VERİ MADENCİLİĞİ
Temel Sınıflandırma Yöntemleri

Yrd. Doç. Dr. Şule Gündüz Öğüdücü
www.cs.itu.edu.tr/~gunduz/courses/verimaden/

Konular

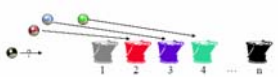
- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

Sınıflandırma

- Sınıflandırma (classification) problemi:
  - nesnelerden oluşan veri kümesi (öğrenme kümesi): $D=\{t_1,t_2,...,t_n\}$
  - her nesne niteliklerden oluşuyor, niteliklerden biri sınıf bilgisi
- Sınıf niteliğini belirlemek için diğer nitelikleri kullanarak bir model bulma
- Öğrenme kümesinde yer almayan nesneleri (sınama kümesi) mümkün olan en iyi şekilde doğru sınıflara atamak
- sınıflandırma=ayrık değişkenler için öngörüde (prediction) bulunma

Sınıflandırma

- Amaç: Bir niteliğin değerini diğer nitelikleri kullanarak belirlemek
  - verinin dağılımına göre bir model bulunur
  - bulunan model, başarımı belirlendikten sonra niteliğin gelecekteki ya da bilinmeyen değerini tahmin etmek için kullanılır
    - model başarımı: doğru sınıflandırılmış sınama kümesi örneklerinin oranı
- Veri madenciliği uygulamasında:
  - ayrık nitelik değerlerini tahmin etmek: sınıflandırma
  - sürekli nitelik değerlerini tahmin etmek: öngörü



- Sınıflandırma: hangi topun hangi sepete koyulabileceği
- Öngörü: Topun ağırlığı

Gözetimli & Gözetimsiz Sınıflandırma

- Gözetimli (Supervised) sınıflandırma = sınıflandırma
  - Sınıfların sayısı ve hangi nesnenin hangi sınıfta olduğu biliniyor.

- Gözetimsiz (Unsupervised) sınıflandırma = demetleme (clustering)
  - Hangi nesnenin hangi sınıfta olduğu bilinmiyor. Genelde sınıf sayısı bilinmiyor.

Konular

- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

Sınıflandırma Uygulamaları

- Kredi başvurusu değerlendirme
- Kredi kartı harcamasının sahtekarlık olup olmadığına karar verme
- Hastalık teşhisi
- Ses tanıma
- Karakter tanıma
- Gazete haberlerini konularına göre ayırma
- Kullanıcı davranışları belirleme

Sınıflandırma için Veri Hazırlama

- Veri dönüşümü:
  - Sürekli nitelik değeri ayrık hale getirilir
  - Normalizasyon ([-1,...,1], [0,...,1])
- Veri temizleme:
  - gürültüyü azaltma
  - gereksiz nitelikleri silme

Sınıflandırma İşlemi

- Sınıflandırma işlemi üç aşamadan oluşur:
  1. Model oluşturma
  2. Model değerlendirme
  3. Modeli kullanma

Sınıflandırma İşlemi: Model Oluşturma

1. Model Oluşturma:
   - Her nesnenin sınıf etiketi olarak tanımlanan niteliğinin belirlediği bir sınıfta olduğu varsayılır
   - Model oluşturmak için kullanılan nesnelerin oluşturduğu veri kümesi öğrenme kümesi olarak tanımlanır
   - Model farklı biçimlerde ifade edilebilir
     - IF – THEN – ELSE kuralları ile
     - Karar ağaçları ile
     - Matematiksel formüller ile

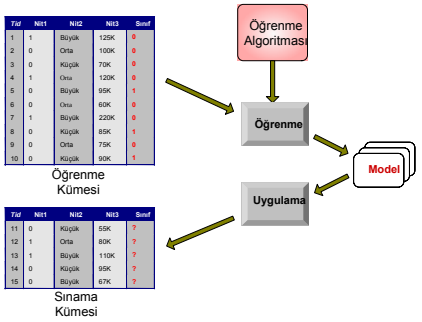
Sınıflandırma İşlemi: Model Değerlendirme

2. Model Değerlendirme:
   - Modelin başarımı (doğruluğu) sınama kümesi örnekleri kullanılarak belirlenir
   - Sınıf etiketi bilinen bir sınama kümesi örneği model kullanılarak belirlenen sınıf etiketiyle karşılaştırılır
   - Modelin doğruluğu, doğru sınıflandırılmış sınama kümesi örneklerinin toplam sınama kümesi örneklerine oranı olarak belirlenir
   - Sınama kümesi model öğrenirken kullanılmaz

Sınıflandırma Işlemi: Modeli Kullanma

3. Modeli kullanma:
   - Model daha önce görülmemiş örnekleri sınıflandırmak için kullanılır
     - Örneklerin sınıf etiketlerini tahmin etme
     - Bir niteliğin değerini tahmin etme

