

১. গ্র্যাডিয়েন্ট ডিজেন্ট (Gradient Descent):

গ্র্যাডিয়েন্ট ডিজেন্ট হলো একটি অপটিমাইজেশন অ্যালগোরিদম যা মেশিন লার্নিং মডেল, বিশেষ করে নিউরাল নেটওয়ার্কে, প্যারামিটার বা ওজন (weights) আপডেট করার জন্য ব্যবহৃত হয়। এর মূল উদ্দেশ্য হলো মডেলটির ভুল (error বা loss) কমানো। গ্র্যাডিয়েন্ট ডিজেন্ট অ্যালগোরিদমের মাধ্যমে, আমরা "কস্ট ফাংশন" (Loss Function) এর মান ন্যূনতম (minimize) করতে চেষ্টা করি।

কিভাবে কাজ করে:

- প্রথমে, মডেলটি র্যান্ডমভাবে কিছু প্যারামিটার (ওজন) নির্বাচন করে।
- তারপর, কস্ট ফাংশন ব্যবহার করে, মডেলটির ভুল হিসাব করা হয়।
- গ্র্যাডিয়েন্ট (ধারা) বের করা হয় কস্ট ফাংশনের পরিবর্তনশীলতার উপর ভিত্তি করে, এবং প্যারামিটারগুলো আপডেট করা হয় এই গ্র্যাডিয়েন্টের বিপরীতে।

উদাহরণ: ধরা যাক, আমাদের একটি সরল রেখা ফিট করতে হবে এবং সেই রেখার জন্য কিছু প্যারামিটার (slope এবং intercept) বের করতে হবে। গ্র্যাডিয়েন্ট ডিজেন্ট আমাদের সাহায্য করবে এই প্যারামিটারগুলিকে সঠিক মানে নিয়ে আসতে। আমরা প্রতিটি আপডেটের সময় প্যারামিটারগুলিকে একটু একটু করে পরিবর্তন করবো যাতে কস্ট ফাংশনের মান কমে।

২. অ্যাকটিভেশন ফাংশন (Activation Functions):

অ্যাকটিভেশন ফাংশন একটি নিউরাল নেটওয়ার্কের নিউরনের আউটপুট ঠিক করতে সাহায্য করে। এটি ইনপুট সিগন্যালগুলির যোগফলকে আউটপুটে রূপান্তরিত করে, যা মডেলকে নির্দিষ্ট সিদ্ধান্ত নিতে সহায়ক হয়।

প্রধান অ্যাকটিভেশন ফাংশনসমূহ:

1. Sigmoid:

- মান: ০ থেকে ১ এর মধ্যে।
- সাধারণত বাইনারি ক্লাসিফিকেশন কাজের জন্য ব্যবহৃত হয়।
- সূত্র: $\sigma(x) = \frac{1}{1 + e^{-x}}$

উদাহরণ: এই ফাংশনটি আপনার ইনপুটকে সিগময়েড রূপে রূপান্তরিত করবে, যেখানে ১ মানে "হ্যাঁ" এবং ০ মানে "না"।

2. ReLU (Rectified Linear Unit):

- মান: ০ থেকে অ-সীমা পর্যন্ত (যদি ইনপুট ০ এর নিচে হয় তবে আউটপুট ০ হয়)।

- এটি বর্তমান নিউরাল নেটওয়ার্কের মধ্যে সবচেয়ে জনপ্রিয় অ্যাকটিভেশন ফাংশন।
- সূত্র: $\text{ReLU}(x) = \max(0, x)$

উদাহরণ: যদি ইনপুট ধরা হয় ৩, তবে আউটপুট হবে ৩। তবে যদি ইনপুট ধরা হয় -২, তবে আউটপুট হবে ০।

3. Tanh (Hyperbolic Tangent):

- মান: -১ থেকে ১ এর মধ্যে।
- এটি সিগময়েডের মত, কিন্তু আউটপুট মানকে -১ থেকে ১ এর মধ্যে সীমাবদ্ধ রাখে।
- সূত্র: $\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$

উদাহরণ: যদি ইনপুট ০.৫ হয়, তাহলে আউটপুট হবে ০.৪৭।

৩. লস ফাংশন (Loss Function) / কস্ট ফাংশন (Cost Function):

লস ফাংশন হলো এমন একটি ফাংশন যা মডেলের পূর্বাভাস এবং প্রকৃত মানের মধ্যে পার্থক্য পরিমাপ করে। এটি আমাদের বলে দেয় মডেলটি কতটুকু ভুল করছে, এবং সেই অনুযায়ী প্যারামিটারগুলো আপডেট করা হয়।

প্রধান লস ফাংশনসমূহ:

1. Mean Squared Error (MSE):

- এটি রিগ্রেশন সমস্যায় সবচেয়ে সাধারণ লস ফাংশন। এটি পূর্বাভাস এবং প্রকৃত মানের পার্থক্যের বর্গ (squared difference) এর গড় হিসাব করে।
- সূত্র:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

যেখানে y_i হল প্রকৃত মান এবং \hat{y}_i হল পূর্বাভাস।

উদাহরণ: যদি আমাদের প্রকৃত মান ৫ এবং পূর্বাভাস ৪.৫ হয়, তাহলে MSE হবে $(5 - 4.5)^2 = 0.25$ ।

2. Binary Cross-Entropy (Log Loss):

- এটি সাধারণত বাইনারি ক্লাসিফিকেশন সমস্যায় ব্যবহৃত হয়। এটি পূর্বাভাসের লগ (log) এর মাধ্যমে ভুল পরিমাপ করে।
- সূত্র:

$$\text{Log Loss} = -\frac{1}{n} \sum_{i=1}^n [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

$\text{Log Loss} = -\frac{1}{n} \sum_{i=1}^n [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$

উদাহরণ: যদি একটি ক্লাসিফিকেশন মডেল ০.৮ পূর্বাভাস দেয় এবং প্রকৃত ফলাফল ১ হয়, তাহলে লস হবে:

$$-\log(0.8) - \log(0.8)$$

3. Categorical Cross-Entropy:

- এটি মাল্টি-ক্লাস ক্লাসিফিকেশন সমস্যায় ব্যবহৃত হয় যেখানে একাধিক আউটপুট ক্লাস থাকে।
- এটি মডেলের পূর্বাভাস এবং প্রকৃত লেবেলের মধ্যে ক্যাটাগরিক্যাল লস পরিমাপ করে।

উদাহরণ: ধরা যাক, আপনি একটি নিউরাল নেটওয়ার্ক ব্যবহার করছেন একটি ক্লাসিফিকেশন সমস্যা সমাধান করতে। আপনার লক্ষ্য হলো একটি ইমেজকে সঠিকভাবে শ্রেণীবদ্ধ করা (যেমন, কুকুর বা বিড়াল)। যদি মডেলটি ভুল পূর্বাভাস দেয়, তবে কস্ট ফাংশন মডেলকে জানাবে কতটুকু ভুল হয়েছে এবং গ্র্যাডিয়েন্ট ডিজেন্ট সেই ভুলের ভিত্তিতে মডেলটির প্যারামিটারগুলো আপডেট করবে।

এভাবে, মডেলটি ধীরে ধীরে শিখবে এবং পারফরম্যান্স উন্নত করবে।