

BREAST CANCER AND MLP

Mostafa Keshavarz

در این مقاله که مورد بررسی اینجانب قرار گرفت هدف کلاس بندی بیماران مشکوک به سرطان سینه به دو دسته کسانی که سرطان دارند و کسانی که ندارند هست. این که فرد موردنظر دچار تومور شده باشد را می توان با آنالیز ابعاد و ویژگی های سلول های بدن فرد تشخیص داد.

ویژگی های دیتاست ما از 699 بیمار گرفته شده است که شامل ویژگی های ظاهری سلول فرد مورد نظر است. از نتایج گرفته شده از این کوشش , مشخص می شود که بدلیل دقت خوب تشخیص این سیستم عصبی مصنوعی می توان به آن برای تشخیص ابتدایی و سریع سرطان تکیه کرد

تکنه قابل توجه این است که در این دیتا ست مقیاس دیتا برای هر ستون متفاوت است و باید با مقادیر همان ستون نورمالیز شوند

ساختار شبکه عصبی

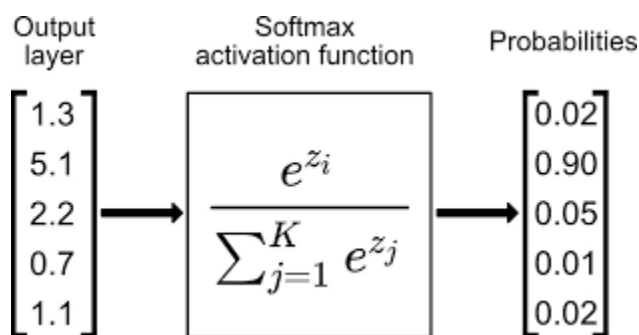
شبکه عصبی مورد استفاده در این مقاله یک شبکه سه لایه معمولی با optimizer گرادیان نزولی است که 9 نرون ورودی دارد 15 نرون مخفی و 2 نرون خروجی به صورت softmax برای کلاس بندی بیماران

بیماران در دسته benign و malignant دسته بندی می شوند.

من در ابتدا همین ساختار را پیاده کردم و نتایج مطلوب و مورد انتظار را گرفتم و سپس بعد از شبکه را با استفاده از نرون های RBF پیاده سازی کردم ولی نتیجه بهبودی نکرد و بدتر شد

اگر بخواهم درباره مفاهیم جدید این شبکه نسبت به مفاهیم صحبت شده در کلاس صحبت کنم می توان به softmax اشاره کرد لایه ایی که کار آن تشخیص احتمال تعلق نمونه وروی به هرکدام از کلاس هاست

Softmax به صورت کلی به فرمول زیر کار می کند



حال موضوع مهم Label های ما به ازای هر ورودی و تایین Loss function است

در این جا ما دو Label داریم برای دو کلاس خود . از آنجا که خروجی ما یک ارایه از احتمالات هست پس مقدار مطلوب ما نیز باید ارایه ایی از احتمالات باشد ولی واضح است که مقدار مطلوب ما ارایه ایی از جنس one hot encode در می آید زیرا احتمال در یک کلاس 1 و در باقی صفر است

پس برای شروع این پروسه ما باید در وهله اول label های هر نمونه را به صورت one hot encode در بیآوریم

یعنی به صورت $[0\ 1]$ و به این صورت $[1,0]$

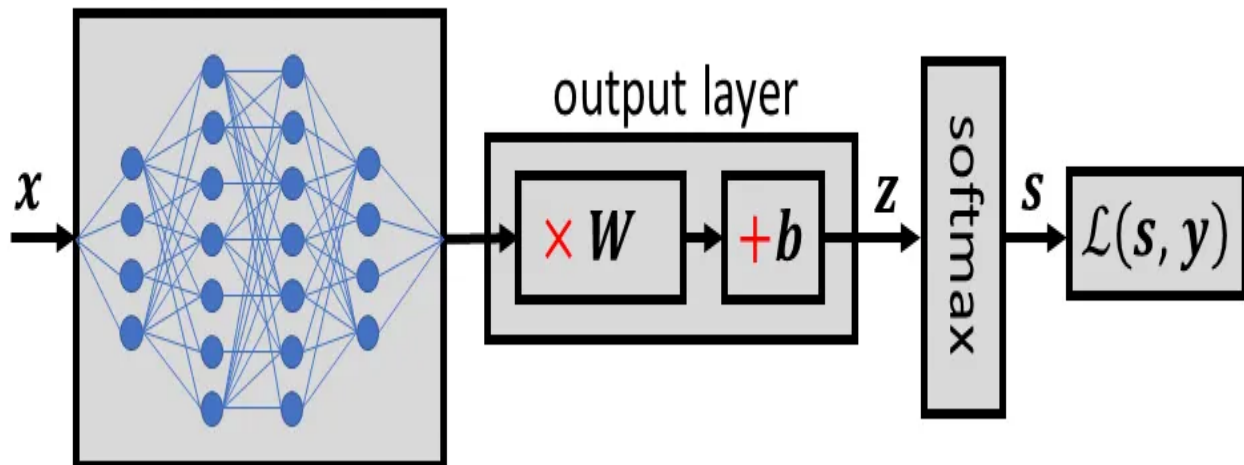
حال به سراغ تائین Loss function می رویم

Loss function مورد استفاده از مورد به اسم Categorical Crossentropy معروف است که به شکل فرمول زیر است