Örnek



Sınıflandırıcı Başarımını Değerlendirme

- Doğru sınıflandırma başarısı
- Hız
  - modeli oluşturmak için gerekli süre
  - sınıflandırma yapmak için gerekli süre
- Kararlı olması
  - veri kümesinde gürülltülü ve eksik nitelik değerleri olduğu durumlarda da iyi sonuç vermesi
- Ölçeklenebilirlik
  - büyük miktarda veri kümesi ile çalışabilmesi
- Anlaşılabilir olması
  - kullanıcı tarafından yorumlanabilir olması
- Kuralların yapısı
  - birbiriyle örtüşmeyen kurallar

Konular

- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

Sınıflandırma Yöntemleri

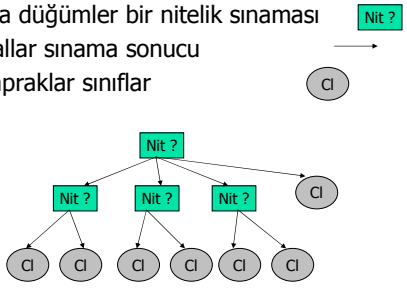
- Karar ağaçları (decision trees)
- Yapay sinir ağları (artificial neural networks)
- Bayes sınıflandırıcılar (Bayes classifier)
- İlişki tabanlı sınıflandırıcılar (association-based classifier)
- k-en yakın komşu yöntemi (k- nearest neighbor method)
- Destek vektör makineleri (support vector machines)
- Genetik algoritmalar (genetic algorithms)
- ...

Konular

- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

Karar Ağaçları

- Akış diyagramı şeklinde ağaç yapısı
  - Ara düğümler bir nitelik sınaması
  - Dallar sınama sonucu
  - Yapraklar sınıflar



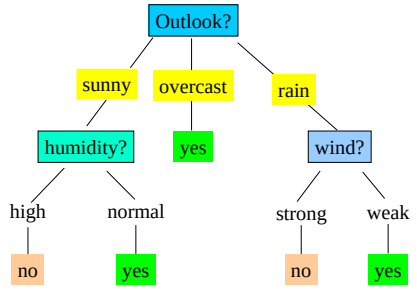
Örnek: Karar Ağacı

- J. Ross Quinlan'ın geliştirdiği ID3 modeline uyarlanmış:
  - hava tenis oynamaya uygun mu?

Day	Outlook	Temperature	Humidity	Wind	Play ball
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Hava durumu Verisi

Örnek: Karar Ağacı



Karar Ağacı Yöntemleri

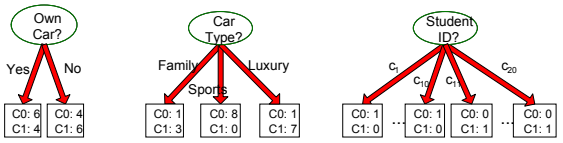
- Karar ağacı oluşturma yöntemleri genel olarak iki aşamadan oluşur:
  1. ağaç oluşturma
     - en başta bütün öğrenme kümesi örnekleri kökte
     - seçilen niteliklere bağlı olarak örnek yinelemeli olarak bölünüyor
  2. ağaç budama
     - öğrenme kümesindeki gürültülü verilerden oluşan ve sınama kümesinde hataya neden olan dalları silme (sınıflandırma başarımını artırır)

Karar Ağacı Oluşturma

- Yinelemeli işlem
  - ağaç bütün verinin oluşturduğu tek bir düğümle başlıyor
  - eğer örnekleri hepsi aynı sınıfa aitse düğüm yaprak olarak sonlanıyor ve sınıf etiketini alıyor
  - eğer değilse örnekleri sınıflara en iyi bölecek olan nitelik seçiliyor
  - işlem sona eriyor
    - örneklerin hepsi (çoğunluğu) aynı sınıfa ait
    - örnekleri bölecek nitelik kalmamış
    - kalan niteliklerin değerini taşıyan örnek yok

Örnekleri En İyi Bölen Nitelik Hangisi?

- Bölmeden önce:
  - 10 örnek C0 sınıfında
  - 10 örnek C1 sınıfında



Hangisi daha iyi?

En iyi Bölme Nasıl Belirlenir?

- "Greedy" yaklaşım
  - çoğunlukla aynı sınıfa ait örneklerin bulunduğu (homojen) düğümler tercih edilir
- Düğümün kalitesini ölçmek için bir yöntem

C0: 5
C1: 5

homojen değil
kalitesi düşük

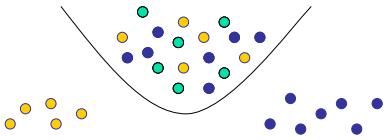
C0: 9
C1: 1

homojen
kalitesi yüksek

En İyi Bölen Nitelik Nasıl Belirlenir?

