



گزارش کار ۶ علوم اعصاب محاسباتی

عارف افضلی

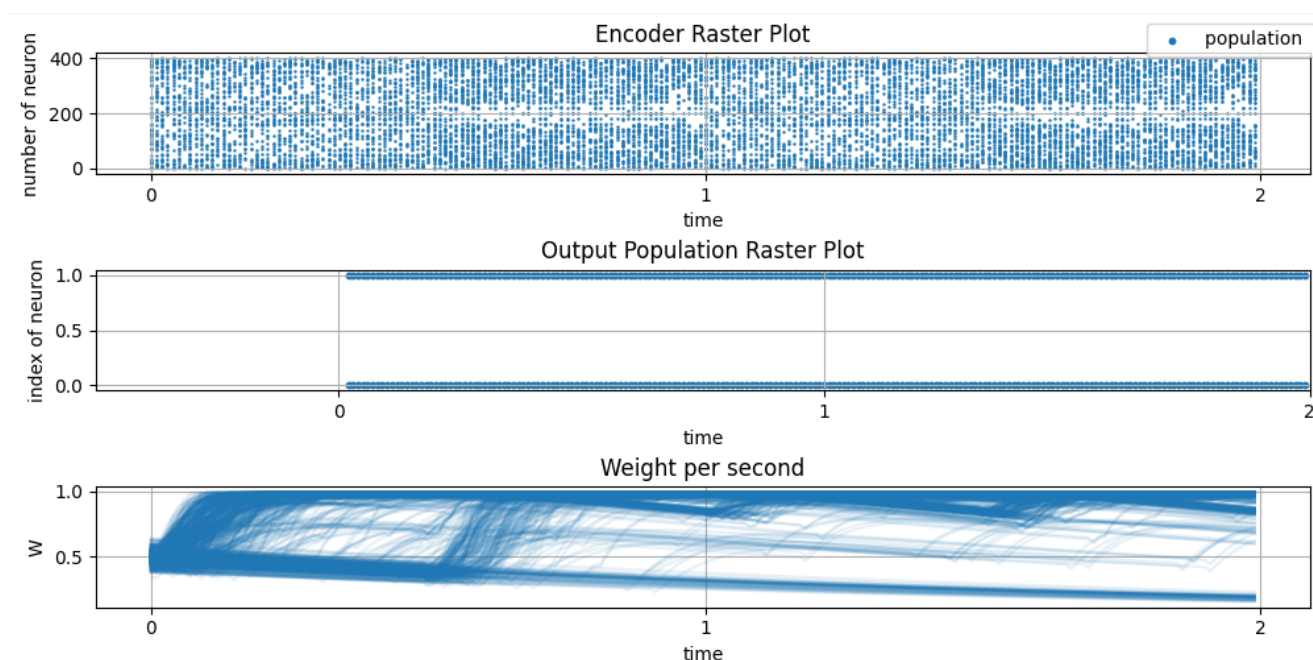
۶۱۰۰۹۸۰۱۴

بهار ۱۴۰۰

مقدمه

در این بخش به بررسی قوانین یادگیری بدون ناظر STDP و Flat-STDP می‌پردازیم. ابتدا یک جمعیت نورونی برای گرفتن داده‌ها از encoder درست می‌کنیم که بعد آن، رزولوشن عکس ورودی خواهد بود. در ادامه یک جمعیت نورونی دیگری درست می‌کنیم با دو نورون که به صورت fullyConnect این دو جمعیت به هم متصل شده‌اند. از توزیع پواسون برای اینکود کردن عکس‌ها استفاده شده است. برای ورودی ۲ عکس دوبار پشت هم اینکود شده‌اند و کنار هم قرار گرفته‌اند که خروجی آن به عنوان جریان ورودی قرار می‌گیرد.

