

Makine öğrenimi sınıflandırma algoritmaları, veri setindeki özellikleri kullanarak veriyi belirli sınıflara veya kategorilere ayırmak için kullanılan algoritmalar. Temel amaç, veri noktalarını farklı sınıflara doğru bir şekilde atamaktır, böylece yeni veri noktalarının hangi sınıfa ait olduğunu tahmin etmek mümkün olur.

Sınıflandırma algoritmaları, genellikle denetimli öğrenme yöntemleri olarak çalışırlar, yani eğitim verisi etiketlenmiş (yani doğru sınıflara atanmış) veri içerir. Bu etiketler, algoritmanın doğru sınıflandırmayı öğrenmesine yardımcı olur.

Popüler sınıflandırma algoritmaları şunları içerir:

- 1- Destek Vektör Makineleri (Support Vector Machines - SVM): SVM, veri noktalarını sınıflandırmak için bir hiperdüzlem oluşturur. Bu hiperdüzlem, sınıflar arasındaki marjinal mesafeyi maksimize eder ve veri noktalarını bu marjinin iki tarafına yerleştirir.
- 2- Karar Ağaçları (Decision Trees): Karar ağaçları, bir dizi karar düğümü ve sonuçları olan bir ağaç yapısı kullanarak veri setini sınıflandırır. Her karar düğümü, belirli bir özelliğe dayalı olarak veri noktalarını böler.
- 3- Rastgele Ormanlar (Random Forests): Rastgele ormanlar, birçok karar ağacının bir araya gelmesiyle oluşur. Her bir ağaç, rastgele örneklenmiş bir alt küme üzerinde eğitilir ve genellikle daha iyi genelleme yeteneğine sahip olur.
- 4- Lojistik Regresyon (Logistic Regression): Lojistik regresyon, bir girdi değişkeninin belirli bir kategoriye ait olma olasılığını tahmin

etmek için kullanılır. İki sınıflı (binary) ve çok sınıflı (multiclass) sınıflandırma problemlerinde kullanılabilir.

- 5- K-En Yakın Komşu (K-Nearest Neighbors - k-NN): k-NN, bir veri noktasını sınıflandırmak için etiketli eğitim verisi içindeki en yakın k komşusunu kullanır. En sık kullanılan etiketi tahmin etmek için komşuların etiketleri kullanılır.

Bu sınıflandırma algoritmaları, farklı veri yapıları ve problemler için uygun olabilir. Hangi algoritmanın seçileceği, veri setinin özelliklerine, problem tipine ve performans ölçütlerine bağlı olarak değişir. Her bir algoritmanın avantajları ve dezavantajları vardır, bu nedenle uygulanacak algoritmanın özellikleri dikkatlice değerlendirilmelidir.

Sınıflandırma kodu :

```
from sklearn import datasets
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score

# Veri setini yükleme
iris = datasets.load_iris()
X = iris.data
y = iris.target

# Veriyi eğitim ve test setlerine ayırma
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Veri ön işleme: özellik ölçeklendirme
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

# SVM modelini oluşturma ve eğitme
classifier = SVC(kernel='linear', random_state=42) # Lineer çekirdek
kullanılıyor
```

```
classifier.fit(X_train, y_train)
```

```
# Test seti üzerinde tahmin yapma
```

```
y_pred = classifier.predict(X_test)
```

```
# Doğruluk skorunu hesaplama
```

```
accuracy = accuracy_score(y_test, y_pred)
```

```
print("Doğruluk:", accuracy)
```