**Veri Madenciliği Uygulama Çalışması**

Konu:Kongresel Oylama Sonuçları(Congressional Voting)

Veritabanı Birleşik Devletler’de 27 Nisan 1984 yılında gerçekleşen bir kongrede oylamanın sonuçlarını içermektedir.

Veritabanında toplam 435 tane örnek(attribute) bulunmaktadır.Sınıf ile beraber toplam 16 tane özellik(feature) mevcuttur.Özelliklerin hepsi boolean veri tipinde cevaplanmıştır(yes/no).

Özellikler(Features):

1. Sınıf(Class) : Demokrat(Democrat),Cumhuriyetçi(Republican)
2. Handicapped infants
3. Water Project cost sharing
4. Adoption of the budget resolution
5. Physician fee freeze
6. El Salvador aid
7. Religious groups in schools
8. Anti satellite test ban
9. Aid to nicaraguan contras
10. Mx missile
11. Immigration
12. Synfuels Corporation cutback
13. Education spending
14. Superfund right to sue
15. Crime
16. Duty free exports
17. Export administration act South Africa

**Kayıp Verilerin Tespit Edilmesi ve Çözüm(Missing values and Reduction)**

Bazı özelliklerde oylamadaki kararsızlıktan ötürü veri “?” olarak gösterilmiştir. Bunlar “Missing value” terimi altında gösterilmektedir.Yukarıdaki sıraya göre özelliklerdeki kayıp veriler aşağıda belirtilmektedir

|  |  |  |  |
| --- | --- | --- | --- |
| Özellik No | Kayıp(missing) | Özellik no | Kayıp(missing) |
| 1 | 0 | 9 | 15 |
| 2 | 12 | 10 | 22 |
| 3 | 48 | 11 | 7 |
| 4 | 11 | 12 | 21 |
| 5 | 11 | 13 | 31 |
| 6 | 15 | 14 | 25 |
| 7 | 11 | 15 | 17 |
| 8 | 14 | 16 | 28 |

Ham verilerimizle yapacağımız veri madenciliği istediğimiz şekilde bize yardım etmeyebilir.Bu yüzden kayıp verilerin ideal olarak yok edilmesi en kötü durumda çeşitli tekniklerle işimize yarayacak değerlerle doldurulması gereklidir.

İlk olarak özelliklere göre cevabı çoğunlukta “?” olarak kişileri veri setinden silmeliyiz.Bir üyenin %35 ten fazla kayıp dataya sahip olanları veri setinden siliyoruz.Silme işlemi yaparken aşağıdaki Java kodu kullanılmıştır.

**package** dosyaOkumaYazma;

**import** java.io.\*;

**import** java.util.\*;

**public** **class** okuma {

**public** **static** **void** main(String[] args) {

**try**{

*fileO*();

}

**catch**(Exception e)

{

System.*out*.println("Hata\n" + e);

}

}

**public** **static** **void** fileO() **throws** IOException{

**int** temp;

**int** no=1;

Stack missing=**new** Stack();

String metin;

String path = "D:\\dataset.csv";

FileReader file = **new** FileReader(path);

BufferedReader reader = **new** BufferedReader(file);

**while**((metin =reader.readLine())!=**null**)

{

temp=0;

**for**(**int** i=0;i<metin.length();i++){

**if**(metin.charAt(i)=='?')

temp++;

}

**if**(temp>5)

missing.push(**new** Integer(no));

//reader.close();

System.*out*.println("Dosya Metni : " + metin + "\n Missing value = " + temp + "\n Attribute No: " +no );

no++;

}

**while**(!missing.empty())

{

System.*out*.println(missing.pop());