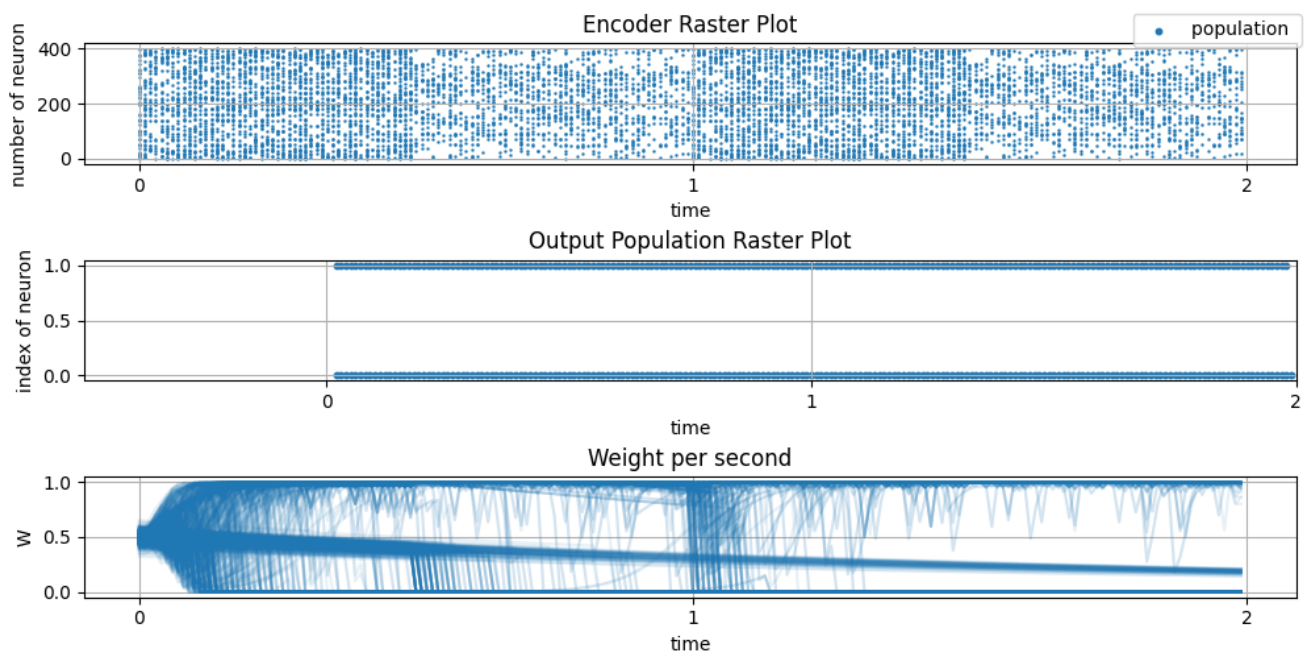
در زیر می‌توانید با مقادیر پیش فرض، یادگیری بدون ناظر STDP را مشاهده کنید.



نمودار اول نشانگر نمودار رستر برای نورون ورودی است و نمودار دوم نشانگر نمودار رستر برای دومین جمعیت نورونی که ۲ نورون دارد است. نمودار سوم نشانگر تغییرات وزن در زمان است. همانطور که مشاهده می‌کنید هر نیم ثانیه عکس عوض می‌شود و در ثانیه دوم، همانند ثانیه اول عکس‌ها تکرار می‌شوند. می‌توان دید که بعد از مدتی، وزن‌ها به مقادیری هم‌گرا می‌شوند و در زمانی که عکس عوض می‌شود وزن‌ها دوباره شروع می‌کنند به تغییر و دوباره سریعتر از قبل به سمت هم‌گرا شدن پیش می‌روند.

