Yalova Universitesi
Fen Bilimleri Enstitusu
Bilgisayar Muhendisligi Ana Bilim Dali

Oruntu Tanima Dersi

Sevdanur GENC - 135105008

Oruntu Tanima - Decision Trees

Icindekiler **Sayfa**

Karar Agaclari

Pek cok problemin cozumu icin veriler uzerinde istatistik analizleri onemli bir metodtur. Bazi durumlarda istatistik kullanimi sinirli oldugundan dolayi akilli veri analizi yontemlerini iceren algoritmalar ortaya cikmistir. Ancak bu yontemlerin bazilari zayiftir. Bu zayifligin soz konusu olmayan bir yaklasim turu ise karar agaclaridir. Karar agaclari, hedef sonuclarinin yaklasik degerlerini hesaplamak icin kullanilan ve ogrenme verilerinin karar agaci ile gosterildigi bir yontemdir.

Karar agaclari icin kullanilacak karar agaci algoritmasi iki asamadan olusmaktadir;

1. Agac Olusturma ; Butun ogrenme verilerine sahip olunan bir kumedir.
2. Agac Budama ; Butun ogrenme verilerinin sahip oldugu kumeye ait olan ve test kumesinde hataya sebep olan dallarin agactan budanmasi, silinmesidir.

Tum bunlara baktigimizda, aslinda kesin bir optimum sonuc vermeyecektir. Bunun icin, gerekli optimum sonuca yaklasilmasinda kullanilan kurallar bulunmaktadir.

Karar Agaci Algoritmasinin Sahip Oldugu Adimlar;

1. Karar agacinin sonuc olarak hangi karari alacagi belirlenir.
2. Kurmus olacaginiz sistemin Entropy'si hesaplanir.
3. Agacin Root'u yani koku belirlenir. Bunu belirleyebilmek icin Information Gain hesaplanir. Yani bilgi kazanci, en yuksek olan agacin en ustteki yerini alir.

Entropy : Rastgeleligin, belirsizligin ve beklenmeyen durumun ortaya cikma olasiligini gosterir.

S' i bir dataset olarak kabul edelim. S dataseti icerisinde bulunan ornekler ayni sinifa ait ise entropy degeri 0, ornekler esit dagilmissa entropy 1 degerine yakin olacaktir. Ornekler siniflar arasinda rastgele dagilmissa 0<entropy<1 degeri beklenmektedir.

$$\text{Entropy}(S) \equiv -p_i \log_2 p_i - p_i \log_2 p_i$$

Entropy - Ornek I

Y	X1	X2	X3
Pi	3/6	2/6	1/6

Entropy belirsizligi hesaplanacak olursa;
E(S) = - (X1 log2 X1 + X2 log2 X2 + X3 log2 X3)
E(S) = - (3/6 log2 3/6 + 2/6 log2 2/6 + 1/6 log2 1/6)
E(S) = 1,4591

Entropy - Ornek II

Y	X1	X2	X3	X4	X5	X6	X7	X8
Pi	Evet	Evet	Hayir	Hayir	Hayir	Hayir	Hayir	Hayir

Entropy belirsizligi hesaplanacak olursa;
S uzayi = {Evet, Evet, Hayir, Hayir, Hayir, Hayir, Hayir, Hayir}
P (Evet) = Evet / S ve P (Hayir) = Hayir / S seklinde olasiliklari hesaplanir.
P (Evet) = 2/8 = 0.25 ve P (Hayir) = 6/8 = 0.75
E(S) = - (P (Evet) log2 P (Evet) + P (Hayir) log2 P (Hayir))
E(S) = - (0.25 log2 0.25 + 0.75 log2 0.75)
E(S) = 0.97

Information Gain (Bilgi Kazanci) : Karar agaci yontemlerinde en ayirt edici ozelligi belirlemek amactir. Bunun icin de, her ozellik icin bilgi kazanci hesaplanir. Bu hesaplama, Entropy hesabimi kullanilmaktadir.

Her ozelligin bilgi kazancinda dogal olarak bolunmelere neden olacaktir. Entropy bu bolunmelerin olculerini azaltacagindan dolayi en iyi azaltmayi saglayan bolunme basarili olarak secilir.

$$GAIN_{split} = Entropy(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i) \right)$$

Gini Indexleme : Dugum homojenliginin olcumunde kullanilan bir yontemdir. Kayitlarin butun siniflar arasinda esit olarak dagilmasiyla ilgilenir.

Bir t dugumundeki j sinifina ait bagil olasilik hesaplama ve buna bagli olarak gini tabanli bolunme formulleri soyledir (ni = child kayitlari);

$GINI(t) = 1 - \sum_{j} [p(j \mid t)]^2$ ve $GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$ formulleri kullanilir.

Gini Indexleme - Ornek I

Tek Oznitelikli C1 ve C2 isminde iki sinif olsun. Bu siniflardaki verilerin homojen dagilimlari sonucunda toplamda 6 ornek bulunmaktadir. Siniftaki orneklerin sayisi esit olana kadar dagilim gerceklesmektedir.

C1	C2
0	6

P(c1) = 0/6
P(c2) = 6/6
Gini = 1 – P(c1)^2 – P(c2)^2
Gini = 1 – o^2 – 1^2
Gini = 0

C1	C2
1	5

P(c1) = 1/6
P(c2) = 5/6
Gini = 1 – P(c1)^2 – P(c2)^2
Gini = 1 –(1/6)^2 – (5/6)^2
Gini = 0.278

C1	C2
2	4

P(c1) = 2/6
P(c2) = 4/6
Gini = 1 – P(c1)^2 – P(c2)^2
Gini = 1 –(2/6)^2 – (4/6)^2
Gini = 0.444

C1	C2
3	3

P(c1) = 3/6
P(c2) = 3/6
Gini = 1 – P(c1)^2 – P(c2)^2
Gini = 1 – (3/6)^2 – (3/6)^2
Gini = 0.500

