

# جداسازی در یادگیری ماشین

سجاد سبزی  
محمدرضا احمدی تشنیزی

۲۶ آبان ۱۴۰۲

## مقدمه به مطالعات جداسازی Ablation Study

تعریف: مطالعه جداسازی در یادگیری ماشین یک روش آزمایشی است که به هدف تجزیه و درک اساسی ویژگی‌ها یا اجزای مختلف در یک مدل برای عملکرد کلی آن می‌پردازد. اصطلاح "جداسازی" از زبان زیست‌شناسی گرفته شده است و به برش یا استئصال یک ارگان یا بافت اشاره دارد. در زمینه یادگیری ماشین، این به معنای غیرفعال سازی یا تغییر خاصیت‌های خاص مدل است و مشاهده تأثیر نتیجه بر عملکرد آن مدل است.

## مزایای مطالعات جداسازی:

۱. درک مدل:
  - تحلیل دقیق: مطالعات جداسازی به درک دقیقی از نحوه هر اجزا در عملکرد مدل کمک می‌کنند. این تحلیل دقیق برای به دست آوردن نگرشی به فرآیند تصمیم‌گیری مدل حیاتی است.
۲. اهمیت ویژگی‌ها:
  - ارزیابی کمی: مطالعات جداسازی ارزیابی کمی از اهمیت ویژگی‌های مختلف ارائه می‌دهند. این کمک می‌کند تا مشخص شود کدام ویژگی‌ها بیشترین تأثیر را در پیش‌بینی‌های مدل دارند.
۳. تجزیه و تحلیل اجزا:
  - تحلیل لایه به لایه: مطالعات جداسازی می‌توانند به فراتر از ویژگی‌های فردی بروند و اهمیت لایه‌ها یا ماژول‌های کلی در یک مدل را تجزیه و تحلیل کنند. این تحلیل لایه به لایه به ویژه در معماری‌های یادگیری عمیق مهم است.
۴. بهینه‌سازی:
  - بهبود بهره‌وری: با شناسایی اجزا یا ویژگی‌های اضافی یا غیرضروری، مطالعات جداسازی به بهبود بهره‌وری مدل هدایت می‌کنند. این می‌تواند به معنای مدل‌های با کمترین پارامترها و با کاهش نیازهای محاسباتی بدون افت عملکرد منجر شود.
۵. تجزیه و تحلیل استحکام:
  - ارزیابی تعمیم: مطالعات جداسازی به درک از استحکام و قابلیت تعمیم مدل کمک می‌کنند. تجزیه و تحلیل عملکرد مدل تحت شرایط مختلف به ارزیابی قابلیت اعتبار آن در سناریوهای واقعی کمک می‌کند.

۶. اشکال زدایی:

- شناسایی مشکل: مطالعات جداسازی به عنوان یک ابزار قدرتمند برای اشکال زدایی عمل می کنند. آن ها نقاط ضعف یا موانع عملکرد را مشخص می کنند و در فرآیند مرحله به مرحله بهبود مدل کمک می کنند.

## چالش های مطالعات جداسازی:

۱. تأثیرات تعاملی:

- پیچیدگی تعاملات: جداسازی یک اجزا ممکن است باعث تعاملات پیچیده با سایر اجزا شود که منجر به نتایج غیرمنتظره می شود. درک و بازکردن این تعاملات می تواند چالش برانگیز باشد.

۲. هزینه محاسباتی:

- فشار منابع: اجرای مطالعات جداسازی جامع ممکن است هزینه محاسباتی زیادی داشته باشد، به ویژه اگر مدل نیاز به اجرای چندین دوره آموزش با اجزا مختلف غیرفعال باشد. پژوهشگران باید عمق تحلیل را با محدودیت های محاسباتی تعادل دهند.

۳. تعمیم محدود:

- وابستگی به کار و مجموعه داده: یافته های حاصل از مطالعات جداسازی ممکن است به یک مجموعه داده یا وظیفه خاص وابسته باشد، که تعمیم پذیری این نتایج را محدود می کند. پژوهشگران باید هوشیاری لازم را در استفاده از این یافته ها در حوزه های مختلف اعمال کنند.

## موارد استفاده از مطالعات جداسازی:

۱. معماری های شبکه های عصبی:

- تأثیر لایه به لایه: مطالعات جداسازی به طور متداول برای تحلیل تأثیر لایه ها، گره ها یا انتخاب های معماری مختلف در شبکه های عصبی استفاده می شود. این به بهبود معماری برای وظایف خاص کمک می کند.