- İyilik Fonksiyonu (Goodness Function)
- Farklı algoritmalar farklı iyilik fonksiyonları kullanabilir:
  - bilgi kazancı (information gain): ID3, C4.5
    - bütün niteliklerin ayrık değerler aldığı varsayılıyor
    - sürekli değişkenlere uygulamak için değişiklik yapılabilir
  - gini index (IBM IntelligentMiner)
    - bütün niteliklerin sürekli değerler aldığı varsayılıyor
    - her nitelik için farklı bölme değerleri olduğu varsayılıyor
    - bölme değerlerini belirlemek için başka yöntemlere (demetleme gibi) ihtiyaç var
    - ayrık değişkenlere uygulamak için değişiklik yapılabilir

Bilgi Kazancı



sepetteki toplar farklı renklerde belirsizlik fazla
topların hepsi aynı renkte ise daha belirsizlik yok

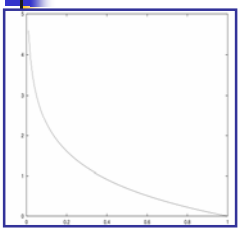
Bilgi / Entropi

- p_1, p_2,.., p_s toplamları 1 olan olasılıklar. Entropi (Entropy)

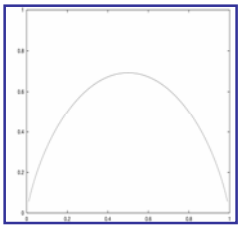
$$H(p_1, p_2, ..., p_s) = -\sum_{i=1}^{s} p_i \log(p_i)$$

- Entropi rastgeleliği, belirsizliği ve beklenmeyen durumun ortaya çıkma olasılığını gösterir
- Sınıflandırmada
  - olayın olması beklenen bir durum
  - entropi=0

Entropi



log (p)



H(p,1-p)

- örnekler aynı sınıfa aitse entropi=0
- örnekler sınıflar arasında eşit dağılmışsa entropi=1
- örnekler sınıflar arasında rastgele dağılmışsa 0<entropi<1

Örnek

- S veri kümesinde 14 örnek: C0 sınıfına ait 9, C1 sınıfına ait 5 örnek.
- Entropi $H(p_1, p_2, ..., p_s) = -\sum_{i=1}^{s} p_i \log(p_i)$
- $H(p_1,p_2) = - (9/14) \text{ Log}_2 (9/14) - (5/14) \text{ Log}_2 (5/14)$
  $= 0.940$

Bilgi Kazancı (ID3 / C4.5)

- Bilgi kuramı kavramlarını kullanarak karar ağacı oluşturulur. Sınıflandırma sonucu için en az sayıda karşılaştırma yapmayı hedefler.
- Ağaç bir niteliğe göre dallandığında entropi ne kadar düşer?
- *A* niteliğinin *S* veri kümesindeki bilgi kazancı

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|Sv|}{|S|} Entropy(S_v)$$

Values(A), *A* niteliğinin alabileceği değerler, S_v, *A=v* olduğu durumda S'nin altkümesi.

Örnek

- Bilgi kazancına göre nitelik seçme

toplam örnek sayısı s=14, iki sınıfa ayrılmış
s1=9(yes), s2=5(no)

Entropy(S) = - (9/14) Log_2 (9/14) - (5/14) Log_2 (5/14) = 0.940

wind için: weak=8, strong=6

weak: no=2, yes=6

strong: no=3, yes=3

Entropy(S_{weak}) = - (6/8)*log2(6/8) - (2/8)*log2(2/8) = 0.811

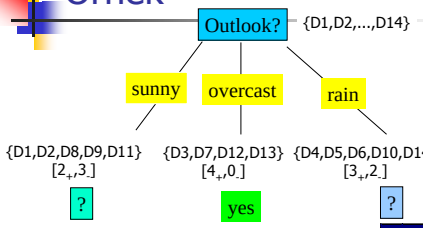
Entropy(S_{strong}) = - (3/6)*log2(3/6) - (3/6)*log2(3/6) = 1.00

Gain(wind)=0.940 - (8/14)*0.811 - (6/14)*1.00

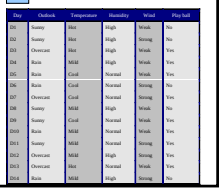
Day	Outlook	Temperature	Humidity	Wind	Play ball
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Gain(Outlook) = 0.246
Gain(Humidity) = 0.151
Gain(wind)=0.048
Gain(Temperature) = 0.029

Örnek



S_{sunny}= {D1,D2,D8,D9,D11}, Entropy(S_{sunny})=0.970
humidity için: high=3, normal=2
high: no=3, yes=0
normal: no=0, yes=2
Entropy(S_{high}) = 0
Entropy(S_{normal}) = 0
Gain(S_{sunny},Humidity)=0.970-(3/5)0.0- (3/5)0.0=0.970



Gini Index (IBM InteligentMiner)

- Veri kümesi S içinde n sınıf varsa ve p_j C_j sınıfının olasılığı ise

$$gini(S) = 1 - \sum_{j=1}^{n} p_j^2$$

- Eğer veri kümesi S_1 ve S_2 altkümelere bölünüyorsa ve her altkümede sırasıyla N_1 ve N_2 örnek varsa:

$$gini_{split}(S) = \frac{N_1}{N} gini(S_1) + \frac{N_2}{N} gini(S_2)$$

- Gini Index değeri en küçük olan nitelik seçilir.