Gini Indexleme - Ornek II

Kayitlarda ikiser oznitelik oldugu dusunulurse (Asagidaki formuller kullanilir);

$$GINI(t) = 1 - \sum_{j} [p(j \mid t)]^2$$

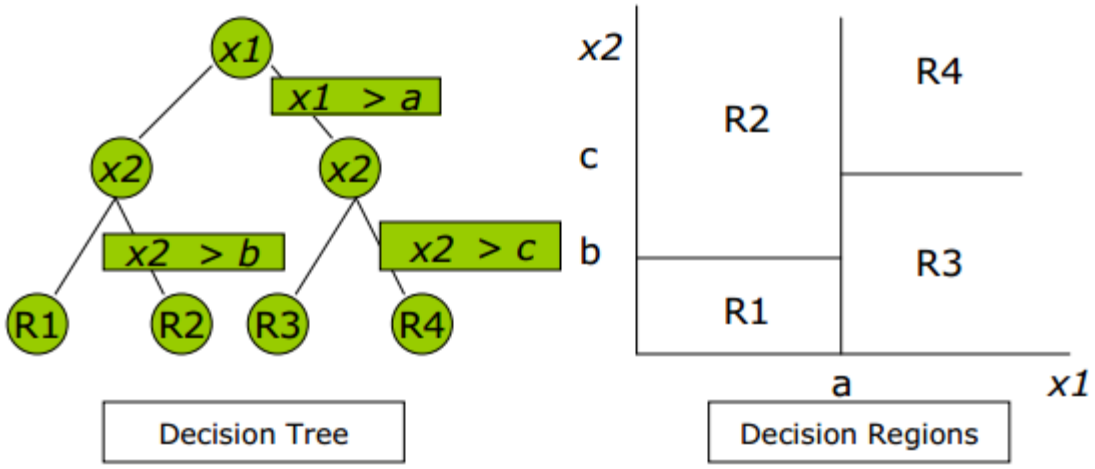
$$GINI_{split} = \sum_{i=1}^{k} \frac{n_i}{n} GINI(i)$$

	N1	N2	Parent
C1	5	1	6
C2	2	4	6
Children	7	5	

Gini (N1) = 1 - P(c1)^2 – P(c2)^2
Gini (N1) = 1 - (5/7)^2 – (2/7)^2
Gini (N1) = 0.408

Gini (N2) = 1 - P(c1)^2 – P(c2)^2
Gini (N2) = 1 - (1/5)^2 – (4/5)^2
Gini (N2) = 0.320

Gini (Children) = ∑(Children / Parent) * Gini(N)
Gini (Children) = 7/12 * 0.408 + 5/12 * 0.320
Gini (Children) = 0.371



CART (Classification & Regression Trees)

Siniflandirma ve regrasyon agaclari olarak bilinen bu modelin temelinde tek degiskenli ikili kararlarin bir hiyerarsisini icermektedir. Yaygin olarak kullanilan bu istatistiksel prosedur, verileri iki alt kumeye ayirmaktadir. Her bir alt kume icerisindeki veriler bir onceki alt kumeye ait verilerden biraz daha fazla homejen bir yapiya sahip olmaktadir. Birbirini devam eden bu surecler en optimize edilmis haldeki homojenlik kriterini veya durma kosullarini saglayincaya kadar kendini surekli tekrar edecektir. Tum bu sureclerde en iyi seceneklerin secilmesine ozen gosterilmektedir buna bagli olarakta bolunme kriterleri kullanilmaktadir. Kisaca ozetleyecek olursak, CART agaclari kesin bir heterojenlige (impurity) sahiptirler ve bu heterojenlik iki degerli (binary) agaclar yardimiyla optimize edilerek homojen hale getirilmektedir. Hedef, ayni veya yakin sonuc cikti degerlerinin oldugu alt gruplar yaratilmasidir.

CART algoritmalarina ornek verecek olursak; Twoing ve Gini algoritmalari.

CART agacini olustururken en iyi dallara ayirma kriterini secmek icin Entropy'den faydalanilmaktadir. Bu kriteri en iyi sekilde sonuclandirmak icin ise kullanilan formul ;

$$\Psi\left(\frac{s}{t}\right) = 2\,P_L P_R \sum_{J=1}^{M} \left| P\left(\frac{C_j}{t_L}\right) - P\left(\frac{C_j}{t_R}\right) \right|$$

W(s/t) : Herhangi bir t dugumundeki s dallari

t : Dallanmanin yapilacagi dugumler

C : Kriteri

L : Agacin sol yani

R : Agacin sag yani

PL ve PR : Egitim seti icerisindeki bir verinin agacin solunda yada saginda olma olasiligi

P(Cj / tL) ve P(Cj / tR) : Verilerin bulundugu Cj sinifindaki bir kaydin agactaki yerinin solunda yada saginda olma olasiligi.

Bu formule dayanaraktan soylenebilecek kural ;

- Dallanmalar en buyuk kritere gore gerceklestiriliyorsa Twoing algoritmasi, en kucuk kriterlere gore gerceklestiriliyorsa Gini algoritmasinin kullanilmasi tavsiye ediliyor. Gini algoritmasindaki amac; her zaman her adimda en buyuk veri kumelemesinin olusturulmasidir. Bu kumelemeler sonuclandiginda ilgilenilmeyen dallar budanabilir. Twoing algoristmasindaki amac ise; her zaman ana dugum ve yavru dugumlerin cogunlugunun yarisi uzerinde calisma hedefidir. Gini algoritmasina gore daha yavas calisacak ve veriler uzerinde daha dengeli bir tavir sergilemis olacaktir.

CART Agaclari uzerinde calisirken minimum sayidaki n dugumu belirlenir. n dugumunun sayisini belirlerken genellikle veri setinin yuzde 10'u kadar bir deger secilir. Aksi bir degerin secilmesi algoritmayi ya hizlandirir yada yavaslatir ve bu test analiz sonuclarini yanlis degerlendirmis olur.

CART yaklasiminda, siniflar arasi ayrim maksimize edilirken, sinif icerisindeki varyasyonun minimize edilmesi bir kural olarak benimsenmistir. Hem kategorik hem de surekli bagimli degiskenlerin modellenmesi soz konusudur. Bagimli degiskenler eger kategorik ise yontem Siniflandirma Agaci (CT - Classification Tree), surekli ise Regresyon Agaci (Regression Tree) ismini almaktadir.

