Hamdi Utku Paralı 19253510

1. Makine öğrenmesi algoritmaları ile ilgili iş görüşmelerinde sorulan ileri seviye sorular ve cevapları nelerdir?

ChatGPT: Makine öğrenmesi algoritmalarıyla ilgili iş görüşmelerinde sorulan ileri seviye sorular çeşitli olabilir. İşte bazı olası sorular ve cevapları:

1. Derin öğrenme nedir ve nasıl çalışır?
   * Derin öğrenme, yapay sinir ağları üzerine kurulu bir makine öğrenmesi yaklaşımıdır. Bir veri setindeki karmaşık ilişkileri otomatik olarak çıkarmak için çok katmanlı sinir ağları kullanır. Verileri giriş katmanına sağlar, ardından gizli katmanlardan geçerek çıktı katmanına ulaşır. Derin öğrenme, genellikle büyük veri setlerine ve yüksek hesaplama gücüne ihtiyaç duyar.
2. Genelleyici model ile aşırı öğrenme arasındaki fark nedir?
   * Genelleyici model, bir veri setindeki desenleri doğru bir şekilde tanımlayabilen ve yeni verilere de uygulanabilen bir modeldir. Aşırı öğrenme ise eğitim verilerine aşırı derecede uyum sağlayan ancak yeni verilere genelleme yapmakta zorlanan bir modeldir. Aşırı öğrenme, modelin eğitim verilerindeki gürültü veya anlamsız desenlerin yanı sıra veri setindeki örneklerin ezici çoğunluğunu öğrenme eğilimi gösterir.
3. Desteğe dayalı vektör makineleri (Support Vector Machines, SVM) nedir ve nasıl çalışır?
   * SVM, sınıflandırma ve regresyon problemleri için kullanılan bir makine öğrenmesi algoritmasıdır. Verileri bir uzayda temsil eder ve bu uzayda en iyi şekilde ayrılabilen bir hiper düzlem bulmaya çalışır. Sınıfları ayırmak için optimum hiper düzlemi belirlerken, marjinal aralığı (margin) maksimize etmeye çalışır. SVM, kernel fonksiyonları aracılığıyla non-lineer problemleri de çözebilir.
4. Karar ağaçları nasıl çalışır ve avantajları nelerdir?
   * Karar ağaçları, veri setindeki özniteliklerin değerlerine göre yapılan kararlarla sınıflandırma veya regresyon yapmak için kullanılan bir algoritmadır. Karar ağaçları, veri setini bölerek ve karar düğümlerindeki koşulları değerlendirerek ağacın dallarında ilerler. Avantajları arasında anlaşılırlık, yüksek hız, eksik verilerle başa çıkma yeteneği ve hem sınıflandırma hem de regresyon problemleri için uygulanabilme özelliği bulunur.
5. Random Forest (Rastgele Ormanlar) algoritması nedir?
6. Derin öğrenmede Gated Recurrent Unit nedir?

ChatGPT: Gated Recurrent Unit (GRU), derin öğrenme alanında kullanılan bir rekürsif sinir ağı (RNN) hücresidir. GRU, özellikle zaman serisi verileri gibi ardışık veri yapılarının analizi için etkili bir şekilde kullanılabilir.

GRU, bir RNN hücresinin geliştirilmiş bir versiyonudur ve aşağıdaki anahtar bileşenlere sahiptir:

1. Hafıza Hücresi (Memory Cell): GRU'nun hafıza hücresi, geçmiş bilgileri kodlar ve önceki adımlardan gelen bilgileri depolar.
2. Sıfırlama Ağı (Reset Gate): Sıfırlama ağı, hafıza hücresinin hangi bilgilerin silineceğine karar verir. Geçmiş bilgilerin bir kısmının unutulmasına izin verir.
3. Güncelleme Ağı (Update Gate): Güncelleme ağı, hangi bilgilerin hafıza hücresine eklenip güncelleneceğine karar verir. Yeni bilgilerin eklenmesiyle hafıza güncellenir.

GRU'nun avantajlarından bazıları şunlardır:

* Vanishing Gradient Sorunu: GRU, sıfırlama ve güncelleme ağları sayesinde vanishing gradient sorununu azaltır. Bu sayede daha uzun süreli bağımlılıkları modellenebilir ve daha iyi sonuçlar elde edilebilir.
* Hafıza Etkisi: GRU, hafıza hücresi sayesinde önceki adımlardan gelen bilgileri tutabilir ve bu bilgileri gelecekteki adımlarda kullanabilir. Bu özellik, özellikle zaman serisi verileri gibi ardışık verilerin analizinde önemlidir.
* Parametre Verimliliği: GRU, LSTM (Long Short-Term Memory) gibi başka bir RNN hücresine göre daha az parametreye sahiptir. Bu, daha hızlı eğitim ve daha az bellek tüketimi anlamına gelir.

GRU, doğal dil işleme, konuşma tanıma, makine çevirisi gibi pek çok görevde başarıyla kullanılan bir derin öğrenme bileşenidir.

