & positionY['positive'] & Velocity['positive'], Torque['negative'])
rule11 = ctrl.Rule(positionX['zero'] & positionY['positive'] & Velocity['zero'], Torque['negative'])
rule12 = ctrl.Rule(positionX['zero'] & positionY['positive'] & Velocity['negative'], Torque['negative'])
rule13 = ctrl.Rule(positionX['zero'] & positionY['zero'] & Velocity['positive'], Torque['zero'])
rule14 = ctrl.Rule(positionX['zero'] & positionY['zero'] & Velocity['zero'], Torque['zero'])
rule15 = ctrl.Rule(positionX['zero'] & positionY['zero'] & Velocity['negative'], Torque['zero'])
rule16 = ctrl.Rule(positionX['zero'] & positionY['negative'] & Velocity['positive'], Torque['positive'])
rule17 = ctrl.Rule(positionX['zero'] & positionY['negative'] & Velocity['zero'], Torque['positive'])
rule18 = ctrl.Rule(positionX['zero'] & positionY['negative'] & Velocity['negative'], Torque['positive'])

rule19 = ctrl.Rule(positionX['negative'] & positionY['positive'] & Velocity['positive'], Torque['negative'])
rule20 = ctrl.Rule(positionX['negative'] & positionY['positive'] & Velocity['zero'], Torque['negative'])
rule21 = ctrl.Rule(positionX['negative'] & positionY['positive'] & Velocity['negative'], Torque['negative'])
rule22 = ctrl.Rule(positionX['negative'] & positionY['zero'] & Velocity['positive'], Torque['negative'])
rule23 = ctrl.Rule(positionX['negative'] & positionY['zero'] & Velocity['zero'], Torque['negative'])
rule24 = ctrl.Rule(positionX['negative'] & positionY['zero'] & Velocity['negative'], Torque['negative'])
rule25 = ctrl.Rule(positionX['negative'] & positionY['negative'] & Velocity['positive'], Torque['positive'])
rule26 = ctrl.Rule(positionX['negative'] & positionY['negative'] & Velocity['zero'], Torque['positive'])
rule27 = ctrl.Rule(positionX['negative'] & positionY['negative'] & Velocity['negative'], Torque['positive'])

```

من دو قانون را در اینجا توضیح میدم:

- positionX: مقدار مثبت باشد positionX['positive']
- positionY: مقدار مثبت باشد positionY['positive']
- Velocity: مقدار مثبت باشد Velocity['positive']

در این صورت، مقدار Torque برابر با منفی یا همان negative قرار میدهم که به لحاظ شهودی منطقی است.

به عبارت ساده‌تر، قانون rule1 می‌گوید که وقتی که ما در شرایط مشخصی هستیم، نیاز به یک نیروی مخالف برای کنترل و تعادل داریم که در اینجا با مقدار منفی Torque نشان داده شده است.

قانون شماره ۱۶ نیز به صورت زیر است:

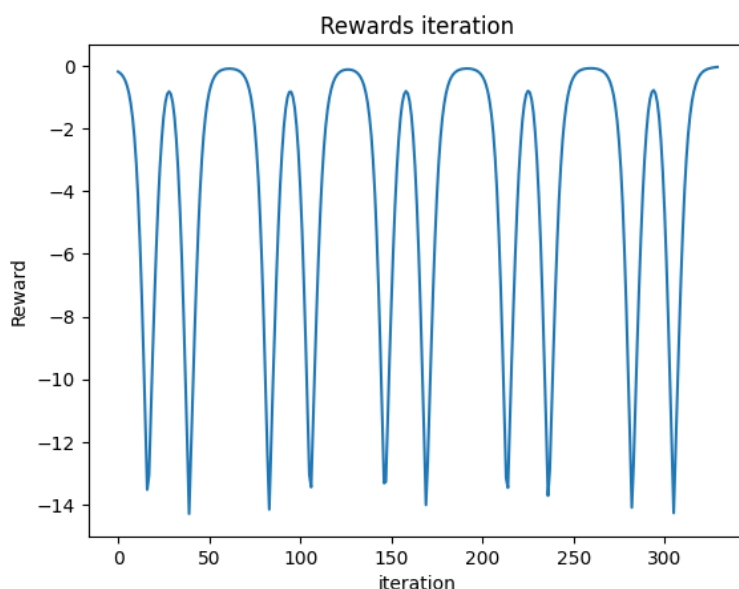
- positionX: مقدار صفر باشد positionX['zero']
- positionY: مقدار منفی باشد positionY['negative']
- Velocity: مقدار مثبت باشد Velocity['positive']

این هم از لحاظ شهودی مشخص است در واقع سرعت زمانی مثبت هست که خلاف عقربه‌های باشیم وقتی اوانگ Y ان مفی و X ان صفر هست طبیعی در خلاف جهت عقربه‌ای ساعت حرکت کنیم.