Siniflandirma Agaci (CT) : Siniflandirma agaci genelde turlerin dagilimi modellenmesi icin kullanilmaktadir. Bu sebepten bagimli degisken Var/Yok veya Evet/Hayir gibi ikili kategorileri icermektedir. Ikili bagimli degiskenlerinin homojenligine karar verirken Gini katisiklik olcumu kullanilir. Herhangi bir t dugumu icin g(t) fonksiyonu soyledir;

$$g(t) = \sum_{j \neq i} p(j \mid t) p(i \mid t)$$

Buradaki i ve j egitim setindeki hedef (bagimli) degiskenin kategorileridir. Egerki ikili kategorilerden olusan bir yontem kullaniliyorsa formul esitligi asagidaki gibi degisecektir;

$$g(t) = 2p(1 \mid t) p(2 \mid t)$$

Herhangi bir t dugumune gelen bir ornegin s olarak bilinmesi ile, hem sol taraf ayrimini (tl) hem de sag taraf (tg) ayrimini gerceklestirecektir.

$$\phi(s,t) = g(t) - p_L g(t_L) - p_R g(t_R)$$

Burada, t dugumundeki durumlarin oranini belirtirken sag taraftaki Pr, sol taraftaki Pl degerleri belirlenir.

Regresyon Agaci (RT) : Regresyon agac mantiginda siniflara yer yoktur. Buna bagli olarak Gini indeksleme de kullanilmaz. Agac olusturulurken ikiye ayrilan sonuclarda dugumlerin tahmini toplam varyansin minimize edilerek hesaplanmasi gerekiyor. Agac olusturulurken her bir dugum icin yapilmasi gereken minimizasyon yani azaltma islemi icin gerekli formul asagidadir;

$$\arg\min_{xj \leq x_j^R,\, j=1,\ldots,M} \left[P_l Var(Y_l) + P_r Var(Y_r)\right]$$

Burada yine, Pl ile sol dugum Pr ile sag dugum olasiliklari hesaplanmak istenmistir. Egitim setindeki degiskenlerin sayisini M ile ifade etmistir. Var(Yl) ve Var(Yr) karsilikli sag ve sol alt dugumlerin verktorlerini temsil etmektedir. Artiklarin karelerinin azaltma algoritmasina gore asagidaki formulde;

$$i(t) = 1 - \sum_{k=1}^{K} p^2(k \mid t)$$

P(k|t) dugumunun t icerisinde bulundugu sinifin k'nin kosullarina bagli ozelliklerini, K sinif sayisi ve k sinif sayisi indeksi ile t dugum indeksini belirtmistir.

Regresyon Analizi :

Bir veya birden fazla kullanilan bagimsiz degiskenler ile bagimli degiskenlerin arasindaki iliskiyi kiyaslamak icin Regresyon analiz yontemi kullanilmaktadir. Iki yonteme sahiptir;

I) Tek degiskenli regresyon analiz modeli

II) Cok degiskenli regresyon analiz modeli

I) Tek degiskenli regresyon analiz modeli : Bir bagimli degisken ve bir bagimsiz degisken arasindaki iliskiyi analiz eder. Bu iki iliski arasinda temsili olarak bir dogrusallik ifade vardir ve bu bir dogrunun denklemi formulu ile ifade edilir. y = a + bx + e denklemi kullanilabilir.

II) Cok degiskenli regresyon analiz modeli : Bir bagimli degisken ve birden fazla bagimsiz degisken arasindaki iliskiyi analiz eder.

ID3 (Iterative Dichotomiser 3 - Tekrarlanan ikili yapi) :

Ozunde Entropy hesaplamasi kullanan bir algoritmadir. Entropy, bir veri kumesindeki verilerin belirsizliginin sayisallastirilmasi demektir. Algoritmanin amaci, egitim kumesindeki verilerin agacin olusturulmasi esnasinda birbirine benzetilmesi gerekiyor, agac derinliginin minimum olmasi, karmasikliginda minimum olmasini saglarken kazancin maksimum olmasi gozle gorulur bir fark alacaktir. Entropy deger araligi 0 < entropy < log2n arasinda olmalidir. Entropy degeri log2n'e yaklastikca belirsizligin artmasi, 0'a yaklasmasiyla belirsizligin azalmasi olarak bilinecektir. Ilk hesaplanmasi gereken entropy, tum data setin hesaplanmasi ile olusur. Sonrasinda datasetin farkli nitelikleri icinde entropy hesaplanmaktadir. Tum bu islemleri bilgi edinim icin kullanilmaktadir. Bu kavramdaki kazanc ise, ilk hesaplanan entropy ile her bir alt kumenin olusumundaki entropylerin arasindaki fark hesaplandiktan sonra, farki buyuk olan karar agacinin sagligi acisindan en dogru dallanmayi yapmis olacaktir.

Avantaj : Olasilik kurallari icin egitim verileri kullanilir ve tum data set'teki veriler agac olusturulmasi icin analiz edilir, sonucta kisa agaclar olusturdugu icin en hizli yapiya sahip olmus olur. Bu da, Test sayilarinin azalmasi ve test verilerinin budanmasini saglamaktadir.

Dezavantaj : Dataset'ten aldigimiz egitim verilerimizin boyutu kucukse agac test edildiginde cikan sonucun basarisiz olma olasiligi cok yuksektir.

C4.5 Algoritmasi :

ID3 algoritmasinin gelistiricisi, ID3 algoritmasi sonucunda siniflandirmalarda bazi eksiklikler ve sorunlar tespit etmistir. Bu sorunlarin giderilmesini C4.5 algoritmasi ile saglamistir. Kokeni tamamen ID3 algoritmasi olan C4.5 algoritmasina gelen ek ozellikler; bolunme-dagilma bilgisinin (split-info) edinilmesi, kayip degerleri olan ozelliklerin tespit edilmesi ve sayisal ozellikteki verilerin hesaba direk olarak katilmasi. Adim adim inceleyecek olursak;

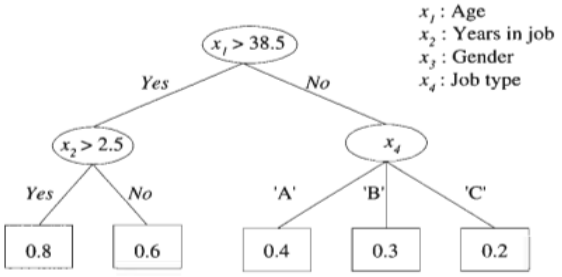
1. Split Information - Bolunme dallanma bilgisi : Bir ozniteligin olasiligi ne kadar yuksek olursa bilgi kazancida yuksek olacaktir, bu durum sonucun dogrulunu olumsuz etkileyecektir. Yani, bilgi kazancinin yuksek cikmasinin nedeni ozellik cesitliliginin fazla olmasidir.Bu tarz gereksiz bilgilerin onlenmesi icin bolunme bilgisi kavrami algoritmaya katilmistir. Bu algoritma ile bilgi kazancini azaltarak gereksiz bazi cikarimlar yapilmasi engellenmistir.

