



# بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ

## گزارش کار تمرین ۳-۲ - یادگیری عمیق

عنوان تمرین :

طراحی معماری بهینه شبکه عصبی

مطالعه سیستماتیک ابلیشن (Ablation Study)

مجموعه داده : Fashion-MNIST

دانشجو : محمد داود وهاب رجائی

شماره دانشجویی : ۴۰۴۱۴۱۹۰۴۱

استاد محترم : دکتر کیوان راد

حل تمرین : دکتر گلی زاده

## فهرست مطالب

چکیده.....	۱
مقدمه و روش‌شناسی.....	۱
فاز اول : اکتشاف و تحلیل مؤلفه‌ها.....	۲
۱. ضرورت غیرخطی بودن (Non-Linearity).....	۲
۲. مقایسه توابع فعال‌ساز (Activation Functions).....	۲
۳. تحلیل روش‌های تنظیم (Regularization).....	۲
فاز دوم : طراحی مدل قهرمان (The Champion Model).....	۳
نتایج آزمایش‌های نهایی.....	۳
تحلیل نهایی.....	۴
نتیجه‌گیری.....	۴
پیوست.....	۴

## چکیده

هدف این پژوهش، یافتن بهترین معماری شبکه عصبی تمام‌متصل (Fully Connected) برای طبقه‌بندی تصاویر Fashion-MNIST از طریق یک رویکرد تجربی سیستماتیک است. در فاز اول، ۱۵ آزمایش اکتشافی برای بررسی تأثیر ایزوله توابع فعال‌ساز، روش‌های تنظیم (Regularization) و عمق شبکه انجام شد. نتایج نشان داد که استفاده از توابع غیرخطی و نرمال‌سازی دسته‌ای (Batch Normalization) بیشترین تأثیر مثبت را دارند. بر اساس این یافته‌ها، در فاز دوم، سه معماری نامزد (Champion) طراحی و تست شدند. مدل نهایی با ترکیب Leaky ReLU و Batch Normalization توانست به دقت اعتبارسنجی ۹۰.۰۷٪ دست یابد که نسبت به مدل پایه خطی (۸۴.۸۳٪)، بهبود قابل توجهی را نشان می‌دهد.

## مقدمه و روش‌شناسی

مطالعه ابلیشن (Ablation Study) روشی است که در آن اجزای یک سیستم هوش مصنوعی به صورت جداگانه حذف یا تغییر داده می‌شوند تا سهم هر جزء در عملکرد کلی مشخص شود. برای این آزمایش، از یک چارچوب کدنویسی استاندارد با قابلیت بازتولید (Reproducibility) استفاده شد. تمامی آزمایش‌ها با  $\text{Seed}=42$ ، تعداد ۲۰ اپک، و بهینه‌ساز Adam اجرا شدند.

معماری پایه (Baseline):

- ورودی: ۷۸۴ (۲۸×۲۸ پیکسل)
- لایه‌های مخفی: [۱۲۸, ۲۵۶]
- خروجی: ۱۰ کلاس

## فاز اول : اکتشاف و تحلیل مؤلفه‌ها

در این مرحله، تأثیر هر مؤلفه به صورت جداگانه بررسی شد. نتایج کلیدی به شرح زیر است :

### ۱. ضرورت غیرخطی بودن (Non-Linearity)

مقایسه مدل خطی خالص با مدل‌های دارای تابع فعال‌ساز نشان داد که یادگیری الگوهای تصویری بدون غیرخطی‌سازی ممکن نیست.

- مدل خطی (Linear Baseline) : دقت ۸۴.۸۳٪
- مدل پایه با ReLU : دقت ۸۹.۹۲٪
- نتیجه : افزودن تابع فعال‌ساز باعث جهش ۵ درصدی در دقت مدل شد.

### ۲. مقایسه توابع فعال‌ساز (Activation Functions)

عملکرد توابع مختلف در شرایط یکسان مقایسه شد :

- Sigmoid (۸۹.۱۳٪) و Tanh (۸۹.۱۵٪) : عملکرد ضعیف‌تر نسبت به ReLU ، احتمالاً به دلیل مشکل محو شدن گرادیان (Vanishing Gradient) در اپک‌های میانی.
- ReLU (۸۹.۹۲٪) : عملکرد استاندارد و سریع.
- Leaky ReLU (۸۹.۷۳٪ - ۹۰.۳۰٪ در تکرارها) : بهترین عملکرد را نشان داد. این تابع با داشتن شیب ملایم در قسمت منفی، از مشکل "مرگ نورون‌ها" (Dead ReLU) جلوگیری می‌کند.
- Maxout و ELU : عملکرد پایین‌تری نسبت به خانواده ReLU داشتند و پیچیدگی محاسباتی Maxout توجیه اقتصادی نداشت.