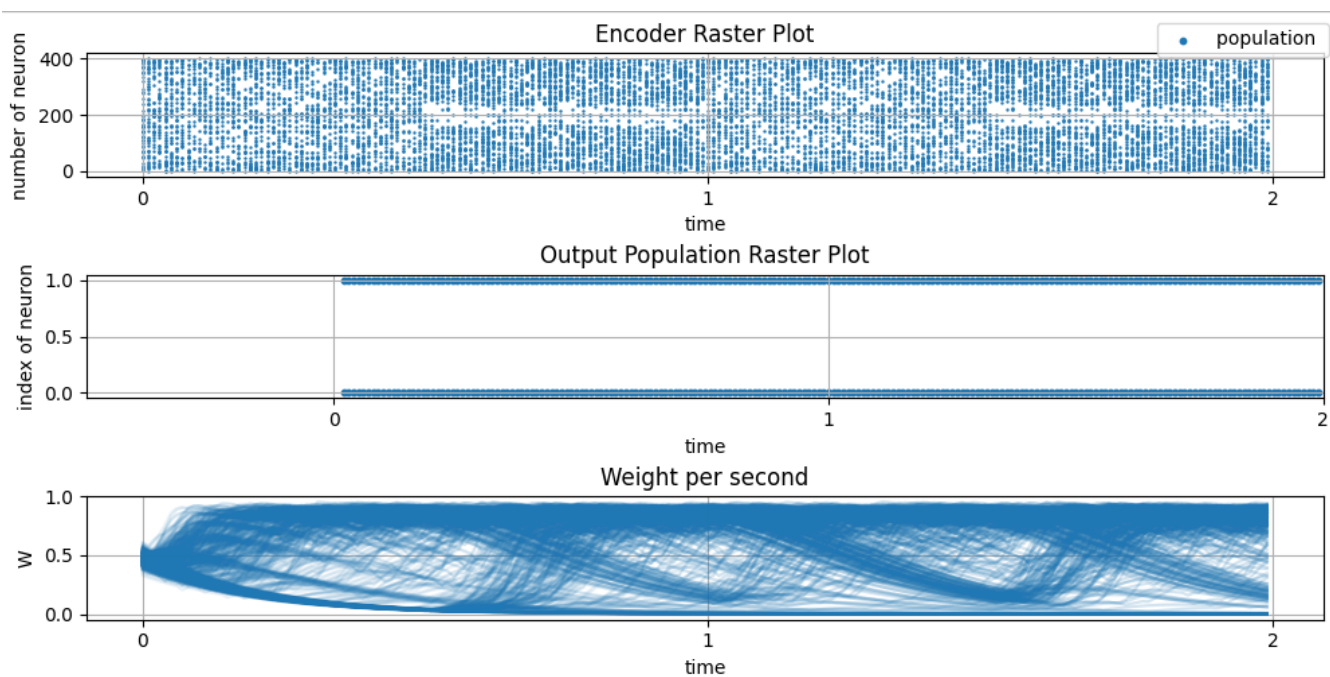
در زیر می‌توانید مدلی با یادگیری بدون ناظر FlatSTDP را ببینید که پارامترهای آن مانند پارامترهای مدل بالاست. این مدل پارامتر دیگری نیز دارد که نشانگر ماندگاری زمان تاثیر تریس یک جامعه نورونی است. هم‌گرایی در این مدل بیشتر به سمت صفر و یک است و همه وزن‌ها به بیشترین و کمترین مقدار ممکن هم‌گرا می‌شوند. دلیل این اتفاق تاثیر تعداد کثیر تریس است که باعث این اتفاق شده. ماندگاری تریس در این مدل ۰.۵ ثانیه است.

ترتیب نمودارها در این مدل و در ادامه گزارش به صورت گفته شده است.