Örnek

$$GINI(S) = 1 - \sum_j [p_j]^2$$

C1	0
C2	6

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1
Gini = 1 – P(C1)² – P(C2)² = 1 – 0 – 1 = 0

C1	1
C2	5

P(C1) = 1/6 P(C2) = 5/6
Gini = 1 – (1/6)² – (5/6)² = 0.278

C1	2
C2	4

P(C1) = 2/6 P(C2) = 4/6
Gini = 1 – (2/6)² – (4/6)² = 0.444

Sürekli Nitelikleri Bölme

- Bölmeleme:
  - Statik: En başta bölmelenir
    - Bölmeler eşit genişlik, eşit derinlik veya demetleme yöntemi ile bulunur.
  - Dinamik:
    - Sürekli nitelik *A* sıralanır. Birbirini izleyen ancak sınıf etiketi farklı olan nitelik değerleri bulunur. En fazla kazanç sağlayan bölme seçilir.

Temperature	40	48	60	72	80	90
Play tennis	No	No	Yes	Yes	Yes	No

Ağaç Oluşturmada Temel Yaklaşımlar

- Bölme kriteri:
  - ağacın bir düğümünde karşılaştırma yapılacak niteliğin seçilmesi
  - farklı algoritmalar farklı iyilik fonksiyonları kullanabilir: bilgi kazancı, gini index,...
- Dallanma kriteri:
  - bir örneğin hangi dala ait olduğunu belirleme
  - ikiye dallanma (gini index), çoklu dallanma (bilgi kazancı)
- Durma kararı:
  - dallanma işleminin devam edip etmeyeceğine karar verme
- Etiketleme kuralı:
  - yaprak düğüm en çok örneği olan sınıfla etiketleniyor

Ağaç Oluşturma:

- Parçala ve çöz (divide and conquer)
  - kökten yapraklara
  - düğümü dallara ayır
    - 'Greedy' algoritma
      - her adımda en iyi çözümü bul: her düğümde dallanmak için en iyi niteliği bul
  - her dal için algoritmayı uygula

Örnek Algoritma: ID3

- Bütün nitelikler ayrık
- Bir düğüm oluştur N:
  - Eğer örneklerin hepsi C sınıfına ait ise, N düğümü C etiketli yaprak
  - Eğer karşılaştırma yapılacak nitelik yoksa N düğümü en çok örneği olan sınıf
- En büyük bilgi kazancı olan niteliği bölmek için seç
  - N'yi seçilen nitelik ile etiketle
  - niteliğin her A_i değeri için bir dal oluştur
  - S_i, örneklerin hepsinin A_i değeri aldığı dal
    - S_i boş → bir yaprak oluşturup en çok örneği olan sınıfla etiketle
    - S_i boş değil → algoritmayı S_i düğümü üzerinde yinele
- Yaprak düğümlere kadar

Ayrıntılı bilgiler: http://dms.irb.hr/tutorial/tut_dtrees.php

Örnek Algoritma: C4.5

- Kökten yapraklara doğru ağaç oluşturma
- Bilgi kazancı yöntemini kullanıyor
- Bütün veri kümesini bellekte tutuyor
  - Büyük veri kümeleri için uygun değil

http://www.rulequest.com/Personal/c4.5r8.tar.gz

Karar Ağacı Kullanarak Sınıflandırma

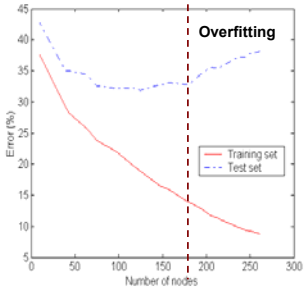
- Doğrudan
  - sınıflandırmak istenilen örneğin nitelikleri ağaç boyunca sınanır
  - ulaşılan yaprağın etiketi sınıf bilgisini verir
- Dolaylı
  - karar ağacı sınıflandırma kurallarına dönüştürülür
  - kökten yaprakların herbirine giden yollar için ayrı bir kural oluşturulur.
  - IF-THEN şeklinde kuralları insanlar daha kolay anlıyor
    - Örnek: IF Outlook="sunny" AND humidity="normal" THEN play tennis