### ۳. تحلیل روش‌های تنظیم (Regularization)

- Batch Normalization (BN) : مؤثرترین تکنیک بود. مدل bn\_relu به دقت ۹۰.۰۳٪ (در آزمایش‌های اولیه) رسید. BN با نرمال‌سازی ورودی هر لایه، آموزش را پایدارتر و سریع‌تر کرد.

- Dropout : استفاده از Dropout با نرخ ۰.۳ باعث افت دقت شد (۸۹.۱۲٪). این نشان می‌دهد که برای این شبکه نسبتاً کوچک و تعداد اپک کم، حذف ۳۰ درصد نورون‌ها باعث "Underfitting" جزئی شده و مانع یادگیری مؤثر می‌شود.
- داده‌افزایی (Augmentation) : تأثیر معناداری در ۲۰ اپک نداشت. داده‌افزایی معمولاً در آموزش‌های طولانی‌مدت برای جلوگیری از بیش‌برازش مؤثر است.

## فاز دوم : طراحی مدل قهرمان (The Champion Model)

بر اساس تحلیل‌های فاز اول، استراتژی زیر برای طراحی مدل بهینه اتخاذ شد :

۱. انتخاب فعال‌ساز Leaky ReLU : به عنوان جایگزین قدرتمندتر ReLU انتخاب شد.
۲. انتخاب تنظیم‌گر Batch Normalization : به عنوان عضو ضروری تثبیت شد.
۳. مدیریت Dropout : از آنجا که نرخ ۰.۳ مخرب بود، در یکی از مدل‌ها نرخ ملایم‌تر (۰.۲) تست شد و در مدل دیگر کلاً حذف شد.
۴. معماری : از آنجا که افزایش عمق (۳ لایه) در فاز اول تأثیر مثبتی نداشت، تمرکز بر روی معماری ۲ لایه باقی ماند، اما یک نسخه "عریض‌تر (Wider)" نیز تست شد.

## نتایج آزمایش‌های نهایی

نام مدل (Run ID)	معماری	فعال ساز	Batch Norm	Dropout	دقت نهایی	تحلیل
Champion ۱	[۱۲۸,۲۵۶]	Leaky ReLU	Yes	۰	۹۰.۰۷٪	برنده : تعادل عالی بین سرعت و دقت
Champion ۲	[۱۲۸,۲۵۶]	Leaky ReLU	Yes	۰.۲	۹۰.۰۵٪	درآپاوت کم هم تفاوت معناداری ایجاد نکرد
Champion ۳	[۲۵۶,۵۱۲]	Leaky ReLU	Yes	۰	۸۹.۸۲٪	افزایش عرض شبکه باعث پیچیدگی بی‌په‌وده شد.

## تحلیل نهایی

۱. سینرژی Leaky ReLU و BN : ترکیب این دو به شبکه اجازه داد تا گرادیان‌ها را به خوبی در شبکه جریان دهد (به لطف Leaky ReLU) و توزیع داده‌ها را در لایه‌های میانی کنترل کند (به لطف BN).
۲. پرهیز از بیش‌برازش (Over-regularization) : در شبکه‌هایی با اندازه متوسط مانند این مدل روی دیتاست Fashion-MNIST، استفاده همزمان از Dropout سنگین و Augmentation در ۲۰ اپک، مانع از همگرایی کامل مدل می‌شود. مدل برنده با حذف این سربارها، سریع‌تر یاد گرفت.
۳. کفایت معماری: نتایج نشان داد که معماری [۱۲۸, ۲۵۶] ظرفیت کافی (Capacity) برای یادگیری ویژگی‌های این دیتاست را دارد و بزرگ‌تر کردن آن (۳ Champion) تنها هزینه محاسباتی را بالا برده و دقت را اندکی کاهش می‌دهد (احتمالاً به دلیل سخت‌تر شدن بهینه‌سازی).

## نتیجه‌گیری

در این تمرین، با اجرای ۱۸ آزمایش کنترل‌شده، ما از یک مدل خطی با دقت ۸۴٪ به یک مدل بهینه با دقت ۹۰.۰۷٪ رسیدیم.

پیکربندی پیشنهادی نهایی برای Fashion-MNIST با محدودیت منابع فعلی عبارت است از:

- لایه‌ها : ۲ لایه مخفی [۱۲۸, ۲۵۶]
- تابع فعال ساز : Leaky ReLU
- تکنیک تنظیم : Batch Normalization (بدون Dropout)
- تعداد اپک : ۲۰

## پیوست

- فایل fashion\_mnist\_fc\_ablation\_results.csv شامل ریز نتایج تمامی آزمایش‌ها.
- فایل model\_champion\_۱\_leaky\_bn.pth (وزن‌های مدل برتر)