1. DOĞRUSAL REGRESYON ALGORİTMASININ GÜÇLÜ VE ZAYIF YÖNLERİ NELERDİR?

ChatGPT: Doğrusal regresyon, makine öğrenmesi ve istatistiksel analizde yaygın olarak kullanılan bir algoritmadır. İşte doğrusal regresyonun güçlü ve zayıf yönlerinin bir özeti:

Güçlü Yönleri:

1. Basit ve Yorumlanabilir: Doğrusal regresyon, veriler arasındaki ilişkiyi anlamak için basit ve yorumlanabilir bir model sunar. Katsayılar doğrudan yorumlanabilir ve değişkenlerin çıktı üzerindeki etkilerini anlamak kolaydır.
2. Hızlı ve Hesaplama İşlemi Dostu: Doğrusal regresyon, hesaplama açısından hızlıdır ve büyük veri setleriyle başa çıkmak için uygundur. Modelin eğitimi ve tahminleri, hesaplama süresi açısından genellikle hızlıdır.
3. İyi Başlangıç Noktası: Doğrusal regresyon, diğer daha karmaşık algoritmaların başlangıç noktası olarak kullanılabilir. Daha karmaşık modellerin performansını değerlendirmek ve karşılaştırmak için temel bir referans noktası sağlar.

Zayıf Yönleri:

1. Lineer İlişkileri Sınırlar: Doğrusal regresyon, yalnızca bağımlı değişken ile bağımsız değişkenler arasındaki doğrusal ilişkileri yakalamak için kullanılır. Eğer veri setinde non-lineer ilişkiler varsa, doğrusal regresyon bu ilişkileri doğru bir şekilde modelleyemez.
2. Aşırı Öğrenme Riski: Doğrusal regresyon, eğitim veri setine aşırı uyum sağlama (aşırı öğrenme) eğilimindedir. Eğer model aşırı öğrenirse, yeni verilere genelleme yapmakta zorlanabilir ve tahminlerde hata yapabilir.
3. Aykırı Değerlere Hassaslık: Doğrusal regresyon, aykırı değerlere hassas olabilir. Aykırı değerler, modelin katsayılarını ve çıktıyı etkileyebilir, bu da modelin doğruluğunu olumsuz yönde etkileyebilir.
4. Bağımlılık Varsayımı: Doğrusal regresyon, hata terimleri arasında bağımsızlık ve homoscedasticity (sabit varyans) gibi varsayımlara dayanır. Bu varsayımlar gerçek veri setlerinde her zaman geçerli olmayabilir ve bu durum modelin performansını etkileyebilir.

Doğrusal regresyonun güçlü ve zayıf yönlerini göz önünde bulundurarak, veri setinizin özelliklerine ve analiz yapmak istediğiniz sorulara bağlı olarak doğru algoritmayı

1. Makine Öğrenmesinde K-means algoritmasının daha iyi sonuç verebilmesi için ne yapmalı?

ChatGPT: K-means algoritması, bir kümeleme algoritması olduğu için, daha iyi sonuçlar elde etmek için bazı stratejileri takip etmek önemlidir. İşte K-means algoritmasının daha iyi sonuç vermesi için dikkate almanız gereken bazı yöntemler:

1. Veri Ön İşleme: Veri ön işleme adımları, K-means algoritmasının performansını artırabilir. Aykırı değerleri tespit etmek ve ele almak, eksik verileri doldurmak, özellik ölçeklendirme ve normalizasyon gibi adımları uygulamak veri setinizin daha uygun şekilde temsil edilmesini sağlar.
2. Özellik Seçimi ve Azaltma: K-means algoritması, veri setindeki tüm öznitelikleri kullanır. Ancak, bazen gereksiz veya çok benzer öznitelikler, algoritmanın performansını olumsuz etkileyebilir. Özellik seçimi veya boyut azaltma yöntemlerini kullanarak, veri setindeki önemli ve temsilci öznitelikleri belirleyebilir ve bu şekilde daha iyi sonuçlar elde edebilirsiniz.
3. Başlangıç Noktası Seçimi: K-means algoritması, başlangıç noktalarının seçimine duyarlıdır. İyi bir başlangıç noktası seçimi, algoritmanın daha hızlı ve daha iyi sonuçlara ulaşmasını sağlayabilir. Farklı başlangıç noktası seçim yöntemlerini deneyerek daha iyi sonuçlar elde edebilirsiniz.
4. K Değerinin Doğru Seçimi: K-means algoritması, küme sayısı (K değeri) tarafından belirlenir. Doğru K değerini seçmek, algoritmanın performansını etkileyebilir. Elbow yöntemi, Silhouette analizi gibi yöntemleri kullanarak doğru K değerini belirlemek için deneme yapabilirsiniz.
5. İterasyon Sayısı: K-means algoritması, belirli bir iterasyon sayısı kadar çalışır. Daha fazla iterasyon yapmak, algoritmanın daha iyi sonuçlar elde etme olasılığını artırabilir. Ancak, aşırı iterasyon da hesaplama süresini uzatabilir. İterasyon sayısını deneyerek en uygun değeri belirleyebilirsiniz.
6. Sonuçların Değerlendirilmesi: Elde edilen kümeleme sonuçlarını değerlendirmek önemlidir. Farklı metrikler kullanarak (örneğin, küme içi varyans, Silhouette skoru, Davies-Bouldin endeksi), kümelerin birbirinden ne kadar farklı olduğunu ve iç tutarlılığını değerlendirebilirsiniz. Bu şekilde, algoritmanın daha iyi sonuçlar vermesi için gerekli düzeltmeleri yapabilirsiniz.

Yukarıdaki stratejileri takip ederek, K-means algoritmasının performansını artırabilir ve daha iyi sonuçlar elde edebilirsiniz. Ancak unutmayın ki K-means, veri setinin doğal olarak kümelere ayrılabilen veriler için daha uygun olduğu durumlarda daha iyi sonuçlar verir.