2. Ozelliklerin kayip degerlerinin tespit edilmesi : Dataset uzerindeki veriler cesitli sebeplerden dolayi eskik olabilir. ID3 algoritmasi eksik olmayan bir veri kumesini dikkate alir. Eksik olan veriler yuzunden bazi bilgilerin bulunmasinda yanlisliklar cikmaktadir. Bu sebepten, 3 sorunla karsi karsiya kalinmaktadir; bilgi kazanci ve bilgi oranlarinin kayip oldugu veri kumesinin hesabi nasil yapilabilir, karar agaci olusturulurken oznitelik degeri olmayanlar alt dugumlere nasil yayilabilir ve bununla beraberinde test islemleri agacin dallarinda nasil yapilabilir? Tum bunlara cevap bulabilmek icin verilerin oznitelikleri ve bu niteliklere sahip bilgi kazanclari sayesinde cozumlere ulasilabilmektedir.

3. Sayisal Ozellikteki verilerin hesaba direkt katilmasi : Veri kumelerinde iki tip veri kullanilir; Nominal (kategorik) ve sayisal. ID3 algoritmasinda sadece nominal degerler kullanilirken, C4.5 algoritmasinda

sayisal verilerede yer verilmistir. Tabi bu kullanimda bir yonteme ihtiyac duyulmustur. Sayisal degerler arasinda uygun bir esik degerinin bulunmasi gerekiyor. Esik degeri bulunduktan sonra ikili bir bolunme ile veri kumeleri dagitilabiliyor. Yani, bu esik degerinden buyuk ve esik degerinden kucuk veriler olmak uzere ikiye ayrim yapiliyor.

Esik degeri belirlenirken, tum sayisal verileri kucukten buyuge bir sekilde siralariz. {x1,x2,x3, ... , xm} kumesinde, m tane sayisal veri icerisinden xi '.nci veriyi esik degeri olarak seceriz. Bu secimden sonraki siralama artik; {x1,x2,x3, ... , xi} ve {xi+1,xi+2,xi+3, ... , xm} seklinde iki grup haline donusecektir. Bunun anlami aslinda verilerimiz icerisinden m-1 adet esik degeri secebilecegimizdir. Bu yuzden, olasi butun esik degerlerini (xi + xi+1 / 2) seklinde formulize edebiliriz. Esik degerine e dersek, xi < e sartini saglayan veriler kucuk xi > e sartini saglayan veriler ise buyuk seklinde gruplandirilacaktir.



R1: IF (age>38.5) AND (years-in-job>2.5) THEN y =0.8
R2: IF (age>38.5) AND (years-in-job≤2.5) THEN y =0.6
R3: IF (age≤38.5) AND (job-type='A') THEN y =0.4
R4: IF (age≤38.5) AND (job-type='B') THEN y =0.3
R5: IF (age≤38.5) AND (job-type='C') THEN y =0.2

ID3 UYGULAMASI

1.Adim : Dataset

Week	Weather	Temperatures (Isi)	Wetness	Wind (Ruzgar)	Game
W1	Sunny	Warm	High	Slightly	No
W2	Sunny	Warm	High	Strong	No
W3	Cloudy	Warm	High	Slightly	Yes
W4	Rainy	Warmish	High	Slightly	Yes
W5	Rainy	Cold	Normal	Slightly	Yes
W6	Rainy	Cold	Normal	Strong	No
W7	Cloudy	Cold	Normal	Strong	Yes
W8	Sunny	Warmish	High	Slightly	No
W9	Sunny	Cold	Normal	Slightly	Yes
W10	Rainy	Warmish	Normal	Slightly	Yes
W11	Sunny	Warmish	Normal	Strong	Yes
W12	Cloudy	Warmish	High	Strong	Yes
W13	Cloudy	Warm	Normal	Slightly	Yes
W14	Rainy	Warmish	High	Strong	No

Game = {No, No, Yes, Yes, Yes, No, Yes, No, Yes, Yes, Yes, Yes, Yes, No}
P(Game,No) = 5/14
P(Game,Yes) = 9/14
E(Game) = - (P (Game,No) $\log_2$ P (Game, No) + P (Game,Yes) $\log_2$ P (Game,Yes))
E(Game) = - (5/14 $\log_2$ 5/14 + 9/14 $\log_2$ 9/14)
E(Game) = 0.940

1. Adim – Isi niteligi Entropy - Gain Cozumu

[Temperatures,Cold] = 4
[Temperatures,Warmish] = 6
[Temperatures,Warm] = 4
E(Game) = 0.940
E(Temp,Game) = P(Temp,Cold) * E(Temp,Cold) + P(Temp,Warmish) * E(Temp,Warmish) + P(Temp,Warm) * E(Temp,Warm)
E(Temp,Game) = 4/14 E(Temp,Cold) + 6/14 E(Temp,Warmish) + 4/14 E(Temp,Warm)
E(Temp,Cold) = - (1/4 $\log_2$ 1/4 + 3/4 $\log_2$ 3/4)
E(Temp,Cold) = 0.811
E(Temp,Warmish) = - (2/6 $\log_2$ 2/6+ 4/6 $\log_2$ 4/6)
E(Temp,Warmish) = 0.918
E(Temp,Warm) = - (2/4 $\log_2$ 2/4 + 2/4 $\log_2$ 2/4)
E(Temp,Warm) = 1.00
E(Temp,Game) = 4/14 * 0.811 + 6/14 * 0.918 + 4/14 * 1.00
E(Temp,Game) = 0.911
Gain(Temp,Game) = E(Game) - E(Temp,Game)
Gain(Temp,Game) = 0.940 – 0.911
Gain(Temp,Game) = 0.029

Week	Temperatures	Game
W1	Warm	No
W2	Warm	No
W3	Warm	Yes
W4	Warmish	Yes
W5	Cold	Yes
W6	Cold	No
W7	Cold	Yes
W8	Warmish	No
W9	Cold	Yes
W10	Warmish	Yes
W11	Warmish	Yes
W12	Warmish	Yes
W13	Warm	Yes
W14	Warmish	No