در نهایت پس از ۳۲۹ iteration برنده شدیم!

You Win in 329 iteration!

نمودار reward نیز به صورت زیر است:



نمودار reward حالت سینوسی داره در واقع مینیم ریوارد برابر با -16.8 و ماکسیمم ا برابر با ۰ هست و این حالت سینوسی ان هم اینجا هم هی به نقطه ایده ال نزدیک میشه و بعد دور میشه و این علت سینوسی بودن نمودار پداش هست.

## سوال دوم

(۱)

الگوریتم فازی Fuzzy C-Means یک الگوریتم خوشه‌بندی است که بر اساس منطق فازی عمل می‌کند. در این الگوریتم، هر نقطه یک احتمال تعلق به هر خوشه دارد و به طور کامل در یک خوشه قرار نمی‌گیرد، بلکه بازدهی قوی یا ضعیفی به خوشه‌ای مرتبط دارد که توسط فاصله معکوس تا مرکز خوشه تعیین می‌شود.

تفاوت عمده بین الگوریتم فازی C-Means و نسخه کلاسیک آن، الگوریتم K-Means، در روش تخصیص به خوشه‌ها است. در K-Means، هر نقطه به طور کامل به یک خوشه تخصیص داده می‌شود، اما در Fuzzy C-Means هر نقطه دارای یک وزن برای هر خوشه است و نقطه به طور کامل در یک خوشه قرار نمی‌گیرد. این وزن‌ها توسط فاصله معکوس تا مرکز خوشه مشخص می‌شوند.

در مورد سرعت نیز، الگوریتم Fuzzy C-Means به طور معمول کندتر از K-Means اجرا می‌شود، زیرا واقعاً کار بیشتری انجام می‌دهد. هر نقطه با هر خوشه مقایسه می‌شود و عملیات بیشتری در هر مقایسه انجام می‌شود K-Means. فقط باید محاسبه فاصله را انجام دهد، در حالی که Fuzzy C-Means باید یک وزن‌دهی کامل بر اساس فاصله معکوس انجام دهد.

نحوه عملکرد این الگوریتم هم به صورت زیر است:

۱. مشخص کردن تعداد خوشه‌ها (K) که می‌خواهیم داده‌ها را به آن‌ها تخصیص دهیم.
۲. مقداردهی اولیه تصادفی به مراکز خوشه‌ها.
۳. محاسبه مقادیر وزن (احتمال) برای هر نقطه و مرکز خوشه بر اساس فاصله آن نقطه تا مراکز خوشه.
۴. به‌روزرسانی مراکز خوشه با توجه به مقادیر وزن و تعداد نقاط در هر خوشه.
۵. تکرار مراحل ۳ و ۴ تا زمانی که مراکز خوشه مستقر شوند و الگوریتم به تعادل برسد.

(۲)

در نوتبوک کد زده شده است.