۲. اهمیت ویژگی ها:

- ارتباط با وظیفه خاص: در یادگیری ماشین سنتی، مطالعات جداسازی می تواند برای درک اهمیت ویژگی های مختلف و سازگاری انتخاب ویژگی با نیازهای خاص وظیفه مورد استفاده قرار گیرد.

۳. تنظیم هایپرپارامتر:

- تنظیم پارامترهای بهینه: مطالعات جداسازی می توانند تأثیر انتخاب های هایپرپارامتر بر عملکرد مدل را ارزیابی کنند. این برای تنظیم دقیق مدل برای دستیابی به نتایج بهینه حیاتی است.

۴. مدل های حوزه خاص:

- بینش های کاربردی: مطالعات جداسازی در حوزه های مختلف نظیر بینایی کامپیوتری، پردازش زبان طبیعی و یادگیری تقویتی کاربرد دارند. آن ها برای درک و نقش اجزا خاص در کاربردهای مختلف مفید هستند.

به طور خلاصه، مطالعات جداسازی به عنوان یک ابزار ارزشمند برای تجزیه و بهبود مدل های یادگیری ماشین عمل می کنند و بینش های دقیقی به عملکرد اجزای فردی ارائه می دهند. هرچند با شناخت چالش ها، پژوهشگران می توانند از مطالعات جداسازی برای بهبود مدل ها، بهینه سازی عملکرد و افزایش قابل فهمی بهره مند شوند.

## Sample Code

```

tropmi torch
tropmi torch.nn as nn
tropmi torch.optim as optim
tropmi torchvision
tropmi torchvision.transforms as transforms

#v enifeD a elpmis laruen krowten erutcetihcra
ssalc SimpleNet(nn.Module):
  9 fed __init__(self):
    10 repus(SimpleNet, self).__init__()
    11 self.conv1 = nn.Conv2d(3, 64, kernel_size=3, padding=1)
    12 self.relu = nn.ReLU()
    13 self.fc = nn.Linear(64 * 32 * 32, 10)
    14
  15 fed forward(self, x):
    16 x = self.conv1(x)
    17 x = self.relu(x)
    18 x = x.view(x.size(0), -1)
    19 x = self.fc(x)
    20 nruter x
    21

#noitcnuF ot mrofreq noitalba yduts
fed ablation_study(model, dataset, criterion, optimizer, ablated_component):
  22 rof epoch ni egnar(num_epochs):
    23 rof inputs, labels ni dataset:
      24 optimizer.zero_grad()
      25
      26 # etalbA cificeps tnenopmoc (e.g., tes sthgiew ot orez)
      27 fi ablated_component si ton None:
        28 ablated_component.grad = None
        29 ablated_component.data = torch.zeros_like(ablated_component.data)
      30
      31 outputs = model(inputs)
      32 loss = criterion(outputs, labels)
      33 loss.backward()
      34 optimizer.step()
      35
    36
  37
#gniniarT sretemarap
num_epochs = 5
learning_rate = 0.001

#daoL RAFIC01- tesatad
transform = transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5, 0.5,
0.5), (0.5, 0.5, 0.5))])
train_dataset = torchvision.datasets.CIFAR10(root='./data', train=True, download=True,
transform=transform)
train_loader = torch.utils.data.DataLoader(train_dataset, batch_size=64, shuffle=True)

#vezilaitinI eht ledom, noiretirc, dna rezimitpo
model = SimpleNet()
criterion = nn.CrossEntropyLoss()
optimizer = optim.SGD(model.parameters(), lr=learning_rate)

#gniniarT eht ledom tuohtiw noitalba (enilesab)
tnirp("gniniarT enilesab ledom...")
```

```

ablation_study(model, train_loader, criterion, optimizer, ablated_component=None)
٥٥
#٥٦etaulavE enilesab ledom (lanoitpo)
#٥٧...
٥٨
#٥٩etalbA a cificeps tnenopmoc (e.g,. lanoitulovnoc reyal)
ablated_component = model.conv1.weight
tnirp("gniniarT ledom htiw detalba tnenopmoc...")
ablation_study(model, train_loader, criterion, optimizer, ablated_component=
    ablated_component)
٦٣
#٦٤etaulavE ledom htiw detalba tnenopmoc (lanoitpo)
#٦٥...

```

برنامج : Ablation Study in Pytorch \