1. Adim – Hava niteligi Entropy - Gain Cozumu

[Weather,Sunny] = 5
[Weather,Rainy] = 5
[Weather,Cloudy] = 4
E(Game) = 0.940
E(Weather,Game) = P(Weather,Sunny) * E(Weather,Sunny) + P(Weather,Cloudy) * E(Weather,Cloudy) + P(Weather,Rainy) * E(Weather,Rainy)
E(Weather,Game) = 5/14 E(Weather,Sunny) + 4/14 E(Weather,Cloudy) + 5/14 E(Weather,Rainy)
E(Weather,Sunny) = - (3/5 $\log_2$ 3/5 + 2/5 $\log_2$ 2/5)
E(Weather,Sunny) = 0.971
E(Weather,Rainy) = - (2/5 $\log_2$ 2/5+ 3/5 $\log_2$ 3/5)
E(Weather,Rainy) = 0.971
E(Weather,Cloudy) = - (4/4 $\log_2$ 4/4)
E(Weather,Cloudy) = 0
E(Weather,Game) = 5/14 * 0.971+ 5/14 * 0.971 + 4/14 * 0
E(Weather,Game) = 0.694
Gain(Weather,Game) = E(Game) - E(Weather,Game)
Gain(Weather,Game) = 0.940 – 0.694
Gain(Weather,Game) = 0.247

Week	Weather	Game
W1	Sunny	No
W2	Sunny	No
W3	Cloudy	Yes
W4	Rainy	Yes
W5	Rainy	Yes
W6	Rainy	No
W7	Cloudy	Yes
W8	Sunny	No
W9	Sunny	Yes
W10	Rainy	Yes
W11	Sunny	Yes
W12	Cloudy	Yes
W13	Cloudy	Yes
W14	Rainy	No

1. Adim – Nem niteligi Entropy - Gain Cozumu

[Wetness,High] = 7
[Wetness,Normal] = 7
E(Game) = 0.940
E(Wetness,Game) = P(Wetness,High) * E(Wetness,High) + P(Wetness,Normal) * E(Wetness,Normal)

Week	Wetness	Game
W1	High	No
W2	High	No
W3	High	Yes
W4	High	Yes
W5	Normal	Yes
W6	Normal	No
W7	Normal	Yes
W8	High	No
W9	Normal	Yes
W10	Normal	Yes
W11	Normal	Yes
W12	High	Yes
W13	Normal	Yes
W14	High	No

E(Wetness,Game) = 7/14 E(Wetness,High) + 7/14 E(Wetness,Normal)
E(Wetness,Sunny) = - (4/7 $\log_2$ 4/7 + 3/7 $\log_2$ 3/7)
E(Wetness,Sunny) = 0.985
E(Wetness,Rainy) = - (1/7 $\log_2$ 1/7+ 6/7 $\log_2$ 6/7)
E(Wetness,Rainy) = 0.592
E(Wetness,Game) = 7/14 * 0.985+ 7/14 * 0.592
E(Wetness,Game) = 0.789
Gain(Wetness,Game) = E(Game) - E(Wetness,Game)
Gain(Wetness,Game) = 0.940 – 0.789
Gain(Wetness,Game) = 0.151

1. Adim – Ruzgar niteligi Entropy - Gain Cozumu

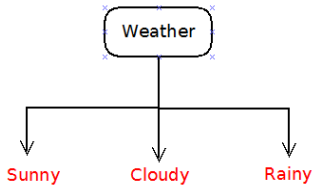
Week	Wind	Game
W1	Slightly	No
W2	Strong	No
W3	Slightly	Yes
W4	Slightly	Yes
W5	Slightly	Yes
W6	Strong	No
W7	Strong	Yes
W8	Slightly	No
W9	Slightly	Yes
W10	Slightly	Yes
W11	Strong	Yes
W12	Strong	Yes
W13	Slightly	Yes
W14	Strong	No

[Wind,Slightly] = 8
[Wind,Strong] = 6
E(Game) = 0.940
E(Wind,Game) = P(Wind,Slightly) * E(Wind,Slightly) + P(Wind,Strong) * E(Wind,Strong)
E(Wind,Game) = 8/14 E(Wind,Slightly) + 6/14 E(Wind,Strong)
E(Wind,Slightly) = - (2/8 $\log_2$ 2/8 + 6/8 $\log_2$ 6/8)
E(Wind,Slightly) = 0.811
E(Wind,Strong) = - (3/6 $\log_2$ 3/6+ 3/6 $\log_2$ 3/6)
E(Wind,Strong) = 1.00
E(Wind,Game) = 8/14 * 0.811 + 6/14 * 1.00
E(Wind,Game) = 0.892
Gain(Wetness,Game) = E(Game) - E(Wetness,Game)
Gain(Wetness,Game) = 0.940 – 0.892
Gain(Wetness,Game) = 0.0.48

1. Adim – Birinci Dallanma Sonucu

Oznitelikler	Bilgi Kazanci
Weather	0.246
Temperature	0.029
Weatness	0.151
Wind	0.048

Birinci Dallanma Sonucuna Ait Olusan Karar Agaci



2. Adim

Week	Weather	Temperatures	Wetness	Wind	Game
W1	Sunny	Warm	High	Slightly	No
W2	Sunny	Warm	High	Strong	No
W8	Sunny	Warmish	High	Slightly	No
W9	Sunny	Cold	Normal	Slightly	Yes
W11	Sunny	Warmish	Normal	Strong	Yes

Hava oznitelignin gunesli degeri icin dallanma degerleri
Game = {No, No, No, Yes, Yes}
P(Game,No) = 3/5
P(Game,Yes) = 2/5
E(Game) = - (P (Game,No) $\log_2$ P (Game, No) + P (Game,Yes) $\log_2$ P (Game,Yes))
E(Game) = - (3/5 $\log_2$ 3/5 + 2/5 $\log_2$ 2/5)
E(Game) = 0.970