Karar Ağacı Kullanarak Sınıflandırma

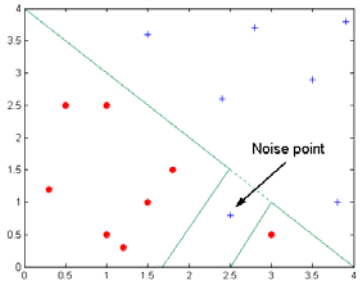
- Avantajları:
  - Karar ağacı oluşturmak zahmetsiz
  - Küçük ağaçları yorumlamak kolay
  - Anlaşılabilir kurallar oluşturulabilir
  - Sürekli ve ayrık nitelik değerleri için kullanılabilir
- Dezavantajları:
  - Sürekli nitelik değerlerini tahmin etmekte çok başarılı değil
  - Sınıf sayısı fazla ve öğrenme kümesi örnekleri sayısı az olduğunda model oluşturma çok başarılı değil
  - Zaman ve yer karmaşıklığı öğrenme kümesi örnekleri sayısına (q), nitelik sayısına (h) ve oluşan ağacın yapısına bağlı.
  - Ağaç oluşturma karmaşıklığı fazla, ağaç budama karmaşıklığı fazla
    - ağaç oluşturmak için zaman karmaşıklığı: O(h q logq)

Karar Ağaçlarında Aşırı Öğrenme

- Öğrenme kümesindeki örneklerin azlığı veya gürültülü olması
- Aşırı öğrenmeyi engelleyen iki yaklaşım
  - işlemi erken sona erdirme
    - işlemi sona erdirmek için eşik değeri belirlemek gerekiyor
  - karar ağacı oluştuktan sonra ağacı küçültme

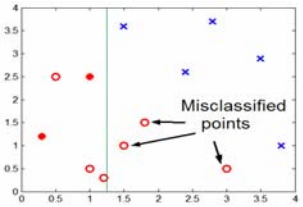


Aşırı Öğrenme:Gürültülü Örnekler



Gürültülü örnekler nedeniyle sınıfları ayıran düzlemin bozulması

Aşırı Öğrenme: Yetersiz Öğrenme Kümesi



- Öğrenme kümesindeki örnek sayısının yetersiz olması nedeniyle sınama kümesindeki örneklerin yanlış sınıflandırılması

Aşırı Öğrenme

- Gereğinden fazla karmaşık karar ağaçları aşırı öğrenmeye neden oluyor.
- Karar ağacının yeni örnekler üzerindeki başarımını tahmin etmek için öğrenme kümesi örnekleri yeterli olmuyor.
- Hatayı tahmin etmek için farklı yöntemler gerekli.

Genel Hatayı Tahmin Etme

- Yerine koyma (Resubstitution) Hatası: öğrenme kümesi kullanılarak hesaplanan hata (Σ e(t))
- Genel (Generalization) hata: sınama kümesi kullanılarak hesaplanan hata (Σ e'(t))
- Genel hatayı tahmin etme yöntemleri:
  - İyimser yaklaşım: e'(t) = e(t)
  - Kötümser yaklaşım:
    - Her yaprak düğüm için: e'(t) = (e(t)+0.5)
    - Toplam hata: e'(T) = e(T) + N × 0.5 (N: yaprak düğüm sayısı)
    - 30 yaprak düğümü olan bir karar ağacı, 1000 öğrenme kümesi örneğinden 10 örneği yanlış sınıflandırırsa
      - Yerine koyma hatası: 10/1000 = %1
      - Genel hata: (10 + 30×0.5)/1000 = % 2.5
  - Ağaç budama: Genel hatayı tahmin etmek için geçerleme kümesi kullanılır

Occam's Razor

- Genel hatası aynı olan iki modelden karmaşıklığı daha az olan seçilmeli
- Karmaşık modellerin veri içindeki gürültüyü öğrenme ihtimalleri daha fazla

Karar Ağacı Boyutunu Belirleme

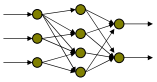
- Veri kümesi öğrenme ve sınama kümesi olarak ayrılır
- Çapraz geçerleme kullanılır.
- Veri kümesinin tümü ağacı oluşturmak için kullanılır
  - istatistiksel bir test ile (chi-square) düğüm eklemenin ya da ağacı küçültmenin katkısı sınanır
  - MDL (Minimum Description Length) yöntemi kullanılır: kodlama en aza indirildiğinde ağacın büyümesi durdurulur

Konular

- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

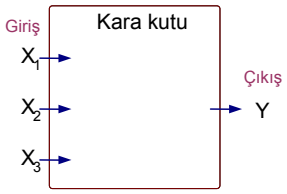
Yapay Sinir Ağları ile Sınıflandırma

- İnsan beynindeki sinir hücrelerinin işlevini modelleyen bir yapı
- Birbiri ile bağlantılı katmanlardan oluşur.
  - katmanlar hücrelerden oluşur
- Katmanlar arasında iletim
- İleti katmanlar arasındaki bağın ağırlığına ve her hücrenin değerine bağlı olarak değişebilir