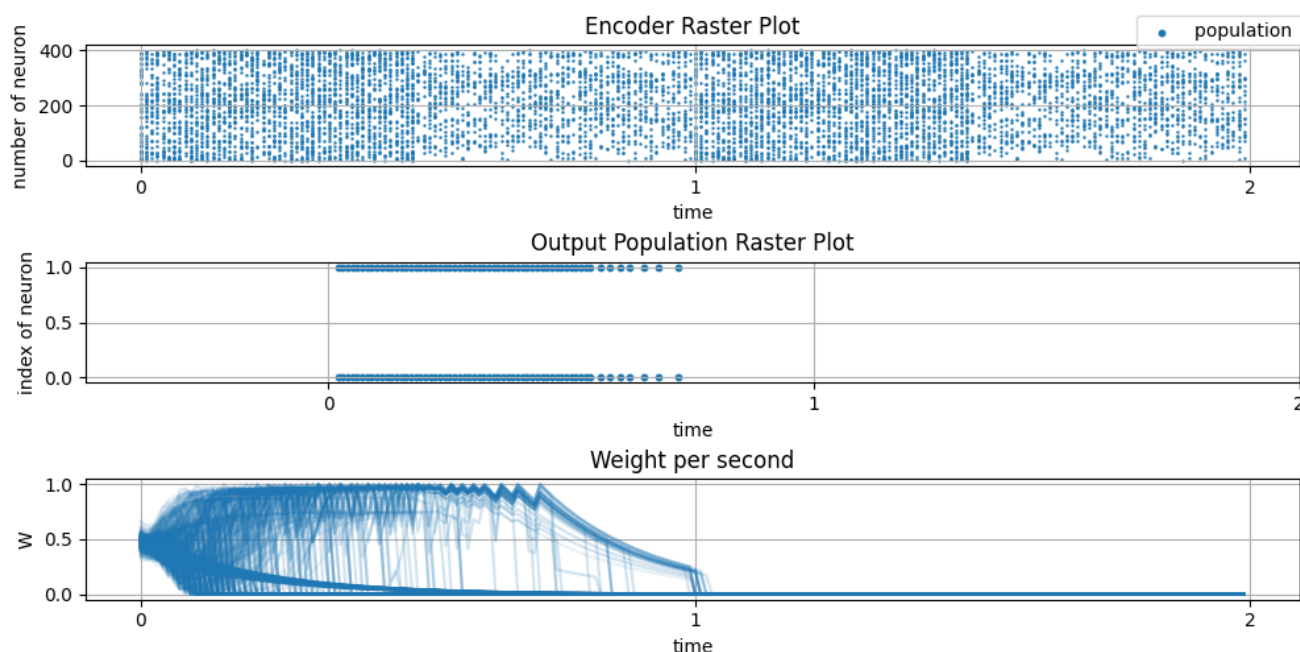


تاثیر *weight_decay*

با زیاد کردن این متغیر، می‌توان سرعت کم شدن وزن را افزایش داد و به این معنی است که زودتر وزن‌ها کاهش می‌یابند و ممکن است آن‌هایی که به مقداری همگرا می‌شدند دیگر هم‌گرا نشوند. در مثال زیر این متغیر را از ۰.۰۰۵ به ۰.۰۵ تغییر دادیم (نسبت به عکس قبل):



همان تغییر قبل را در FlatSTDP می‌توان داد و که خروجی زیر را دارد:

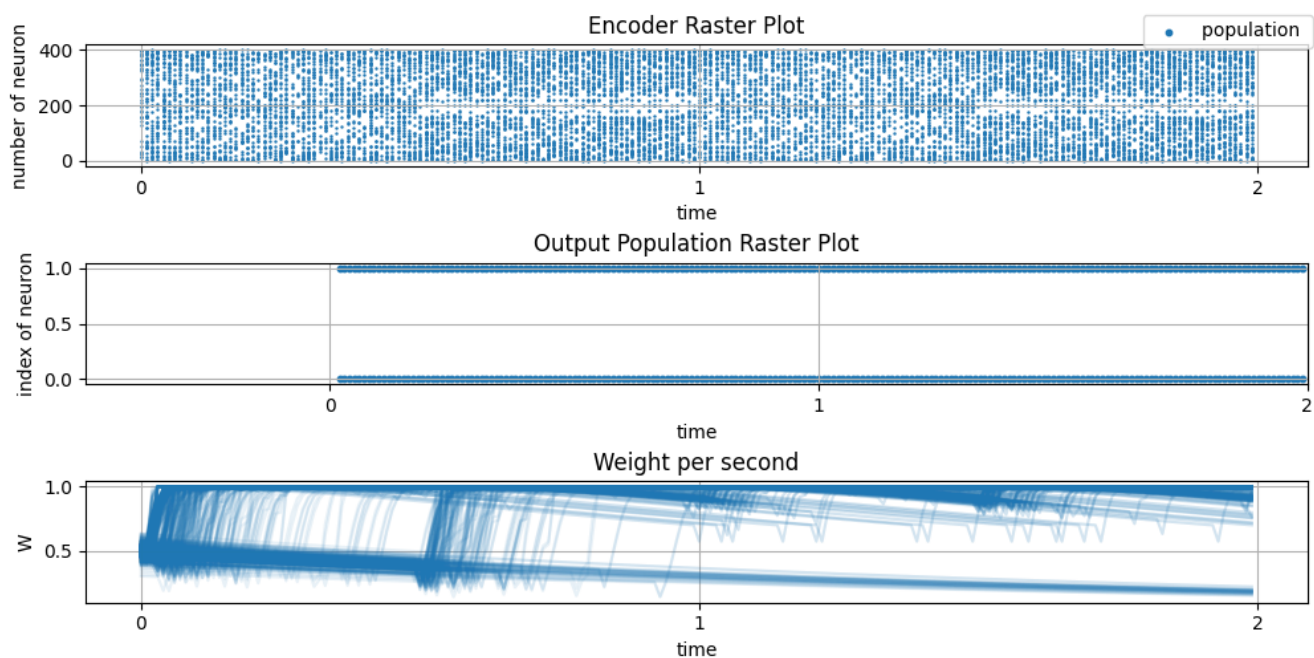


در این مدل تاثیر کم شدن وزن‌ها به آپدیت شدن آن‌ها غلبه کرده است و با توجه به این قضیه، می‌بینیم که پس از مدتی وزن‌ها صفر شده و دیگر تاثیری از نورون ورودی نخواهد گرفت.