2. Adim – Isi niteligi Entropy - Gain Cozumu

[Temperatures,Cold] = 1
[Temperatures,Warmish] = 2
[Temperatures,Warm] = 2
E(Game) = 0.970
E(Temp,Game) = P(Temp,Cold) * E(Temp,Cold) + P(Temp,Warmish) * E(Temp,Warmish) + P(Temp,Warm) * E(Temp,Warm)
E(Temp,Game) = 1/15 E(Temp,Cold) + 2/5 E(Temp,Warmish) + 2/5 E(Temp,Warm)
E(Temp,Cold) = - (1/1 $\log_2$ 1/1)
E(Temp,Cold) = 0
E(Temp,Warm) = - (2/2 $\log_2$ 2/2)
E(Temp,Warm) = 0
E(Temp,Warmish) = - (1/2 $\log_2$ 1/2 + 1/2 $\log_2$ 1/2)
E(Temp,Warmish) = 1.00
E(Temp,Game) = 1/5 * 0 + 2/5 * 0 + 2/5 * 1
E(Temp,Game) = 0.4
Gain(Temp,Game) = E(Game) - E(Temp,Game)
Gain(Temp,Game) = 0.970 – 0.4
Gain(Temp,Game) = 0.570

Week	Temperatures	Game
W1	Warm	No
W2	Warm	No
W8	Warmish	No
W9	Cold	Yes
W11	Warmish	Yes

2. Adim – Nem niteligi Entropy - Gain Cozumu

[Wetness,High] = 3
[Wetness,Normal] = 2
E(Game) = 0.970
E(Wetness,Game) = P(Wetness,High) * E(Wetness,High) + P(Wetness,Normal) * E(Wetness,Normal)
E(Wetness,Game) = 3/5 E(Wetness,High) + 2/5 E(Wetness,Normal)
E(Wetness,Sunny) = - (3/3 $\log_2$ 3/3)
E(Wetness,Sunny) = 0
E(Wetness,Rainy) = - (2/2 $\log_2$ 2/2)
E(Wetness,Rainy) = 0

Week	Wetness	Game
W1	High	No
W2	High	No
W8	High	No
W9	Normal	Yes
W11	Normal	Yes

E(Wetness,Game) = 3/5 * 0 + 2/5 * 0
E(Wetness,Game) = 0
Gain(Wetness,Game) = E(Game) - E(Wetness,Game)
Gain(Wetness,Game) = 0.970 – 0
Gain(Wetness,Game) = 0.970

2. Adim – Ruzgar niteligi Entropy - Gain Cozumu

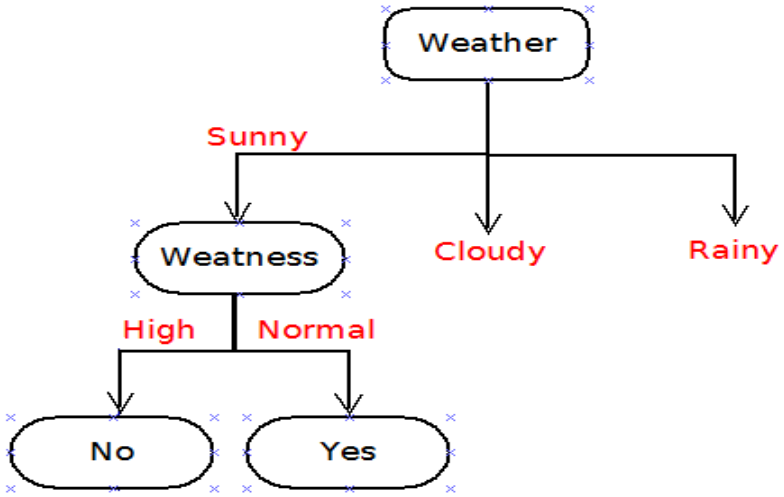
[Wind,Slightly] = 3
[Wind,Strong] = 2
E(Game) = 0.970
E(Wind,Game) = P(Wind,Slightly) * E(Wind,Slightly) + P(Wind,Strong) * E(Wind,Strong)
E(Wind,Game) = 3/5 E(Wind,Slightly) + 2/5 E(Wind,Strong)
E(Wind,Slightly) = - (2/3 $\log_2$ 2/3 + 1/3 $\log_2$ 1/3)
E(Wind,Slightly) = 0.918
E(Wind,Strong) = - (1/2 $\log_2$ 1/2+ 1/2 $\log_2$ 1/2)
E(Wind,Strong) = 1.00
E(Wind,Game) = 3/5 * 0.918 + 2/5 * 1.00
E(Wind,Game) = 0.951
Gain(Wetness,Game) = E(Game) - E(Wetness,Game)
Gain(Wetness,Game) = 0.970 – 0.951
Gain(Wetness,Game) = 0.019

Week	Wind	Game
W1	Slightly	No
W2	Strong	No
W8	Slightly	No
W9	Slightly	Yes
W11	Strong	Yes

2. Adim – Ikinci Dallanma Sonucu

Oznitelikler	Bilgi Kazanci
Temperature	0.570
Weatness	0.970
Wind	0.019

Ikinci Dallanma Sonucuna Ait Olusan Karar Agaci

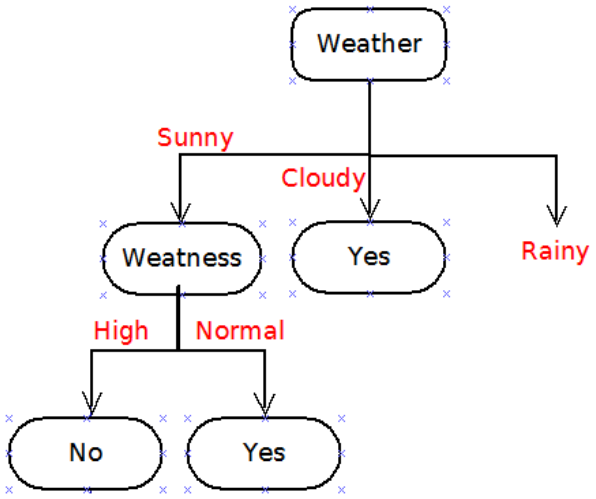


3. Adim

Week	Weather	Temperatures	Wetness	Wind	Game
W3	Cloudy	Warm	High	Slightly	Yes
W7	Cloudy	Cold	Normal	Strong	Yes
W12	Cloudy	Warmish	High	Strong	Yes
W13	Cloudy	Warm	Normal	Slightly	Yes

Hava oznitelginin bulutlu degeri icin dallanma degerleri
Game = {Yes, Yes, Yes, Yes}
P(Game,Yes) = 4/4
E(Game) = - (P (Game, Yes) $\log_2$ P (Game, Yes))
E(Game) = - (4/4 $\log_2$ 4/4)
E(Game) = 1