Örnek:

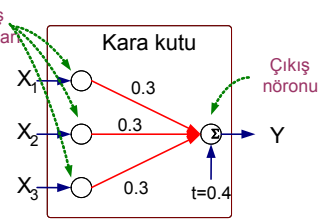
X_1	X_2	X_3	Y
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	0	1	0
0	1	0	0
0	1	1	1
0	0	0	0



- En az iki giriş 1 ise çıkış 1, diğer durumlarda çıkış 0

Örnek

X_1	X_2	X_3	Y
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1
0	0	1	0
0	1	0	0
0	1	1	1
0	0	0	0

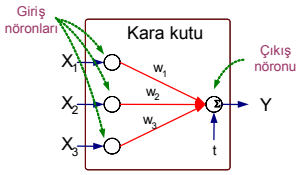


$$Y = I(0.3X_1 + 0.3X_2 + 0.3X_3 - 0.4 > 0)$$

$$I(z) = \begin{cases} 1 \text{ eğer } z>0 \\ 0 \text{ diğer} \end{cases}$$

Yapay Sinir Ağları

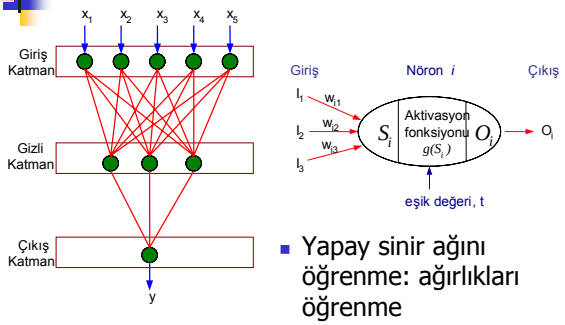
- Birbiri ile bağlantılı nöronlar ve ağırlıklar
- Çıkış nöronu kendisine gelen girişleri ağırlıklı olarak topluyor
- Çıkış nöronu bir eşik değeri ile karşılaştırılıyor



$$Y = I(\sum_i w_i X_i - t)$$

$$Y = sign(\sum_i w_i X_i - t)$$

Çok Katmanlı



- Yapay sinir ağını öğrenme: ağırlıkları öğrenme

Yapay Sinir Ağı ile Öğrenme

- Yapay sinir ağı oluşturma
  - giriş verisini modelleme
  - gizli katman sayısını, gizli katmanlardaki nöron sayısını belirleme
- Yapay sinir ağını eğitme
- Sinir ağını küçültme
- Sonucu yorumlama

Yapay Sinir Ağını Oluşturma

- Giriş nöron sayısı
  - Öğrenme kümesindeki verilerin nitelik sayısı
- Gizli nöron sayısı
  - öğrenme sırasında ayarlanır
- Çıkış nöron sayısı
  - sınıf sayısı

Yapay Sinir Ağını Eğitme

- Amaç: Veri kümesindeki örneklerin hepsini doğru sınıflandıracak ağırlıkları belirlemek
  - ağırlıklara rasgele değerler ata
  - öğrenme kümesindeki giriş değerlerini teker teker sinir ağına uygula
  - çıkışı hesapla
  - hata değerini hesapla $E = \sum_i [Y_i - f(w_i, X_i)]^2$
  - ağırlıkları hata fonksiyonunu enküçültecek şekilde düzelt

Yapay Sinir Ağını Küçültme

- Tam bağlı ağın anlaşılması çok güç
- n giriş nöron, h gizli nöron, m çıkış nöronu
  h(m+n) ağırlık
- Küçültme: ağırlıklardan bazıları sınıflandırma sonucunu etkilemeyecek şekilde silinir

Yapay Sinir Ağları

- Avantajları
  - doğru sınıflandırma oranı genelde yüksek
  - kararlı – öğrenme kümesinde hata olduğu durumda da çalışıyor
  - çıkış ayrık, sürekli ya da ayrık veya sürekli değişkenlerden oluşan bir vektör olabilir
- Dezavantajları
  - öğrenme süresi uzun
  - öğrenilen fonksiyonun anlaşılması zor

Konular

- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

Bayes (İstatistiksel) Modelleme

- Bayes teoremini kullanan istatistiksel sınıflandırıcı
- Örneklerin hangi sınıfa hangi olasılıkla ait oldukları
- Naïve Bayes sınıflandırıcı
  - niteliklerin hepsi aynı derecede önemli
  - nitelikler birbirinden bağımsız
    - bir niteliğin değeri başka bir nitelik değeri hakkında bilgi içermiyor
  - sınıflandırma ve öğrenme problemleri

