

PYTHON DASTURLASH TILI YORDAMIDA BIR QATLAMLI PERSEPTRONNI O‘QITISH VA TASNIFLASH MISOLIDA TAHLIL QILISH.

Tojimamatov Isroil Nurmamatovich

Farg‘ona davlat universiteti

Amaliy matematika katta o‘qituvchisi

E-mail: isroiltojimatov@gmail.com

Raximova Gulsanam Nabijon qizi

Farg‘ona davlat universiteti talabasi

E-mail: abdurasulovagulsanam16@gmail.com

Annotatsiya. Ushbu maqolada Python dasturlash tili yordamida bir qatlamli perseptron modelini qurish, o‘qitish va uni tasniflash masalasida qo‘llash jarayoni ko‘rib chiqiladi. Perseptron algoritmining ishlash prinsipi, o‘qitish bosqichlari hamda o‘rganish tezligining natijalarga ta’siri tahlil qilinadi. Amaliy tajriba orqali modelning ma’lumotlarni chiziqli ajratish imkoniyatlari baholanadi. Maqola natijalari sun’iy intellekt va mashinaviy o‘rganish sohalarida boshlang‘ich neyron tarmoq modellarining ahamiyatini ko‘rsatadi.

Kalit so‘zlar: bir qatlamli perseptron, Python, neyron tarmoq, tasniflash, o‘qitish algoritmi, mashinaviy o‘rganish.

Annotation. This article explores the process of building, training, and applying a single-layer perceptron model using the Python programming language for classification tasks. The working principle of the perceptron algorithm, training stages, and the influence of the learning rate on results are analyzed. Through practical experimentation, the model’s ability to linearly separate data is evaluated. The findings highlight the importance of simple neural network models in artificial intelligence and machine learning fields.

Keywords: single-layer perceptron, Python, neural network, classification, training algorithm, machine learning.

Аннотация. В данной статье рассматривается процесс построения, обучения и применения однослойного персептрона с использованием языка программирования Python для решения задачи классификации. Анализируются принцип работы

алгоритма персептрона, этапы обучения и влияние скорости обучения (learning rate) на результаты. На основе практического эксперимента оцениваются возможности модели по линейному разделению данных. Результаты показывают значение простых нейронных сетей в областях искусственного интеллекта и машинного обучения.

Ключевые слова: однослойный персептрон, Python, нейронная сеть, классификация, алгоритм обучения, машинное обучение

KIRISH

So‘nggi yillarda sun‘iy intellekt va mashinaviy o‘rganish texnologiyalari jadal sur‘atlarda rivojlanib, turli sohalarda keng qo‘llanilmoqda. Neyron tarmoqlar bu yo‘nalishning eng muhim tarkibiy qismlaridan biri hisoblanadi. Ayniqsa, ularning soddalashtirilgan modeli — bir qatlamli perseptron — mashinaviy o‘rganishning nazariy asoslarini o‘rganishda muhim ahamiyatga ega.

Bir qatlamli perseptron modeli ilk bor 1958-yilda amerikalik olim Frank Rosenblatt tomonidan taklif etilgan bo‘lib, u chiziqli ajratiladigan ma‘lumotlarni tasniflash uchun mo‘ljallangan edi. Mazkur modelning asosiy g‘oyasi kiruvchi signallarni og‘irlik koeffitsientlari yordamida qayta ishlash va natijada chiqish qarorini aniqlashdan iboratdir.

Bugungi kunda zamonaviy dasturlash tillari, xususan Python, neyron tarmoqlarni amaliy jihatdan sinab ko‘rish va o‘rganish uchun qulay vositalarni taqdim etadi. Pythonning NumPy, Matplotlib va scikit-learn kabi kutubxonalari yordamida perseptron modelini tezkor qurish, o‘qitish va tahlil qilish imkoniyati mavjud.

Ushbu maqolada Python dasturlash tili yordamida bir qatlamli perseptron modeli quriladi, o‘qitiladi va tasniflash misolida sinovdan o‘tkaziladi. Shuningdek, modelning o‘rganish tezligi (learning rate), iteratsiyalar soni va og‘irliklarning tasodifiy boshlang‘ich qiymatlarining natijaga ta‘siri tahlil qilinadi. Maqolaning maqsadi — bir qatlamli perseptronning ishlash tamoyillarini amaliy tajriba asosida ko‘rsatish hamda uning mashinaviy o‘rganishdagi o‘rni va cheklovlarini yoritishdir.

ASOSIY QISM

Bir qatlamli perseptron — bu kiruvchi ma'lumotlarni chiziqli kombinatsiya shaklida qayta ishlaydigan va chiqishda faollashtiruvchi (aktivatsiya) funksiya orqali qaror qabul qiladigan eng sodda sun'iy neyron modeli hisoblanadi.

Perseptronning asosiy matematik ifodasi quyidagicha yoziladi:

$$y=f(\sum_{i=1}^n w_i x_i + b$$

Bu yerda:

x_i — kirish signallari

w_i — mos og'irlik koeffitsientlari

b — siljish (bias) qiymati

$f(\cdot)$ — aktivatsiya funksiyasi (odatda signum yoki sigmoid)

y — chiqish qiymati (tasnif natijasi).