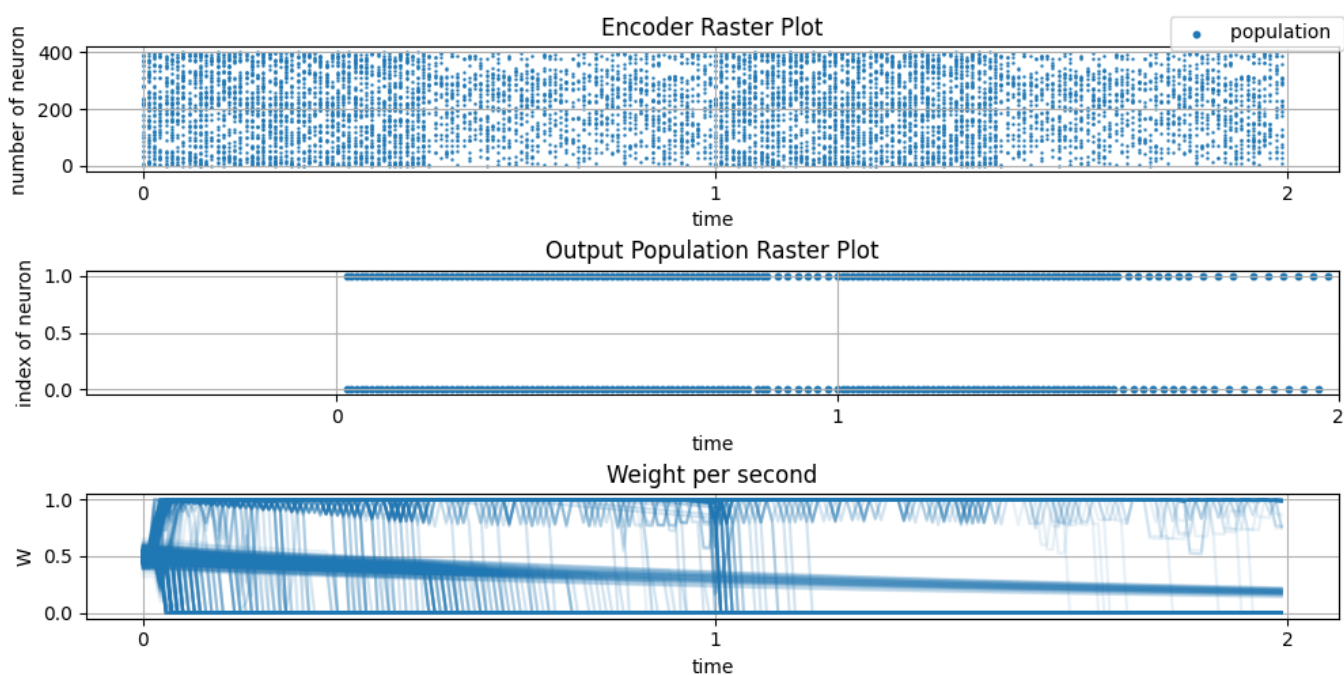
تاثیر γ (ضریب learning rate)

با افزایش گاما انتظار داریم که learning rate به صورت خطی افزایش یابد که این تغییر، افزایش تغییرات وزن‌ها را در پی دارد. این تغییر در پارامتر γ صورت می‌گیرد که در حالت پایه مقدار ۰.۲ را داشت و در نمودارهای زیر مقدار ۱.۰ را خواهد داشت.

همانطور که انتظار داشتیم در شکل زیر که مربوط به مدل STDP است، می‌بینیم که تغییرات وزن‌ها سریع‌تر انجام می‌شود نسبت به حالت پایه. با این اتفاق وزن‌ها، به یک سری از مقادیر نمی‌رسند پس از آن‌ها رد می‌کنند و به سمت ماکسیمم و مینیمم مقدار ممکن پیش می‌روند.