Bayes Teoremi

- X sınıflandırılacak örnek. Hipotez h, X örneğinin C sınıfına ait olduğu
- h hipotezinin sonrasal olasılığı (*posteriori probability*)

$$P(h|X)=\frac{P(X|h)P(h)}{P(X)}$$

- MAP (maximum posteriori) hipotez

$$h_{MAP}\equiv\arg\max_{h\in H}P(h|D)=\arg\max_{h\in H}P(D|h)P(h).$$

- Çok sayıda olasılığı önceden kestirmek gerekiyor

Örnek

- + ○ —H→ 🍎
  X C
- P(H)=P(🍎) P(X)=P(+ ○)
- P(X|H)= P(🍎 ise + ○)

$$P(h|X)=\frac{P(X|h)P(h)}{P(X)}$$

Naïve Bayes Sınıflandırıcı

- $X=(x_1,x_2,...x_n)$ örneğinin C sınıfında olma olasılığı ($P(C|X)$) nedir?
- $\frac{P(X|C_i)P(C_i)}{P(X)}$ değerini enbüyütme

  → $P(X/C_i)P(C_i)$ değerini enbüyütme

- $P(C_i)=|S_i|/|S|$, S_i: C_i sınıfına ait örneklerin sayısı
- $P(X|C_i)=\prod_{k=1}^{n}P(x_k|C_i)$, $P(x_k|C_i)=s_{ik}/s_i$
- Hesaplama maliyetini azaltıyor, sadece sınıf dağılımları hesaplanıyor
- Naïve: nitelikler bağımsız

Hava Durumu Verisi için Olasılıklar

Outlook			Temperature			Humidity			Windy			Play	
	Yes	No		Yes	No		Yes	No		Yes	No	Yes	No
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

Outlook	Temp	Humidity	Windy	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Hava Durumu Verisi için Olasılıklar

Outlook			Temperature			Humidity			Windy			Play	
	Yes	No		Yes	No		Yes	No		Yes	No	Yes	No
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

- Yeni veri

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

$$P(C_i|X)=P(X|C_i)\times P(C_i)=\prod_{k=1}^{n}P(x_k|C_i)\times P(C_i)$$

İki Sınıf için olasılık:
P("yes"|X) = 2/9 × 3/9 × 3/9 × 3/9 × 9/14 = 0.0053
P("no"|X) = 3/5 × 1/5 × 4/5 × 3/5 × 5/14 = 0.0206
Normalize edilmiş olasılıklar:
P("yes") = 0.0053 / (0.0053 + 0.0206) = 0.205
P("no") = 0.0206 / (0.0053 + 0.0206) = 0.795

Sürekli Veriler için Olasılık

- Verinin normal dağılımdan geldiği varsayılıyor.
  - Her sınıf-nitelik çifti için bir olasılık hesaplanıyor.

$$P(A_i \mid c_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(A_i-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

  - Gelir için sınıf=-1
    - ortalama=110
    - varyans=2975

$$P(Gelir = 120 \mid -1) = \frac{1}{\sqrt{2\pi}(54.54)} e^{-\frac{(120-110)^2}{2(2975)}} = 0.0072$$

Tid	Geri Ödeme	Medeni Durum	Gelir	Dolandırıcı
1	Evet	Bekar	125K	-1
2	Hayır	Evli	100K	-1
3	Hayır	Bekar	70K	-1
4	Evet	Evli	120K	-1
5	Hayır	Boşanmış	95K	1
6	Hayır	Evli	60K	-1
7	Evet	Boşanmış	220K	-1
8	Hayır	Bekar	85K	1
9	Hayır	Evli	75K	-1
10	Hayır	Bekar	90K	1

Örnek

X=(Geri Ödeme="Hayır",Medeni Durum="Evli",Gelir=120k)

P(Geri Ödeme="Evet"|-1)=3/7
P(Geri Ödeme="Hayır"|-1)=4/7
P(Geri Ödeme="Evet"|1)=0
P(Geri Ödeme="Hayır"|1)=1
P(Medeni Durum="Evli"|-1)=4/7
P(Medeni Durum="Bekar"|-1)=2/7
P(Medeni Durum="Boşanmış"|-1)=1/7
P(Medeni Durum="Evli"|1)=0
P(Medeni Durum="Bekar"|1)=2/3
P(Medeni Durum="Boşanmış"|1)=1/3

Gelir:
- Sınıf=-1
  - ortalama=110
  - varyans=2975
- Sınıf=1
  - ortalama=90
  - varyans=25

Tid	Geri Ödeme	Medeni Durum	Gelir	Dolandırıcı
1	Evet	Bekar	125K	-1
2	Hayır	Evli	100K	-1
3	Hayır	Bekar	70K	-1
4	Evet	Evli	120K	-1
5	Hayır	Boşanmış	95K	1
6	Hayır	Evli	60K	-1
7	Evet	Boşanmış	220K	-1
8	Hayır	Bekar	85K	1
9	Hayır	Evli	75K	-1
10	Hayır	Bekar	90K	1

