**Mesterséges neuron modell**

A mesterséges neuron modell az agyban található biológiai neuronok egyszerűsített matematikai modellje. Egy mesterséges neuron három fő részből áll:

* **Bemenetek:** A bemeneti értékek, amelyek más neuronok kimenetei vagy a hálózat bemeneti adatpontjai.
* **Súlyok:** A bemenetekhez rendelt súlyok, amelyek meghatározzák, mennyire fontosak az adott bemenetek.
* **Összegző és aktivációs függvény:** A bemenetek és súlyok szorzatainak összege egy eltolással (b), amelyet egy aktivációs függvény g(z) segítségével alakítunk át a neuron kimenetére.

**Teljesen összekötött rétegek (Fully Connected Layers)**

A teljesen összekötött rétegben minden bemeneti neuron össze van kötve minden kimeneti neuronnal. A rétegek közötti kapcsolatok súlyai a tanulási folyamat során módosulnak.

Ha egy rétegben n bemeneti neuron és m kimeneti neuron van, akkor a súlyok mátrixa W mérete m×n, és az eltolási vektor b mérete m. A réteg fő komponensei a következők:

* **Bemenetek (x):** Az előző réteg vagy az input adatok kimenetei.
* **Súlyok mátrixa (W):** A bemenetek és a kimenetek közötti kapcsolatokat reprezentáló súlyok. Ha a bemenet mérete n, és a kimenetek száma m, akkor a súlyok mátrixa W mérete m×n.
* **Eltolási vektor (b):** A kimeneti neuronok eltolásai, amelyek egy m hosszú vektor.
* **Aktivációs függvény (g):** A bemenetek és súlyok kombinációjára alkalmazott nemlineáris függvény.

**Hipotézis függvénye**

A teljesen összekötött réteg hipotézis függvénye leírja, hogyan történik a bemenetek átalakítása a kimenetekké a súlyok és eltolások alkalmazásával. Matematikaian a következőképpen néz ki: **z = Wx + b**

**Aktivációs függvények**

Az aktivációs függvények adják a neurális hálózat nemlinearitását, és számos fajtájuk létezik, melyek közül a leggyakrabban használtak:

* **Sigmoid**
* **ReLU**

**Költségfüggvények**

A költségfüggvények mérik a modell előrejelzéseinek és a valós értékek közötti eltéréseket, és segítenek az optimalizációban. A különböző feladatokhoz különböző költségfüggvényeket használunk:

* **Mean Squared Error (MSE):** Regressziós problémák esetén. Az MSE a tényleges és a prediktált értékek közötti négyzetes különbségek átlaga.
* **Binary Cross-Entropy (Log Loss):** Bináris klasszifikáció esetén. Az egyes predikciók logaritmusainak összege.
* **Categorical Cross-Entropy:** Többosztályos klasszifikáció esetén. Az egyes osztályok valószínűségi értékeinek logaritmusainak súlyozott összege.

**Multilayer Perceptron (MLP)**

Az MLP több rétegből álló mesterséges neurális hálózat. Legalább egy rejtett réteggel rendelkezik a bemeneti és kimeneti réteg között. Az MLP felépítése a következő:

* **Bemeneti réteg:** Az adatok bemeneti pontjai.
* **Rejtett rétegek:** Egy vagy több teljesen összekötött réteg, ahol az aktivációs függvények alkalmazása történik.
* **Kimeneti réteg:** Az eredmények előállítására szolgáló réteg.

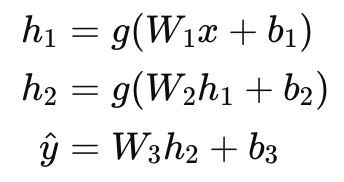
**Paraméterek**

**Súlyok (W):** A rétegek közötti kapcsolatok súlyai.

**Eltolás (b):** A neuronok eltolási értékei.

**Aktivációs függvény (g):** A neurális hálózat nemlinearitását biztosítja.

**Hipotézisfüggvény**

Az MLP hipotézisfüggvénye a bemenetekből a kimenetek előállítását írja le. Például, ha két rejtett réteg van:[](https://github.com/Miki0195/4.Felev/blob/main/Neuronh%C3%A1l%C3%B3k/vizsga/adom5.png)

**Aktivációs függvények**

Az aktivációs függvények adják a neurális hálózat nemlinearitását, és számos fajtájuk létezik, melyek közül a leggyakrabban használtak:

* **Sigmoid**
* **ReLU**

**Költségfüggvények**

A költségfüggvények mérik a modell előrejelzéseinek és a valós értékek közötti eltéréseket, és segítenek az optimalizációban. A különböző feladatokhoz különböző költségfüggvényeket használunk:

* **Mean Squared Error (MSE):** Regressziós problémák esetén. Az MSE a tényleges és a prediktált értékek közötti négyzetes különbségek átlaga.
* **Binary Cross-Entropy (Log Loss):** Bináris klasszifikáció esetén. Az egyes predikciók logaritmusainak összege.
* **Categorical Cross-Entropy:** Többosztályos klasszifikáció esetén. Az egyes osztályok valószínűségi értékeinek logaritmusainak súlyozott összege.