$$\text{Loss} = - \sum_{i=1}^{\text{output size}} y_i \cdot \log \hat{y}_i$$

در این رابطه \hat{y} مقدار مطلوب و y خروجی شبکه است و سپس از Loss function برای استفاده در فرمول گرادیان نزولی استفاده میکنیم ولی باید به یک نکته توجه کرد و آن نیز مشتق تابع Softmax است که به صورت ماتریس

some network



$$s_i = \frac{e^{z_i}}{\sum e^{z_i}}$$

$$\frac{\partial s_i}{\partial z_j} = s_i \frac{\partial}{\partial z_j} \log(s_i)$$

$$\log(s_i) = z_i - \log\left(\sum e^{z_i}\right)$$

$$\frac{\partial}{\partial z_j} \log(s_i) = \frac{\partial z_i}{\partial z_j} - \frac{\partial}{\partial z_j} \log\left(\sum e^{z_i}\right) = 1\{i=j\} - \frac{1}{\sum e^{z_i}} \left(\frac{\delta}{\delta z_j} \left(\sum e^{z_i}\right)\right) = 1\{i=j\} - e^{z_j}$$

$$\frac{\partial s_i}{\partial z_j} = s_i(1\{i=j\} - s_j)$$

پس ماتریس ژاکوبین Softmax به صورت زیر می شود

$$J_{softmax} = \begin{pmatrix} s_1 \cdot (1 - s_1) & -s_1 \cdot s_2 & -s_1 \cdot s_3 & -s_1 \cdot s_4 \\ -s_2 \cdot s_1 & s_2 \cdot (1 - s_2) & -s_2 \cdot s_3 & -s_2 \cdot s_4 \\ -s_3 \cdot s_1 & -s_3 \cdot s_2 & s_3 \cdot (1 - s_3) & -s_3 \cdot s_4 \\ -s_4 \cdot s_1 & -s_4 \cdot s_2 & -s_4 \cdot s_3 & s_4 \cdot (1 - s_4) \end{pmatrix}$$

نمودار ها خروجی

در ابتدا نمودار های خروجی را برای شبکه گفته شده در مقاله بیان میکنیم



در نمودار واضح است که Loss ما به ازای هر تناوب کمتر و کمتر شده است و دقت نیز بیشتر شده است

دقت نهایی که از شبکه بدست آمده 95 درصد است که دقت مطلوبی است و به دقت ذکر شده در مقاله یعنی 96 درصد نزدیک است

در این فرایند Training از معیار های دیگری نیز برای تشخیص عملکرد شبکه استفاده شده است

یادآوری (Recall) : نسبت بین تشخیص های درست سرطان و به کل نمونه های واقعی سرطان

صحت (Precision) : نسبت بین تشخیص های صحیح سرطان توسط شبکه به کل نمونه های سرطان تشخیص داده شده توسط شبکه چه صحیح و چه اشتباه

در حالت ایده آل این دو معیار باید یک باشند ولی از انجایی که در شبکه همیشه نمونه هایی به صورت False negative یا False positive وجود دارد پس باید یک Trade off بین این دو متریک برقرار کرد و هردوی آن ها را باهم دید

حال برای حل این مشکل از معیار F1 score استفاده می شود

F1 Score: این معیار یک میانگین هارمونیک بین Precision و Recall هست و فرمول آن به صورت زیر است:

$$F1 = \frac{1}{\frac{1}{Precision} + \frac{1}{Recall}}$$

هرچه این معیار به سمت یک نزدیک تر باشد یعنی شبکه ما عملکرد بهتری در زمینه های Precision و Recall دارد

حال به بررسی خروجی برای شبکه RBF می پردازیم :



در این نمودار واضحا پیداست که شبکه RBF با گزشت زمان آموزش مناسبی پیدا نکرده و پیشرفتی از خود نشان نداده است که این نشان از ضعف شبکه RBF برای این نمونه از کار است

این مقاله نشان دهنده قدرت شبکه عصبی در تقسیم بندی کلاس ها بود

نکته قابل توجه این است که شبکه RBF اصلا عملکرد مطلوبی را در این زمینه نداشت شاید می توان این را در این نکته توجیح کرد که این نوع از شبکه ها زاتا برای دیتا هایی که امکان نویز پذیری دارند مناسبند و برای کارهای کلاس بندی به این شکل توجیه مناسبی ندارند

برای دریافت کد ها و دیتاها و مدل های آماده شده به لینک زیر برید:

<https://github.com/mostafaksh78/Neural-network-project-40025724.git>