Örnek

X=(Geri Ödeme="Hayır",Medeni Durum="Evli",Gelir=120k)

P(Geri Ödeme="Evet"|-1)=3/7
P(Geri Ödeme="Hayır"|-1)=4/7
P(Geri Ödeme="Evet"|1)=0
P(Geri Ödeme="Hayır"|1)=1
P(Medeni Durum="Evli"|-1)=4/7
P(Medeni Durum="Bekar"|-1)=2/7
P(Medeni Durum="Boşanmış"|-1)=1/7
P(Medeni Durum="Evli"|1)=0
P(Medeni Durum="Bekar"|1)=2/3
P(Medeni Durum="Boşanmış"|1)=1/3

Gelir:
- Sınıf=-1
  - ortalama=110
  - varyans=2975
- Sınıf=1
  - ortalama=90
  - varyans=25

- P(X|Sınıf=-1) =
  P(Geri Ödeme="Hayır"|Sınıf=-1)
  × P(Medeni Durum="Evli"|Sınıf=-1)
  × P(Gelir=120K| Sınıf=-1)
  = 4/7 × 4/7 × 0.0072 = 0.0024

- P(X|Sınıf=1) =
  P(Geri Ödeme="Hayır"|Sınıf=1)
  × P(Medeni Durum="Evli"| Sınıf=1)
  × P(Gelir=120K| Sınıf=1)
  = 1 × 0 × 1.2 × 10^{-9} = 0

P(X|-1)P(-1) > P(X|1)P(1)
P(-1|X) > P(1|X)
=> Sınıf = -1

Olasılığın Sıfır Olması

- Her sınıfta bir niteliğin her değeri olmazsa
  - koşullu olasılıklardan biri 0
  - o sınıfa ait olma olasılığı 0
- Olasılıklar

$$\text{Original}: P(A_i \mid C) = \frac{N_{ic}}{N_c}$$

$$\text{Laplace}: P(A_i \mid C) = \frac{N_{ic}+1}{N_c + c}$$

c: sınıf sayısı
Toplamları 1 olmak zorunda

Bayes Sınıflandırıcılar

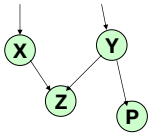
- Avantajları:
  - gerçeklemesi kolay
  - çoğu durumda iyi sonuçlar
- Dezavantajları
  - varsayım: sınıf bilgisi verildiğinde nitelikler bağımsız
  - gerçek hayatta değişkenler birbirine bağımlı
  - değişkenler arası ilişki modellenemiyor
- Çözüm:
  - Bayes ağları

Konular

- Sınıflandırma işlemi
  - Sınıflandırma tanımı
  - Sınıflandırma uygulamaları
- Sınıflandırma yöntemleri
  - Karar ağaçları
  - Yapay sinir ağları
  - Bayes sınıflandırıcılar
  - Bayes ağları

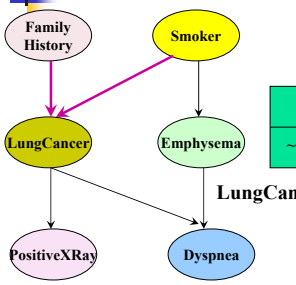
Bayes Ağları

- Niteliklerin altkümesinin birbiri ile bağımsız olduğunu varsayıyor
- Yönlü çevrimsiz çizge (directed acyclic graph) ve koşullu olasılık tablolarından oluşur
- Her değişken A için bir tablo var
  - niteliğin ebeveynlerine olan koşullu olasılıkları



- düğümler: rasgele değişkenler
- ayrıtlar: olasılıklı bağlılık
- *X* ve *Y* , *Z* değişkeninin ebeveyni
- *Y*, *P* değişkeninin ebeveyni
- *Z* ve *P* arasında bağ yok

Örnek



Bayes Ağları

	(FH, S)	(FH, ~S)	(~FH, S)	(~FH, ~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

LungCancer için koşullu olasılık tablosu

P(LC="yes"|FH="yes", S="yes")=0.8
P(LC="no"| FH="no", S="no")=0.9

Bayes Ağlarının Eğitilmesi

- Ağ yapısı ve tüm değişkenler biliniyorsa koşullu olasılıklar hesaplanır
- Ağ yapısı belli ancak bazı değişkenler eksik ise yinelemeli öğrenme uygulanır
  - gradient descent algoritması

- D. Heckerman. A Tutorial on Learning with Bayesian Networks. In *Learning in Graphical Models,* M. Jordan, ed.. MIT Press, Cambridge, MA, 1999. Also appears as Technical Report MSR-TR-95-06, Microsoft Research, March, 1995.