

**Electrical Engineering Department**

Computational Intelligence – Dr.F.Abdollahi

**HW1 – sigmoid function and tanh**

نام دانشجو: علی بابالو – ۹۸۲۳۰۱۲

ایمیل :‌ alibabaloo@aut.ac.ir

برای تمرین داخل کلاس مربوط به تابع سیگموید و مشتق اول آن مطابق با رابطه داده شده در پاورپوینت روابط را نوشتیم و خروجی را ریترن می­کنیم تنها نکته مربوط به این قسمت تمرین این است که برای فرستادن ماتریکس dtype باید بصورت np array باشد و لیست نمیتوان به تابع کال کرد.

در سلول بعدی تابع sigmatrix را تعریف کردیم. چون در قسمت اخر این پارت تمرین باید یک ماتریکس M به تابع میفرستادیم در تابع ابتدا چک می­کنیم که تابع کال شده ارگیومنت بهش فرستاده شده یا خیر. ابتدا خواستم از \*args استفاده کنم بعد با یک ایف ساده ( if args: … else: … ) که چک کند ارگومان دارد یا خیر ولی در ادامه چون تابع سیگموید رو صدا میکردم این ورودی رو بصورت یک tupple به تابع سیگموید میفرستاد و تایپ توپل بصورت عددی نیست برنامه ارور میداد و تابع سیگموید کار نمیکرد پس از یه روش دیگه استفاده کردم اومدم یک ارگومان M درنظر گرفتم که در حالت عادی none هستش و اگر موقع فراخوانی تابع بهش ورودی بدیم غیر none میشه با این روش شرط رو چک کردم که اگر M بصورت None باشه و اگر نباشه. اگر نان بود که همون M رو میفرستیم داخل تابع سیگموید و مشتقش میفرستیم و با استفاده از تابع np.sum کل ایندکس های ماتریس خروجی رو جمع کنه و نتیجه رو داخل یک دیکشنری ریترن بکنه. اما اگر تابع ارگومان ورودی نداشته باشه اول از یوزر تعداد سطر و ستون هارو میپرسیم و با توجه به تعداد آنها داخل یک حلقه از کاربر ایندکس های ماتریسش رو میگیریم و داخل یک لیست append میکنیم بعد اون لیست رو با تابع np.asarray به یک np array تبدیل میکنیم که به سیگموید و مشتقش بفرستیم. مانند قسمت قبل هم خروجی رو داخل یه دیکشنری نشون میدیم. فقط نکته ای که هست بای دیفالت در نظر گرفتم که ماتریس ورودی بصورت C-index ordering داده بشه نه فورترنی.

برای قسمت متلب این پارت تمرین هم که توضیح خاصی وجود نداره فقط عین قسمت پایتون اول چک میکنیم ارگومان وجود داره یا نه بعد مانند قسمت پایتون عمل میکنیم.

در سلول اخرم اون ماتریس M گفته شده رو میفرستیم به تابع و خروجیشم که مشخصه.

برای تمرین اولم که مربوط به تابع tanh میشه سلول اول و دوم و سوم تست های خودم هستش که ولش میکنیم. سل اخر تابع رو تعریف کردیم که دراون اول مانند قسمت قبلی اول سطر و ستون های ماتریس ورودی رو میگیریم و با توجه به اون یه حلقه درست میکنیم که ایندکس های ماتریس رو بگیره بعد اونارو به داخل یه لیست خالی بترتیب اپند میکنه. بعد اونو با تابع reshape به ماتریس با ابعاد کاربر تبدیل میکنیم بعد از اون لیست با دستور np.tanh، تانژانت هایپربولیک میگیریم بعدم با sum مجموع همه درایه هارو میگیریم و اونو ریترن میکنیم. قسمت متلبش هم که نکته خاصی نداره همین بالایی رو فقط تو متلب نوشتیم.