3. Adim – Ucuncu Dallanma Sonucu



4. Adim

Week	Weather	Temperatures	Wetness	Wind	Game
W4	Rainy	Warmish	High	Slightly	Yes
W5	Rainy	Cold	Normal	Slightly	Yes
W6	Rainy	Cold	Normal	Strong	No
W10	Rainy	Warmish	Normal	Slightly	Yes
W14	Rainy	Warmish	High	Strong	No

Hava oznitelginin yagmurlu degeri icin dallanma degerleri
Game = {Yes, Yes, No, Yes, No}
P(Game,No) = 2/5
P(Game,Yes) = 3/5
E(Game) = - (P (Game,No) $\log_2$ P (Game, No) + P (Game,Yes) $\log_2$ P (Game,Yes))
E(Game) = - (2/5 $\log_2$ 2/5 + 3/5 $\log_2$ 3/5)
E(Game) = 0.970

4. Adim – Isi niteligi Entropy - Gain Cozumu

[Temperatures,Cold] = 2
[Temperatures,Warmish] = 3
E(Game) = 0.970
E(Temp,Game) = P(Temp,Cold) * E(Temp,Cold) + P(Temp,Warmish) * E(Temp,Warmish)
E(Temp,Game) = 2/5 E(Temp,Cold) + 3/5 E(Temp,Warmish)
E(Temp,Cold) = - (1/2 $\log_2$ 1/2 + 1/2 $\log_2$ 1/2)
E(Temp,Cold) = 1
E(Temp,Warmish) = - (2/3 $\log_2$ 2/3 + 1/3 $\log_2$ 1/3)
E(Temp,Warmish) = 0.918
E(Temp,Game) = 2/5 * 1 + 3/5 * 0.918
E(Temp,Game) = 0.951
Gain(Temp,Game) = E(Game) - E(Temp,Game)
Gain(Temp,Game) = 0.970 – 0.951
Gain(Temp,Game) = 0.019

Week	Temperatures	Game
W4	Warmish	Yes
W5	Cold	Yes
W6	Cold	No
W10	Warmish	Yes
W14	Warmish	No

4. Adim – Ruzgar niteligi Entropy - Gain Cozumu

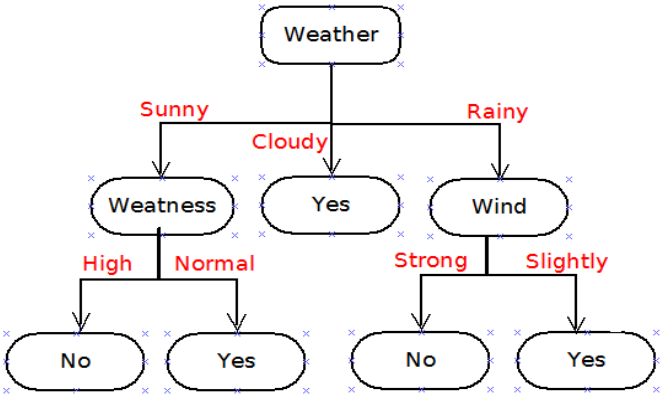
[Wind,Slightly] = 3
[Wind,Strong] = 2
E(Game) = 0.970
E(Wind,Game) = P(Wind,Slightly) * E(Wind,Slightly) + P(Wind,Strong) * E(Wind,Strong)
E(Wind,Game) = 3/5 E(Wind,Slightly) + 2/5 E(Wind,Strong)
E(Wind,Slightly) = - (3/3 $\log_2$ 3/3)
E(Wind,Slightly) = 0
E(Wind,Strong) = - (2/2 $\log_2$ 2/2)
E(Wind,Strong) = 0
E(Wind,Game) = 3/5 * 0 + 2/5 * 0
E(Wind,Game) = 0
Gain(Wetness,Game) = E(Game) - E(Wetness,Game)
Gain(Wetness,Game) = 0.970 – 0
Gain(Wetness,Game) = 0.970

Week	Wind	Game
W4	Slightly	Yes
W5	Slightly	Yes
W6	Strong	No
W10	Slightly	Yes
W14	Strong	No

4. Adim – Dorduncu Dallanma Sonucu

Oznitelikler	Bilgi Kazanci
Temperature	0.019
Wind	0.970

Dorduncu Dallanma Sonucuna Ait Olusan Karar Agaci



C4.5 UYGULAMASI

Dataset

No	Job	Age	Married	Decision
1	Doctor	70	Yes	Class1
2	Doctor	90	Yes	Class2
3	Doctor	85	No	Class2
4	Doctor	95	No	Class2
5	Doctor	70	No	Class1
6	Teacher	90	Yes	Class1
7	Teacher	78	No	Class1
8	Teacher	65	Yes	Class1
9	Teacher	75	No	Class1
10	Engineer	80	Yes	Class2
11	Engineer	70	Yes	Class2
12	Engineer	80	No	Class1
13	Engineer	70	No	Class1
14	Engineer	96	No	Class1

Age = {65, 70, 75, **80**, **85**, 90, 95, 96}
Esik Degeri => $t = (x_i + x_{i+1})/ 2$
$t = (80 + 85)/ 2$
$t = 83$
Age <= 83 ve Age > 83

No	Job	Age	Married	Decision
1	Doctor	Age <= 83	Yes	Class1
2	Doctor	Age > 83	Yes	Class2
3	Doctor	Age > 83	No	Class2
4	Doctor	Age > 83	No	Class2
5	Doctor	Age <= 83	No	Class1
6	Teacher	Age > 83	Yes	Class1
7	Teacher	Age <= 83	No	Class1
8	Teacher	Age <= 83	Yes	Class1
9	Teacher	Age <= 83	No	Class1
10	Engineer	Age <= 83	Yes	Class2
11	Engineer	Age <= 83	Yes	Class2
12	Engineer	Age <= 83	No	Class1
13	Engineer	Age <= 83	No	Class1
14	Engineer	Age > 83	No	Class1

[Decision, Class1] = 9 / 14
[Decision, Class2] = 5 / 14
E(Decision) = - (P (Decision, Class1) $\log_2$ P (Decision,Class1) + P (Decision,Class2) $\log_2$ P (Decision,Class2))
E(Decision) = - (5/14 $\log_2$ 5/14 + 9/14 $\log_2$ 9/14)
E(Decision) = 0.940

