Matris Manipülasyonu, Özdeğerler ve Özvektörler ile Makine Öğrenmesi Arasındaki İlişki

Matris: Sayılarla doldurulmuş dikdörtgen biçimli tablolardır. Genellikle veri temsili, dönüşüm işlemleri ve doğrusal sistemlerin gösteriminde kullanılır.

Özdeğer: Bir kare matrisin, belirli bir vektörü yalnızca ölçeklendirerek değiştirdiği durumlarda ortaya çıkan skaler değerdir. Matematiksel olarak, Av=λv eşitliğini sağlayan λ değeridir.

Özvektör: Bir matrisin uyguladığı dönüşüm sonucu sadece ölçeklenen, yönü değişmeyen vektördür. Yukarıdaki eşitlikte v olarak gösterilir.

Makine öğrenmesinde veriler genel olarak matrisler halinde temsil edilir. Bu veriler üzerinde çalışıldığında ise matris manipülasyonu kaçınılmaz hale gelir.

1.Lineer Modellerde yani lineer regresyon, lojistik regresyon, SVM gibi algoritmalarda tüm hesaplamalar matris çarpımları ve transpoz işlemleriyle gerçekleşir.

2.Yüksek boyutlu veriyi daha düşük boyutlara indirme (Dimesionalty Reduction) hesaplamaları hızlandırmak ve anlamlı öznitelikler çıkarmak için PCA gibi yöntemler kullanılır. PCA yöntemi ise kovaryans matrisin özdeğer ve özvektörlerini hesaplayarak çalışır.

3.Derin öğrenmede bulunan sinir ağlarındaki her katman girdi matrisiyle ağırlık matrisini çarparak manipüle eder.

5.K-means algoritmasında örneklerle merkezler arası uzaklıklar hesaplanır. Spektral Kümeleme algoritmasında ise benzerlik matrisi oluşturulur ve Laplace matrisinin özvektörleri kullanılarak kümeler belirlenir.

6.Doğal dil işleme algoritmasında kelime vektörleri kullanılır ve cümle benzerliği matris çarpımlarıyla hesaplanır.

**Kaynakça**

<https://odsc.medium.com/intro-to-vectors-and-matrices-in-machine-learning-4904be73d251>

<https://www.geeksforgeeks.org/ml-linear-algebra-operations/>

<https://www.trybackprop.com/blog/linalg101/part_1_vectors_matrices_operations>

Numpy.linalg.eig Fonksiyonunun İncelemesi

“eig” fonksiyonu özdeğer ve özvektör hesaplama işini gerçek sayılar için LAPACK’taki dgeev fonksiyonuna, karmaşık sayılar için zgeev fonksiyonuna devreder. “dgeev” fonksiyonu, bir kare matrisin hm özdeğerlerini hem de sağ özvektörlerini hesaplar. İsteğe bağlı sol vektörleri de hesaplayabilse de Numpy bunu yapmaz.

“eig” fonksiyonu doğrudan karakteristik çözüm yapmak yerine Schur ayrışımını kullanır.

A=QTQT

* A: Girdi matrisi
* Q: Ortonormal matris (dönüşüm matrisidir)
* T: Üst üçgensel matris (Schur formu)

T matrisinin köşegenindeki elemanlar A matrisinin özdeğerleridir. Schur ayrışımından sonra sağ özvektörler, bu dönüşüm Q kullanılarak şöyle hesaplanır:

Av=λv⇒(A−λI)v=0

Bu denklem çözülerek özvektörlere ulaşılır.

Bu yöntemin kullanılması karakteristik polinom kullanılan çözüme göre daha avantajlıdır çünkü büyük matrislerde daha az veri kaybı yaşatır.