(۳)

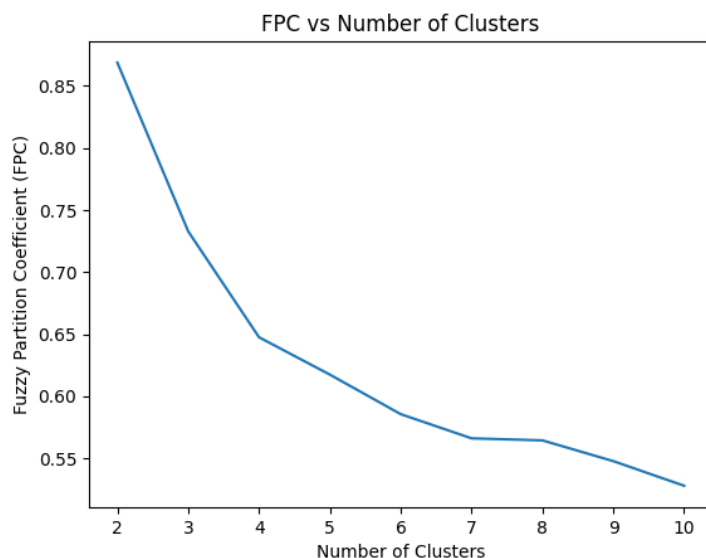
در نوتبوک زده شده است.

(۴)

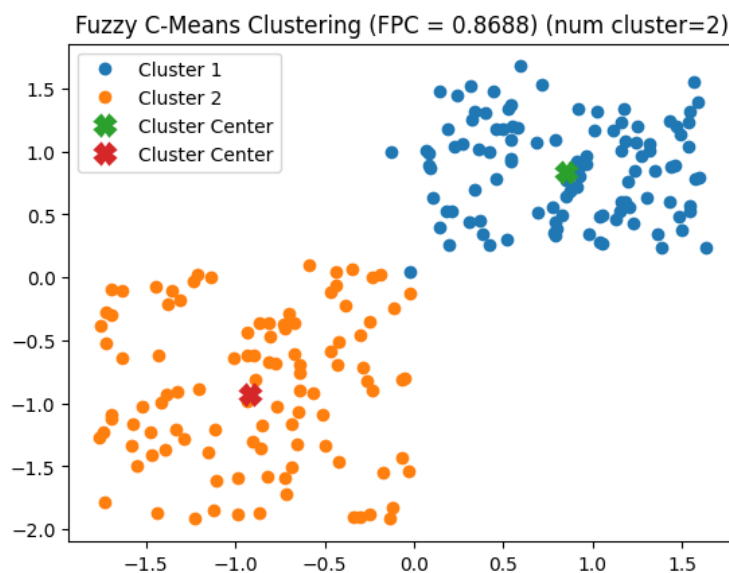
FPC یا Fuzzy Partition Coefficient ضریب تقسیم فازی است که در الگوریتم FCM برای ارزیابی خوشه‌بندی استفاده می‌شود. این مقدار نشان دهنده میزان "فازی بودن" یا "اشباع بودن" خوشه‌بندی است. برای درک بهتر، بیایید به طور خلاصه روش عملکرد الگوریتم FCM را بررسی کنیم. در الگوریتم FCM، هر نقطه داده به صورت فازی به هر خوشه تعلق می‌گیرد و درصد تعلق هر نقطه به هر خوشه با استفاده از ماتریس عضویت محاسبه می‌شود. این عضویت‌ها در مجموعه‌ای از مقادیر میانگین متعلق به ۰ و ۱ قرار می‌گیرند. با توجه به این تعاریف، FPC نسبت مقدار عضویت بیشینه به مجموع مقادیر عضویت برای تمام نقاط داده است. به عبارت دیگر، FPC نشان می‌دهد که چه قدر از نقاط داده به خوشه‌های متمایز تعلق می‌گیرند و چه قدر نقاط داده به طور فازی بین خوشه‌ها قرار دارند.