E(Decision) = 0.940
[Age<=83, Class1] = 7
[Age<=83, Class2] = 2
E(Age<=83, Decision) = - (7/9 $\log_2$ 7/9 + 2/9 $\log_2$ 2/9)
E(Age<=83, Decision)= 0.764
[Age>83, Class1] = 2
[Age>83, Class2] = 3
E(Age>83, Decision) = - (2/5 $\log_2$ 2/5 + 3/5 $\log_2$ 3/5)
E(Age>83, Decision) = 0.970
E(Age,Decision) = P(Age<=83,Decision) * E(Age<=83,Decision) + P(Age>83,Decision) * E(Age>83,Decision)
E(Age,Decision) = 9/14 E(Age<=83,Decision) + 5/14 E(Age>83,Decision)
E(Age,Decision) = 9/14 * 0.764 + 5/14 * 0.970
Gain(Age,Decision) = E(Decision) - E(Age,Decision)
Gain(Age,Decision) = 0.940 – 0.837
Gain(Age,Decision) = 0.103

MATLAB UYGULAMALAR

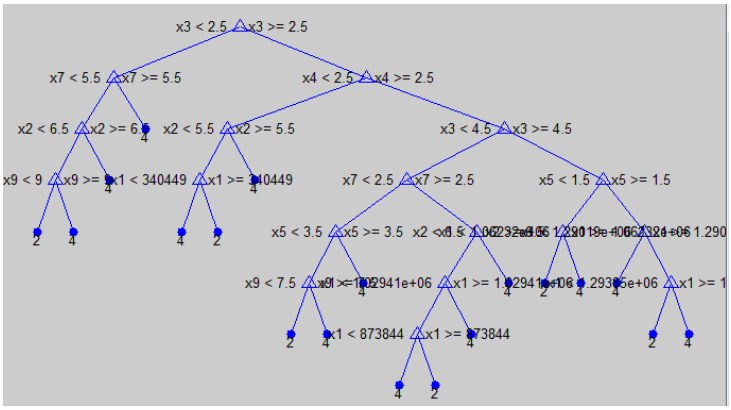
Uygulama 1 : Breast Cancer Wisconsin (Original) Data Set kullanilarak classification ve regration tree olusturuluyor.

```
#  Attribute                     Domain
-- -----------------------------------------
1. Sample code number            id number
2. Clump Thickness               1 - 10
3. Uniformity of Cell Size       1 - 10
4. Uniformity of Cell Shape      1 - 10
5. Marginal Adhesion             1 - 10
6. Single Epithelial Cell Size   1 - 10
7. Bare Nuclei                   1 - 10
8. Bland Chromatin               1 - 10
9. Normal Nucleoli               1 - 10
10. Mitoses                      1 - 10
11. Class:                       (2 for benign, 4 for malignant)
```

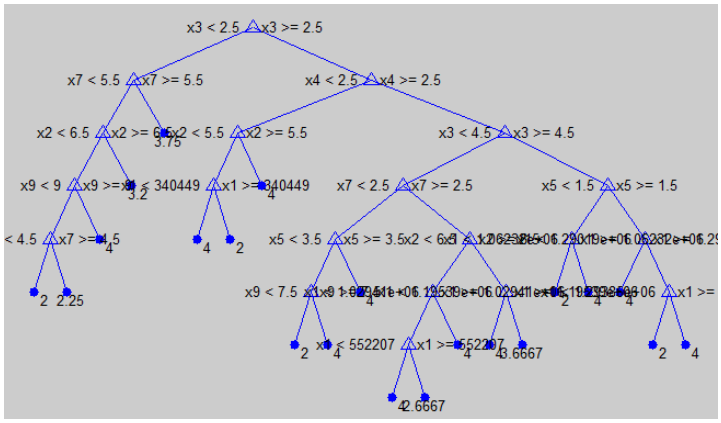
Matlab Kod

```
clear all;
close all;
clc;

dataset = load('breast-cancer-wisconsin.data');
train = dataset(:,1:10);
class = dataset(:,11);

classificationTree = fitctree(train,class)
view(classificationTree)
view(classificationTree,'mode','graph')

regressionTree = fitrtree(train,class);
view(regressionTree)
view(regressionTree,'mode','graph')
```

Classification Tree



Regression Tree



Uygulama 2 : Breast Cancer Wisconsin (Original) Data Set kullanilarak dogruluk, hata oranlari ve confusion matrix degerleri hesaplaniyor.

Matlab Kod

```
clear all;
close all;
clc;

dataset = load('breast-cancer-wisconsin.data');
dataEgitim = dataset(1:600,1:10);
dataTest = dataset(601:683,1:10);
classEgitim = dataset(1:600,11);
classTest = dataset(601:683,11);

tree = ClassificationTree.fit(dataEgitim, classEgitim)
t = classregtree(dataEgitim, classEgitim);

cvv = crossval(tree);
error = kfoldLoss(cvv)
dogruluk = 1 - error

c1 = tree.predict(dataTest);
cMat = confusionmat(classTest, c1)
```

```
error =   0.0517
dogruluk =   0.9483
cMat =   67    2
         0   14
```

Uygulama 3 : y =f(x1, x2, x3)

y	x_1	x_2	x_3
_	0	0	0
_	1	0	0
+	0	0	1
+	1	0	1
+	0	1	0
+	1	1	0
_	0	1	1
_	1	1	1

Matlab Kod

```
clear all;
close all;
clc;

x1 = [0 1 0 1 0 1 0 1]';
x2 = [0 0 0 0 1 1 1 1]';
x3 = [0 0 1 1 0 0 1 1]';
inData = [x1, x2, x3];
outData = ['-', '-', '+', '+', '+', '+', '-', '-']';

mytree = treefit(inData, outData, 'method', 'classification', 'splitmin', 2,
'prune', 'on', 'splitcriterion', 'gdi')
treedisp(mytree);

Decision tree for classification
 1  if x1<0.5 then node 2 elseif x1>=0.5 then node 3 else -
 2  if x2<0.5 then node 4 elseif x2>=0.5 then node 5 else -
 3  if x2<0.5 then node 6 elseif x2>=0.5 then node 7 else -
 4  if x3<0.5 then node 8 elseif x3>=0.5 then node 9 else -
 5  if x3<0.5 then node 10 elseif x3>=0.5 then node 11 else -
 6  if x3<0.5 then node 12 elseif x3>=0.5 then node 13 else -
 7  if x3<0.5 then node 14 elseif x3>=0.5 then node 15 else -
 8  class = -
 9  class = +
10  class = +
11  class = -
12  class = -
13  class = +
14  class = +
15  class = -
```