



دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

تمرین سوم هوش محاسباتی: شبکه های عصبی و کاربردها

Neural Networks & Applications

نگارش

دانیال شفیعی

مهدی مهدیه

امیررضا نجفی

استاد راهنما

دکتر کارشناس

خرداد ۱۴۰۴

فهرست مطالب

۰	مقدمه	۲
۱	مفاهیم و حل مسئله	۲
۲	کدزنی و پیاده سازی	۴

۰ مقدمه

هدف از این تمرین آشنایی بیشتر با شبکه های عصبی و استفاده ی بیشتر از آن ها در کاربردهای عملی است.

۱ مفاهیم و حل مسئله

۱. بله، هر نورون در یک شبکه عصبی حامل نوعی اطلاعات است؛ اما ماهیت و میزان «وضوح» این اطلاعات بسته به عمق لایه و ویژگی های بنیادین شبکه متفاوت است.
چهار ویژگی بنیادی و سلسله مراتبی بودن نمایش:

(آ) توابع غیرخطی (Nonlinearity)

- هر نورون پس از ترکیب خطی ورودی ها (ضرب وزن ها + بایاس) خروجی را از طریق تابعی مانند ReLU، sigmoid یا tanh عبور می دهد.
- بدون غیرخطی سازی، شبکه عملاً یک عملگر خطی بزرگ خواهد بود و قادر به تشخیص زیرویرگی های پیچیده نیست.
- تابع فعال سازی باعث می شود هر نورون تنها در صورت وقوع یک الگوی خاص «فعال» شود و در نتیجه به عنوان یک تشخیص دهنده ساده عمل کند.

(ب) نمایش توزیع شده (Distributed Representation)

- برخلاف سیستم های سمبلیک که هر مفهوم را با یک واحد منفرد نمایش می دهند، شبکه های عصبی مفاهیم را به صورت همزمان در بردار فعال سازی تعداد زیادی نورون کدگذاری می کنند.
- این پراکندگی اطلاعات باعث افزایش مقاومت شبکه در برابر نویز و آسیب به نورون های منفرد می شود.
- هر نورون سهم جزئی اما معنادار در تشخیص زیرویرگی های ساده یا انتزاعی دارد.

(ج) یادگیری گرادیان محور (Gradient-based Learning)

- با استفاده از الگوریتم پس انتشار (Backpropagation)، وزن ها و بایاس هر نورون به روزرسانی می شود تا خطای خروجی به کمترین مقدار برسد.
- در طی آموزش، هر نورون به زیرویرگی هایی پاسخ می دهد که برای کاهش خطا در مسئله مشخص مفیدند.
- در پایان آموزش، وزن های ورودی هر نورون تعیین می کنند که آن نورون به چه الگو یا ویژگی حساس باشد.

(د) سلسله‌مراتب ویژگی‌ها (Hierarchical Feature Learning)

- لایه‌های ابتدایی شبکه‌های عمیق معمولاً به زیرویژگی‌های ساده مانند لبه‌های عمودی/افقی یا بافت‌ها حساس‌اند.
- لایه‌های میانی ترکیب این زیرویژگی‌ها را انجام داده و الگوهای پیچیده‌تر را می‌آموزند.
- در لایه خروجی (مثلاً نورون‌های softmax) احتمال تعلق هر ورودی به یک کلاس نهایی (مثلاً «گره» یا «سگ») کدگذاری می‌شود.

۲. در شبکه‌های عصبی، «دانش» در قالب پارامترها (وزن‌ها و بایاس‌ها) ذخیره می‌شود و از طریق فرآیند آموزش شکل می‌گیرد؛ در ادامه، یک پاسخ یکپارچه و مرتب‌شده ارائه شده است:

(آ) شکل‌گیری دانش در شبکه‌های عصبی

- تعریف ساختار شبکه (Architecture): انتخاب تعداد لایه‌ها (Input, Hidden, Output)، نوع آن‌ها (fully-connected، کانولوشن، بازگشتی و ...) و تعداد نورون در هر لایه.
- مقداردهی اولیه پارامترها (Initialization): وزن‌ها و بایاس‌ها معمولاً با توزیع‌های تصادفی (مثل Xavier یا He) مقداردهی می‌شوند.
- انتشار رو به جلو (Forward Propagation): برای هر ورودی x ، در هر لایه:

$$z^{(\ell)} = W^{(\ell)} a^{(\ell-1)} + b^{(\ell)}, \quad a^{(\ell)} = \sigma(z^{(\ell)})$$

در نهایت $a^{(L)}$ خروجی نهایی شبکه است.

- محاسبه خطا (Loss Calculation): با تابع هزینه $L(y_{\text{pred}}, y_{\text{true}})$ ، مانند MSE برای رگرسیون یا Cross-Entropy برای طبقه‌بندی.
- پس انتشار خطا (Backpropagation): مشتق تابع هزینه را نسبت به پارامترها محاسبه می‌کنیم:

$$\frac{\partial L}{\partial W^{(\ell)}}, \quad \frac{\partial L}{\partial b^{(\ell)}}$$

- به‌روزرسانی پارامترها (Optimization): با الگوریتم‌هایی مثل Gradient Descent یا Adam:

$$W^{(\ell)} \leftarrow W^{(\ell)} - \eta \frac{\partial L}{\partial W^{(\ell)}}, \quad b^{(\ell)} \leftarrow b^{(\ell)} - \eta \frac{\partial L}{\partial b^{(\ell)}}$$

این چرخه تا رسیدن به همگرایی تکرار می‌شود.

(ب) فرمول‌بندی «معادل بودن» دو شبکه عصبی

- معادل تابعی (Exact Functional Equivalence) دو شبکه $M(x)$ و $N(x) = f_{\theta_N}(x)$ اگر:

$$\forall x \in X, \quad N(x) = M(x).$$

- معادل ساختاری تحت تبدیلات (Structural Equivalence) در لایه‌های Dense، جابجایی نورون‌ها (پرموتیشن π) همراه با جابجایی سطرها/ستون‌های متناظر در W, b ، خروجی را تغییر نمی‌دهد.

iii. تقریب معادل (*Approximate Equivalence*) با فاصله خروجی $d(x) = \|N(x) - M(x)\|_p$:

$$\forall x \in X, d(x) < \epsilon \quad \text{یا} \quad \text{KL}(N(x) \| M(x)) < \delta.$$

(ج) مثال ریاضی

i. حالت ساده خطی: دو شبکه خطی با یک لایه پنهان و $\sigma(z) = z$:

$$N(x) = W_2 (W_1 x + b_1) + b_2, \quad M(x) = W'_2 (W'_1 x + b'_1) + b'_2.$$

آنها معادل اند اگر:

$$W_2 W_1 = W'_2 W'_1, \quad W_2 b_1 + b_2 = W'_2 b'_1 + b'_2.$$

ii. اشاره‌ای به حالت غیرخطی: در شبکه‌های غیرخطی (مثلاً ReLU)، تبدیلات پیچیده‌ترند؛ اما با ادغام BatchNorm یا تبدیلات جبری می‌توان مشابهت رفتار را نشان داد.

۲ کدزنی و پیاده سازی