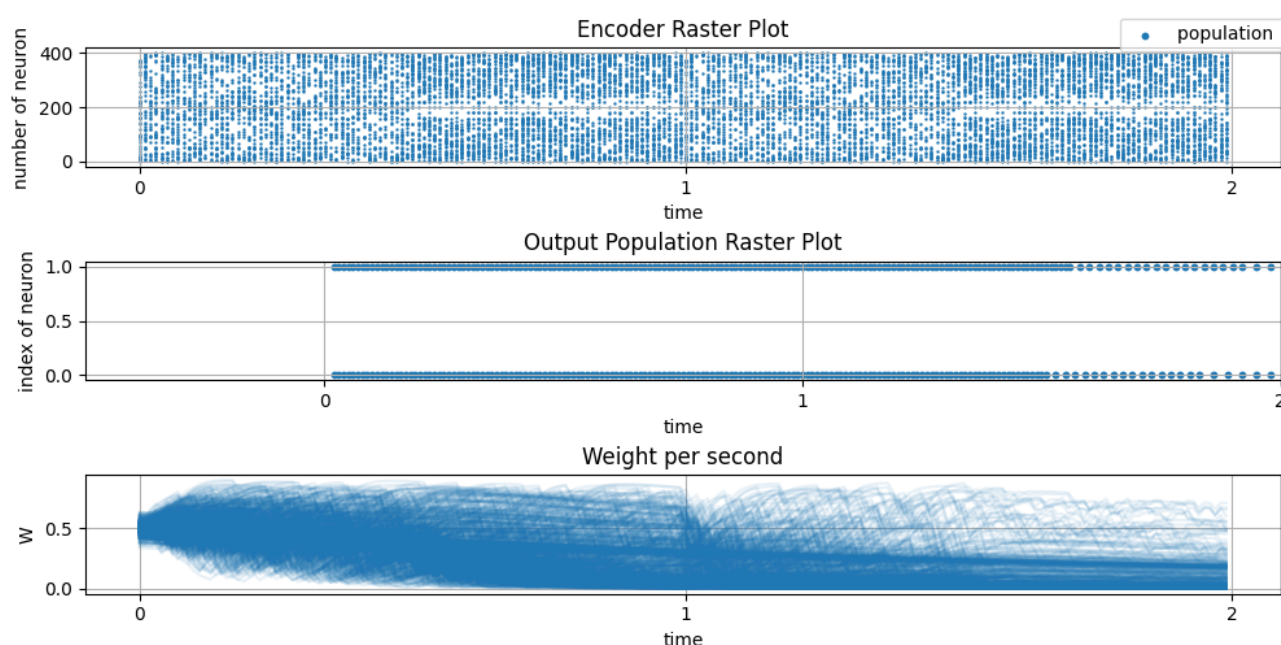


حال اگر به همین تغییر در مدل FlatSTDP نگاه کنیم می‌بینیم که دقیقا همان اتفاق بالا افتاده است و چون خودش از قبل از بخشی از لوکال مینیما ها رد می‌شد، در اینجا در همان ثانیه‌های اول به مقادیر ثابت خود تقریباً می‌رسند. آن تغییرات کوچکی که در بالا نزدیک به ۱ اتفاق می‌افتد به دلیل بوجود آمدن و یا از بین رفتن تاثیر یک تریس یک نورون است.



تاثیر ماندگاری تریس

برای مدل STDP، برای تغییر ماندگاری تریس‌ها باید τ_s را در دو جمعیت نوروئی تغییر داد. در حالت پایه ثابت زمانی جمعیت نوروئی ورودی برابر ۱۰ و ثابت زمانی جمعیت نوروئی دوم ۱ بود ولی با تغییر آن به مقادیر به ترتیب ۴ و ۴، می‌بینیم که به نتیجه بهتری نسبت به حالت پایه رسیدیم به این دلیل که وزن‌ها به مقادیر مشخصی هم‌گرا می‌شوند و وزن‌هایی که تاثیری در این شناسایی ندارند، به سمت صفر می‌روند و به مرور زمان تاثیرشان از بین می‌رود.



در مدل FlatSTDP برای تغییر ماندگاری باید پارامتر trace_periode را تغییر داد که در حالت

پایه مقدار آن ۰.۵ ثانیه بود و آن را به مقدار ۰.۰۵ تغییر دادیم. همانطور که مشاهده می‌کنیم با این تغییر مدل بهتر از حالت پایه کار می‌کند یعنی وزن‌ها به مقادیر مشخصی هم‌گرا می‌شوند و وزن‌هایی که تاثیری در این شناسایی ندارند، به سمت صفر می‌روند و به مرور زمان تاثیرشان از بین می‌رود.