مقدار FPC بین ۰ تا ۱ قرار می‌گیرد، که عدد بزرگتر نشان دهنده خوشه‌بندی بهتر و متمایزتر است. اگر FPC به سمت ۱ نزدیک باشد، به این معنی است که نقاط داده به طور واضح و محکم در خوشه‌های متمایز قرار دارند. اما اگر FPC به سمت ۰ نزدیک باشد، نقاط داده به طور فازی و بدون مرز بین خوشه‌ها قرار دارند.

من نمودار FPC را رسم کردم که به صورت زیر شد:



با توجه به تعریف FPC بهترین تعداد cluster برابر با ۲ هست که نمودار آن به صورت زیر است :



برای data2 نیز کدها در نوتبوک موجود است. در اینجا نیز بهترین تعداد cluster برابر با ۲ هست.

سوال سوم

سوال 3

الف

W زن A س - در ابتدا برآوردی از توزیع عضویت ها را طبق جدول داده شده در سوال عمل می کنیم  
توزیع:

$$\mu_{thin}(w_1) = \left(1 + \left(\frac{55-25}{5}\right)^2\right)^{-1} = (1+36)^{-1} = \frac{1}{37} = \boxed{0.0271} \quad 45 \text{ زن } 55 \text{ زن}$$

$$\mu_{fat}(w_1) = 1 - \left(\frac{55-150}{100}\right)^2 = \boxed{0.0975}$$

$$\mu_{young}(A_1) = \left(1 + \left(\frac{45-25}{5}\right)^2\right)^{-1} = \left(1 + \left(\frac{20}{5}\right)^2\right)^{-1} = (1+16)^{-1} = \boxed{0.0588}$$

توزیع:

$$\mu_{thin}(w_2) = \left(1 + \left(\frac{95-25}{5}\right)^2\right)^{-1} = (1+44)^{-1} = \frac{1}{45} = \boxed{0.0051} \quad 60 \text{ زن } 95 \text{ زن}$$

$$\mu_{fat}(w_2) = 1 - \left(\frac{95-150}{100}\right)^2 = \boxed{0.6975}$$

$$\mu_{young}(A_2) = \left(1 + \left(\frac{60-25}{5}\right)^2\right)^{-1} = (1+49)^{-1} = \boxed{0.02}$$

توزیع نسبت میان هر دو زن که تفکیک است

در این روش محاسبه می کنیم

عملیات: توزیع نسبت میان راست

توزیع	تفکیک	result
1	0	1
0	1	0

$$\mu_{fairly fat}(w_1) = \sqrt{0.0975} = \boxed{0.31224}$$

$$\mu_{fairly fat}(w_2) = \sqrt{0.6975} = \boxed{0.835164}$$

$$\Rightarrow \frac{\mu_{fairly fat}(w_2) - \mu_{fairly fat}(w_1) + 1}{2}$$

$$A = \mu(\mu_{fairly fat}(w_1), \mu_{fairly fat}(w_2)) = \frac{0.83516 - 0.31224 + 1}{2} = \boxed{0.7614}$$

عملیات

توزیع میان زن که تفکیک است

$$B = \mu(\mu_{young}(A_1), \mu_{young}(A_2)) = \frac{0.02 - 0.0588 + 1}{2} = \boxed{0.4806} \quad B$$

$$C = A \cap B = \min(A, B)$$

عملیات: میان راست

$$= \min(0.7614, 0.4806) = \boxed{0.4806}$$

(ب) اگر فزایل حد لاغز شد آنده تفرد نسبتاً جود است

$$A \rightarrow B \Rightarrow \text{Minimum (Mamdani)} \quad \underbrace{\quad}_B \quad \underbrace{\quad}_A \quad \mu_{A \rightarrow B} = \min(\mu_A, \mu_B)$$

$$\mu_{\text{very fair}}(w_1) = (0.027)^2 = \boxed{0.000729}$$

$$\mu_{\text{fairly young}}(A_2) = \sqrt{0.02} = \boxed{0.14142}$$

$$\mu_{A \rightarrow B} (A, B) = \min(A, B) = \min(0.000729, 0.14142) = \boxed{0.000729}$$