

فصل ۹

تنظیم کردن ابرپارامترها

اهداف این جلسه

- آشنایی با نحوه پردازش تصاویر MRI
- در پایان این جلسه شما قادر خواهید بود که:
- آموزش شبکه‌های پیچشی
- تشخیص تومورهای مغزی در داده‌های MRI
- آموزش یک مدل پیچشی با استفاده از ابزار PyTorch

۱.۹ قسمت اول: تنظیم پارامترها

۱.۱.۹ آموزش یک دسته بند

در این بخش می‌خواهیم یک شبکه‌ی پیش‌بینی برای تشخیص تومورهای مغزی از روی داده‌های MRI مربوط به مغز طراحی کنیم. ** لطفا تمامی مراحل را به ترتیب انجام دهید و نتایج خواسته شده را گزارش کنید. ** با توجه به اینکه تعداد ایپاک‌های آموزشی و پارامترهای شبکه زیاد هستند، پیشنهاد می‌شود که حتماً از Colab در حالت GPU استفاده کنید.

- داده‌های مربوط به مغز را با استفاده از لینک زیر دانلود کنید. در صورتیکه مایل به استفاده از Colab هستید، می‌توانید از راهنمای موجود در لینک قرار داده شده برای لود کردن داده‌ها استفاده کنید: [Tumor-Classification](#)

- ابتدا داده‌های آموزشی و تست را لود کنید.

- سائز و نوع features و label داده‌های train چاپ کنید.

- یک batch از داده‌های آموزش را نشان داده شده است.

- یک شبکه کاملاً متصل به شکل زیر ایجاد کنید:

```
nn.Linear(*,512)
```

```
nn.ReLU
```

```
nn.Linear(512,24)
```

```
ReLU
```

```
nn.Linear(24,2)
```

علامت * در لایه اول به این معنی است که باید سائز ورودی را با توجه به داده‌ها مشخص کنید و به جای ستاره قرار دهید.

- ویژگی‌های مدل ساخته شده را چاپ می‌کنیم.

- تابع train-loop را برای آموزش مدل، مطابق با شرایط زیر بنویسید:

۱. در این تابع از موارد زیر استفاده کنید:

model: the neural network

dataLoader: an object that takes batch-output and batch-labels and computes the loss for the batch

optimizer: optimizer for parameters of the model

۲. در هر مرحله از انجام محاسبات backward، باید مقدار loss و تعداد batch های در نظر گرفته شده در آن مرحله نشان داده شوند.

- تابع test-loop را برای تست شبکه، مطابق با شرایط زیر بنویسید:

۱. در این تابع از موارد زیر استفاده کنید:

model: the neural network

dataLoader: an object that fetches data

lossFunction: a function that takes batch-output and batch-labels and computes the loss for the batch

۲. در پایان هر مرحله از تست باید مقدار درصد دقت و میانگین loss نشان داده شود.

- با در نظر گرفتن مقادیر زیر برای پارامترها، شبکه را برای ۱۰۰ اپیاک آموزش دهید.

LearningRate = 1e-3

LossFunction = CrossEntropy

optimizer = SGD

- نمودار درصد دقت و مقدار loss شبکه بر روی داده‌های تست را بر حسب اپیاک رسم کنید.

- یک شبکه پیچشی به شکل زیر ایجاد کنید:

```
nn.Conv2d(3,32,kernel_size=3,stride=1, padding=1)
```

```
nn.ReLU()
```

```
nn.Conv2d(32,64,kernel_size=3,stride=1, padding=1)
```

```
nn.ReLU()
```

```
nn.MaxPool2d(2,2)
```

```
nn.Flatten()
```

```
nn.Linear(*,512)
```

```
nn.ReLU()
```

```
nn.Linear(512,24)
```

```
nn.ReLU()
```

```
nn.Linear(24,2)
```

علامت * در لایه اول به این معنی است که باید سایز ورودی را با توجه به داده‌ها مشخص کنید و به جای ستاره قرار دهید.

- ویژگی‌های مدل ساخته شده را چاپ کنید.

- این شبکه را نیز با همان پارامترهای قبلی برای ۱۰۰ اپیاک آموزش دهید.

- نمودار درصد دقت و مقدار loss شبکه را بر روی داده‌های تست بر حسب اپیاک رسم کنید.

چه تفاوتی بین نمودار رسم شده برای مدل ۱ و ۲ وجود دارد؟ چرا؟

۲.۱.۹ تنظیم پارامترهای یک دسته بند برای تشخیصی تومورهای مغزی

Wandb یکی از ابزارات معروف برای بهینه‌سازی مدل‌های یادگیری عمیق و همچنین برای تنظیم کردن ابرپارامترها مانند نرخ یادگیری و سایر پارامترهای قابل تنظیم در شبکه‌های عصبی مورد استفاده قرار می‌گیرد. ابتدا نیاز است که در تارنمای [Wandb](#) ثبت نام نمایید و سپس API Key را از حساب کاربری خود برای کار در این جلسه استفاده نمایید. همچنین ابزار comet نیز یک وسیله کاربردی است که برای تنظیم نرخ پارامترها استفاده می‌شود.

در قسمت قصد داریم هابپارامتر نرخ یادگیری را در جهت کاهش میزان خطا بر روی داده‌های آموزشی، تنظیم نماییم. ابتدا یک روش را به صورت اختیاری انتخاب نمایید و از ابزار wandb و یا comet استفاده کنید:

- با استفاده از جستجوی تصادفی

- با استفاده از روش بیزین