Herkansing wiskunde

Gradient descent met 3 voorbeelden en 2 gewichten

1e Iteratie

Laten we gradient descent toepassen op de Mean Squared Error (MSE) met minimaal 3 trainingsvoorbeelden en twee gewichten w_1 en w_2. We zullen starten met de initiële gewichten $w_1 = 1$ en $w_2 = 0.5$.

Trainingsvoorbeelden:
Laten we drie trainingsvoorbeelden gebruiken:

1. $(x_1 = 1, x_2 = 2, y = 3)$
2. $(x_1 = 2, x_2 = 3, y = 5)$
3. $(x_1 = 3, x_2 = 4, y = 7)$

Initiële Gewichten:
$w_1 = 1$ en $w_2 = 0.5$

Stap 1: Bereken de voorspelde waarden en de kosten:
Voor de initiële gewichten:

Voorbeeld 1:
Voorspelling: $\hat{y}_1 = w_1 \cdot x_1 + w_2 \cdot x_2 = 1 \cdot 1 + 0.5 \cdot 2 = 2$
Kosten: $(2 - 3)^2 = 1$

Voorbeeld 2:
Voorspelling: $\hat{y}_2 = w_1 \cdot x_1 + w_2 \cdot x_2 = 1 \cdot 2 + 0.5 \cdot 3 = 3$
Kosten: $(3 - 5)^2 = 4$

Voorbeeld 3:
Voorspelling: $\hat{y}_3 = w_1 \cdot x_1 + w_2 \cdot x_2 = 1 \cdot 3 + 0.5 \cdot 4 = 4.5$
Kosten: $(4.5 - 7)^2 = 6.25$

Gemiddelde kosten: $\text{MSE} = \frac{1+4+6.25}{3} = 3.0833$

Stap 2: Bereken de gradient van de kostenfunctie:

De partiële afgeleiden van de kostenfunctie ten opzichte van de gewichten zijn:

$\frac{\partial \text{MSE}}{\partial w_1} = \frac{2 \cdot (\hat{y}_1 - y_1) \cdot x_1 + 2 \cdot (\hat{y}_2 - y_2) \cdot x_1 + 2 \cdot (\hat{y}_3 - y_3) \cdot x_1}{3}$

$\frac{\partial \text{MSE}}{\partial w_2} = \frac{2 \cdot (\hat{y}_1 - y_1) \cdot x_2 + 2 \cdot (\hat{y}_2 - y_2) \cdot x_2 + 2 \cdot (\hat{y}_3 - y_3) \cdot x_2}{3}$

Berekenen we deze waarden in:

Voorbeeld 1:

$\frac{\partial \text{MSE}}{\partial w_1} = \frac{2 \cdot (2-3) \cdot 1 + 2 \cdot (3-5) \cdot 1 + 2 \cdot (4.5-7) \cdot 1}{3} = -3.8333$

$\frac{\partial \text{MSE}}{\partial w_2} = \frac{2 \cdot (2-3) \cdot 2 + 2 \cdot (3-5) \cdot 3 + 2 \cdot (4.5-7) \cdot 4}{3} = -9.5$

Stap 3: Werk een updatestap uit:

We gebruiken een leersnelheid $\alpha = 0.01$ en passen de gewichten aan met behulp van de gradient:

Nieuwe $w_1 = w_1 - \alpha \cdot \frac{\partial \text{MSE}}{\partial w_1} = 1 - 0.01 \cdot (-3.8333) = 1.038333$

Nieuwe $w_2 = w_2 - \alpha \cdot \frac{\partial \text{MSE}}{\partial w_2} = 0.5 - 0.01 \cdot (-9.5) = 0.595$

Stap 4: Herhaal de stappen:

We herhalen de stappen 1 tot 3 om verdere gewichtsupdates uit te voeren en de kosten te verlagen.

Door deze stappen te herhalen, kunnen we gradient descent toepassen om de gewichten aan te passen en zo de kosten te minimaliseren.

2e Iteratie

Laten we het proces van gradient descent verder uitwerken en verdere gewichtsupdates uitvoeren om de kosten te verlagen:

Stap 1: Bereken de voorspelde waarden en de kosten:

Voor de huidige gewichten $w_1 = 1.038333$ en $w_2 = 0.595$:

Voorbeeld 1:

Voorspelling: $\hat{y}_1 = 1.038333 \cdot 1 + 0.595 \cdot 2 = 2.228333$

Kosten: $(2.228333 - 3)^2 = 0.5503$

Voorbeeld 2:

Voorspelling: $\hat{y}_2 = 1.038333 \cdot 2 + 0.595 \cdot 3 = 3.261666$

Kosten: $(3.261666 - 5)^2 = 3.0641$

Voorbeeld 3:

Voorspelling: $\hat{y}_3 = 1.038333 \cdot 3 + 0.595 \cdot 4 = 4.294999$

Kosten: $(4.294999 - 7)^2 = 4.7432$

Gemiddelde kosten: $\text{MSE} = \frac{0.5503+3.0641+4.7432}{3} = 2.7859$

Stap 2: Bereken de gradient van de kostenfunctie:

De partiële afgeleiden van de kostenfunctie ten opzichte van de gewichten zijn:

$$\frac{\partial \text{MSE}}{\partial w_1} = \frac{2 \cdot (\hat{y}_1 - y_1) \cdot x_1 + 2 \cdot (\hat{y}_2 - y_2) \cdot x_1 + 2 \cdot (\hat{y}_3 - y_3) \cdot x_1}{3}$$

$$\frac{\partial \text{MSE}}{\partial w_2} = \frac{2 \cdot (\hat{y}_1 - y_1) \cdot x_2 + 2 \cdot (\hat{y}_2 - y_2) \cdot x_2 + 2 \cdot (\hat{y}_3 - y_3) \cdot x_2}{3}$$

Berekenen we deze waarden in:

Voorbeeld 1:

$$\frac{\partial \mathrm{MSE}}{\partial w_1} = \frac{2\cdot(2.228333-3)\cdot 1+2\cdot(3.261666-5)\cdot 1+2\cdot(4.294999-7)\cdot 1}{3} = -2.5588$$

$$\frac{\partial \mathrm{MSE}}{\partial w_2} = \frac{2\cdot(2.228333-3)\cdot 2+2\cdot(3.261666-5)\cdot 3+2\cdot(4.294999-7)\cdot 4}{3} = -7.9709$$

Stap 3: Werk een updatestap uit:

We gebruiken opnieuw een leersnelheid $\alpha = 0.01$ en passen de gewichten aan met behulp van de gradient:

Nieuwe $w_1 = 1.038333 - 0.01 \cdot (-2.5588) = 1.064921$

Nieuwe $w_2 = 0.595 - 0.01 \cdot (-7.9709) = 0.674709$

Stap 4: Herhaal de stappen:

We herhalen deze stappen meerdere keren totdat de kosten niet meer significant veranderen of convergeren naar een minimaal punt. De stopvoorwaarde kan bijvoorbeeld zijn dat de verandering in kosten tussen twee iteraties kleiner is dan een bepaalde tolerantie.

Door deze stappen te herhalen, passen we de gewichten aan om de kosten te verlagen en zo een model te vinden dat de trainingsvoorbeelden goed voorspelt.

Uitleg

Uitleg variabelen

x1 en x2: In het voorbeeld van een huis verkopen stellen x1 en x2 de kenmerken of eigenschappen van het huis voor die worden gebruikt om de verkoopprijs te voorspellen. Specifiek kan x1 de grootte van het huis in vierkante meters voorstellen, terwijl x2 het aantal slaapkamers vertegenwoordigt. Deze kenmerken zijn de inputvariabelen die het model gebruikt om voorspellingen te doen.

y: y vertegenwoordigt de uitvoer van het model, oftewel de voorspelde verkoopprijs van het huis. Dit is wat we proberen te voorspellen met behulp van ons model.

w1 en w2: w1 en w2 zijn de gewichten van het model die de impact van de kenmerken (x1 en x2) op de voorspelde verkoopprijs bepalen. In het voorbeeld van een huis kan w1 bijvoorbeeld de impact van de grootte van het huis (x1) op de verkoopprijs vertegenwoordigen, terwijl w2 de impact van het aantal slaapkamers (x2) op de verkoopprijs vertegenwoordigt. Deze gewichten worden aangepast tijdens het trainingsproces om het model in staat te stellen nauwkeurige voorspellingen te doen.

Uitleg stappen

Stap 1: Bereken de voorspelde waarden en de kosten

We beginnen met het voorspellen van de uitvoerwaarden ($\hat{y}$) voor elk trainingsvoorbeeld met behulp van de huidige gewichten. Vervolgens berekenen we de kosten voor elk voorbeeld door het verschil te nemen tussen de voorspelde waarde en de werkelijke waarde, het kwadraat van dit verschil te nemen, en vervolgens het gemiddelde van deze kwadratische verschillen te berekenen. Dit geeft ons de Mean Squared Error (MSE), een maatstaf voor hoe goed onze huidige gewichten presteren.

Stap 2: Bereken de gradient van de kostenfunctie

De gradient van de kostenfunctie geeft aan hoe snel de kosten veranderen met betrekking tot veranderingen in de gewichten. We berekenen de partiële afgeleiden van de MSE ten opzichte van elk gewicht. Deze afgeleiden vertellen ons in welke richting en hoeveel we elk gewicht moeten aanpassen om de kosten te verlagen.

Stap 3: Werk een update stap uit

We bepalen hoeveel we de gewichten moeten aanpassen door de gradiënten te vermenigvuldigen met een leersnelheid (α), die aangeeft hoe groot de stappen moeten zijn. Vervolgens passen we de gewichten aan door de leersnelheid vermenigvuldigd met de bijbehorende gradient van elke parameter af te trekken van de huidige waarden van de gewichten.

Stap 4: Herhaal de stappen

Door de stappen 1 tot 3 te herhalen, passen we de gewichten iteratief aan om de kosten te verlagen. We blijven dit herhalen totdat we een punt bereiken waarop de kosten niet meer significant dalen of een vooraf ingesteld aantal iteraties hebben bereikt.

Dit iteratieve proces van het aanpassen van de gewichten op basis van de gradients van de kostenfunctie is wat gradient descent in essentie doet. Het uiteindelijke doel is om de gewichten te vinden die leiden tot minimale kosten, waardoor ons model de beste voorspellingen kan doen voor nieuwe gegevens.