O'qitish jarayonida perseptron o'z og'irliklarini quyidagi qoidaga ko'ra yangilab boradi:

$$w_i^{new} = w_i^{old} + \eta(t - y)x_i$$

Bu algoritim chiziqli ajratiluvchi ma'lumotlar uchun konvergensiyani kafolatlaydi, ya'ni ma'lumotlar to'plami chiziq bilan ajratilsa, perseptron har doim to'g'ri yechimni topadi.

Ushbu tajribada NumPy kutubxonasidan foydalanilib, oddiy ikki sinfli (0 va 1) tasniflash vazifasi bajariladi. Quyidagi dastur kodida modelning asosiy bosqichlari ko'rsatilgan.

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
X = np.array([
```

```
    [0, 0],
```

```
    [0, 1],
```

```
    [1, 0],
```

```
    [1, 1]
```

])

```
y = np.array([0, 0, 0, 1])
```

Quyida kod qismida model parametrlari va funksiyalarni aniqlash korsatilgan:

```
def activation(x):
```

```
    return np.where(x >= 0, 1, 0)
```

```
weights = np.random.rand(2)
```

```
bias = np.random.rand(1)
```

```
learning_rate = 0.1
```

Keyingi bosqichda o‘qitish jarayoni amalga oshiriladi.

```
    for epoch in range(20):
```

```
        for i in range(len(X)):
```

```
            linear_output = np.dot(X[i], weights) + bias
```

```
            y_pred = activation(linear_output)
```

```
            error = y[i] - y_pred
```

```
            weights += learning_rate * error * X[i]
```

```
            bias += learning_rate * error
```

Kod oxirida natija tekshiriladi.

```
print("O‘qitilgan og‘irliklar:", weights)
```

```
print("Bias:", bias)
```

```
for i in range(len(X)):
```

```
    linear_output = np.dot(X[i], weights) + bias
```

```
    y_pred = activation(linear_output)
```

```
    print(f"Kirish: {X[i]}, Natija: {y_pred}")
```

Tajriba natijasida perseptron AND mantiqiy amalini muvaffaqiyatli o‘rganadi:

$$(0,0) \rightarrow 0$$
$$(0,1) \rightarrow 0$$
$$(1,0) \rightarrow 0$$
$$(1,1) \rightarrow 1$$

Model chiziqli ajratiluvchi ma'lumotlar uchun yuqori aniqlikka erishadi. Biroq, XOR kabi chiziqli ajratilmaydigan vazifalarda bir qatlamli perseptron samarali ishlamaydi. Bu holat neyron tarmoqlarda ko'p qatlamli arxitekturalar zarurligini ko'rsatadi.

Shu tarzda, Python dasturlash muhiti bir qatlamli perseptronning ishlash tamoyillarini chuqur tahlil qilish va vizuallashtirish uchun qulay vosita sifatida namoyon bo'ladi.

XULOSA

Yuqorida o'tkazilgan tajriba va tahlillar shuni ko'rsatadiki, bir qatlamli perseptron modeli mashinaviy o'rganishning eng sodda, ammo muhim nazariy asoslaridan biridir. Model chiziqli ajratiluvchi ma'lumotlar to'plamida yuqori samaradorlik bilan ishlaydi va tasniflash jarayonini muvaffaqiyatli amalga oshiradi.

Python dasturlash tili va uning kutubxonalari (NumPy, Matplotlib) perseptron algoritmini amaliy jihatdan tez va qulay tarzda amalga oshirish imkonini beradi. O'rganish tezligi (learning rate), iteratsiyalar soni va og'irliklarning dastlabki qiymatlari modelning o'qitilish jarayonida muhim rol o'ynaydi.

Biroq, bir qatlamli perseptronning asosiy cheklovi shundaki, u chiziqli ajratilmaydigan ma'lumotlarni to'g'ri tasniflay olmaydi. Shu sababli, ko'p qatlamli neyron tarmoqlar (MLP) va murakkab aktivatsiya funksiyalarining joriy etilishi neyron tarmoqlarning keyingi avlodini shakllantirishda muhim bosqich bo'lgan.

Umuman olganda, ushbu maqolada ko'rib chiqilgan usul mashinaviy o'rganishning boshlang'ich bosqichlarini o'rganish, neyron tarmoqlarning ishlash mexanizmini tushunish hamda Python tilida amaliy ko'nikma hosil qilish uchun foydali namuna hisoblanadi.

Foydalanilgan adabiyotlar

1. Rosenblatt, F. (1958). *The Perceptron: A probabilistic model for information storage and organization in the brain*. Psychological Review, 65(6), 386–408.
2. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
3. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

4. Geron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.
5. Raschka, S., & Mirjalili, V. (2017). *Python Machine Learning*. Packt Publishing.
6. O'zbekiston Respublikasi Raqamli texnologiyalar vazirligi. (2023). *Sun'iy intellekt asoslari bo'yicha o'quv qo'llanma*. Toshkent: IT Park nashriyoti.