**Skalár És Vektor Alakú Címkék**

**Skalár alakú címke:**

Általában bináris klasszifikációs problémákban használatos. Egyetlen numerikus érték, amely egy osztályt jelöl. Például bináris klasszifikációnál az osztályok lehetnek 0 vagy 1. Egyszerűbb adatstruktúrát eredményez.

**Vektor alakú címke:**

Többdimenziós vektor, amely az osztályokat kódolja. Gyakran "one-hot" vektor formájában, ahol az aktuális osztályhoz tartozó elem 1, a többi 0. Szükséges multi-class klasszifikációs feladatoknál, ahol több kategória van. Ez lehetővé teszi a hálózat számára, hogy egyszerre több osztályt is megkülönböztessen.

**Regresszió**

**Mi az a regresszió?**

A regresszió egy gépi tanulási feladat, amelynek célja egy folytonos kimeneti változó előrejelzése. A regressziós modellek megtanulják a bemeneti és kimeneti változók közötti kapcsolatot, hogy új bemenetek esetén pontos előrejelzéseket adjanak.

**Mire jó a regresszió?**

* **Folytonos értékek előrejelzése:** Például házárak, hőmérséklet, bevételek stb.
* **Kapcsolatok feltárása:** Segít megérteni a bemeneti változók és a kimeneti érték közötti kapcsolatokat.

**Feladatuk:**

* **Lineáris regresszió:** Egyszerű, lineáris kapcsolatokat feltételez a bemenetek és a kimenet között.
* **Nemlineáris regresszió:** Komplexebb modelleket használ, amelyek nemlineáris kapcsolatokat tanulnak meg.

**Klasszifikáció**

**Mi az a klasszifikáció?**

A klasszifikáció egy gépi tanulási feladat, amelynek célja, hogy bemeneti adatok alapján kategóriákat (osztályokat) hozzárendeljen. A klasszifikációs modellek megtanulják a bemeneti változók és az osztályok közötti kapcsolatot, hogy új bemenetek esetén helyes osztályozást végezzenek.

**Mire jó a klasszifikáció?**

* **Kategóriák felismerése:** Például e-mailek spam/nem spam osztályozása, képek kategorizálása stb.
* **Döntéshozatal automatizálása:** Segít automatikusan meghozni döntéseket különböző kategóriák alapján.

**Feladatuk:**

* **Bináris klasszifikáció:** Két osztály közötti döntést hoz (pl. igen/nem).
* **Multi-class klasszifikáció:** Több osztály közötti döntést hoz (pl. képek kategorizálása különböző objektumok alapján).

**Regresszió, Bináris- és Multi-class Klasszifikáció**

**Regresszió**

* **MLP:** A regressziós feladatok esetében az MLP az input változókból készít egy előrejelzést a folytonos kimeneti változó értékére. A kimeneti réteg általában egyetlen neuronból áll, amely kimenetként egy folytonos értéket ad meg.
* **Teljesen összekötött rétegek:** A teljesen összekötött rétegek hasonlóan működnek, mint az MLP, ahol minden bemeneti csomópont kapcsolódik minden kimeneti csomóponthoz. A kimeneti réteg egyetlen neuronból áll, amely a regressziós probléma megoldásához szükséges kimeneti értéket adja meg.

**Bináris Klasszifikáció**

* **MLP:** Bináris klasszifikáció esetén az MLP kimeneti rétege egyetlen neuronból áll, amelynek a kimenete egy 0 vagy 1 érték, ami megfelel a két osztály (például pozitív és negatív) valószínűségének.
* **Teljesen összekötött rétegek:** Ugyanaz a helyzet, mint az MLP esetében, ahol a kimeneti réteg egyetlen neuronból áll, amely kimenetként egy 0 vagy 1 értéket ad meg.

**Multi-Class Klasszifikáció**

* **MLP:** A multi-class klasszifikáció esetében az MLP kimeneti rétege több neuronból áll, ahol minden neuron egy adott osztályhoz tartozik. Az egyes neuronok kimenetei a különböző osztályokhoz tartozó valószínűségeket reprezentálják.
* **Teljesen összekötött rétegek:** A teljesen összekötött rétegek hasonlóképpen működnek, mint az MLP esetében, ahol a kimeneti réteg több neuronból áll, ahol minden neuron egy adott osztályhoz tartozik.

**MLP És Teljesen Összekötött Rétegek Kapcsolata**

Az MLP (Multilayer Perceptron) és a teljesen összekötött réteg gyakorlatilag ugyanazt a neurális hálózati architektúrát jelentik. Az MLP egy olyan neurális háló, amely több rétegű, ahol minden rétegben minden neuron kapcsolódik az előző réteg összes neuronjához. Egy teljesen összekötött réteg (vagy más néven dense réteg) pontosan ezt a kapcsolatot valósítja meg.

*Másképpen: MLP és a teljesen összekötött réteg közötti kapcsolat az, hogy az MLP egy olyan neurális hálózati architektúra, amely a teljesen összekötött rétegeket használja az információ továbbítására és feldolgozására. A teljesen összekötött rétegek alkotják az MLP strukturális alapját, és lehetővé teszik a neurális hálózatok nagyobb szabadságát a tanulás és az információfeldolgozás terén.*