}

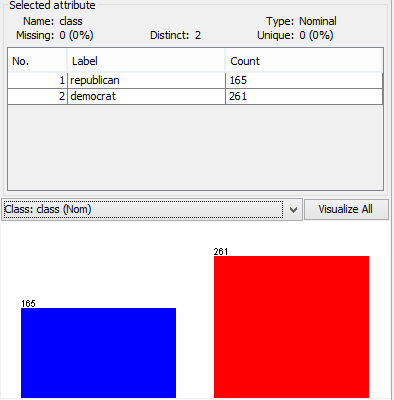
}

}

Eksik verileri tespit eden JAVA kodu

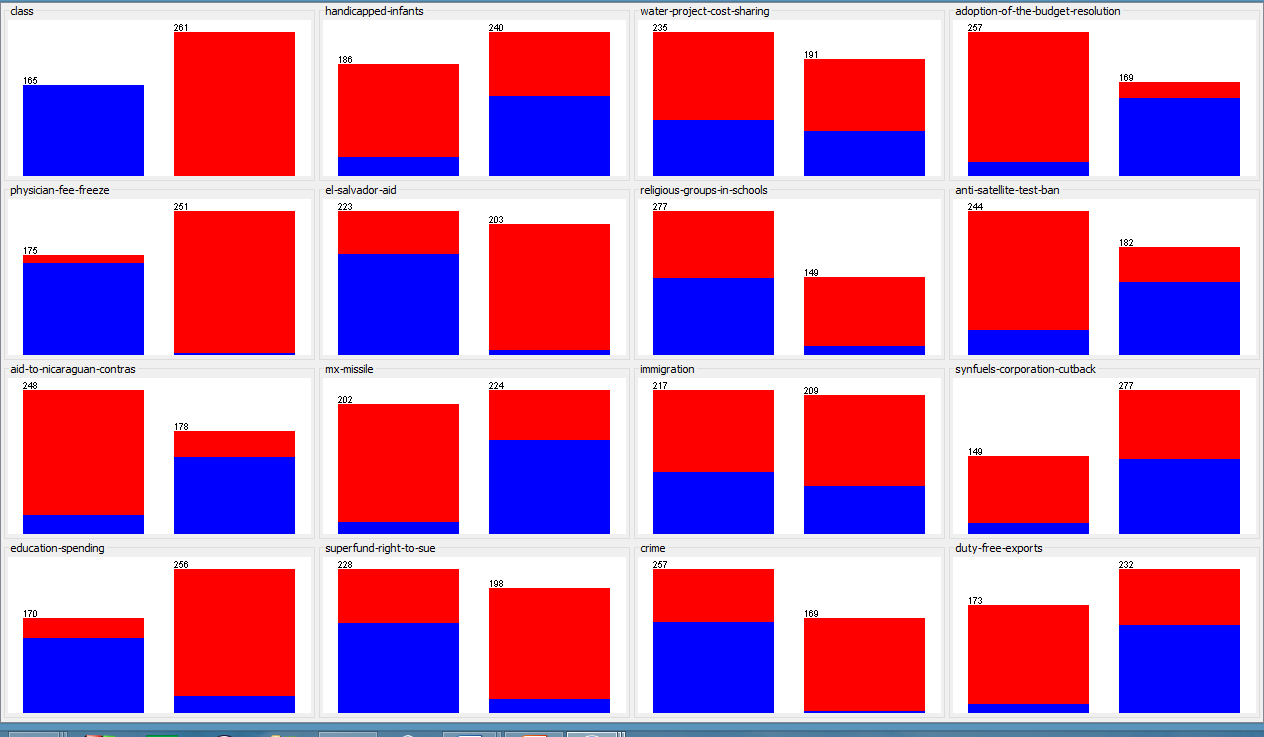
Kodumuza göre sırayla 105,108,130,181,184,249,262,296,378 numaralı örnekler veri setinden silinmiştir.

İkinci aşamada silinebilinecek özellik(feature) araştırıldı ve kayıp veri miktarı %20 den fazla olan “Export Administration act South Africa” özelliği silindi.En son Weka’da Preprocessing sekmesi altında Filter-unsupervised-attribute-ReplaceMissingValue metodu kullanılarak işlem tamamlandı.

**Sınıf Dağılımı ve Özelliklere Düşen Örnek Dağılımı**

Republican ve Democrat olmak üzere iki farklı sınıf mevcuttur.Bunlardan 165 tanesi republican,261 tanesi democrat sınıfına aittir.

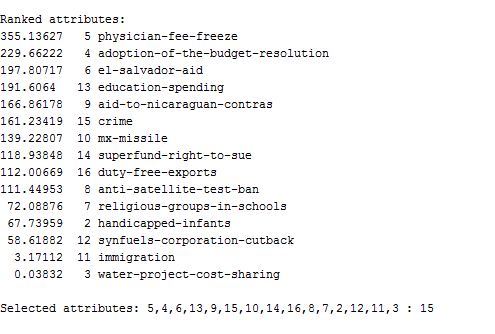
Yandaki şekilde görüldüğü gibi mavi renk republican sınıfını,kırmızı renk democrat sınıfını göstermektedir.



Yukarıdaki şekilde örneklerin özellikler üzerinde dağılımını göstermektedir.

|  |  |  |
| --- | --- | --- |
| Özellik Adı | Y | N |
| Handicapped-infants | 186 | 240 |
| Water-project-cost-sharing | 235 | 191 |
| Adoption-of-the-budget-resolution | 257 | 169 |
| Physician-fee-freeze | 175 | 251 |
| El-salvador-aid | 223 | 203 |
| Religious-groups-in-schools | 277 | 149 |
| Anti-satellite-test-ban | 244 | 182 |
| Aid-to-nicaraguan-contras | 248 | 178 |
| mx-missile | 202 | 224 |
| İmmigration | 217 | 209 |
| Synefuels-corporation-cutback | 149 | 277 |
| Education-spending | 170 | 256 |
| Superfund-right-to-sue | 228 | 198 |
| crime | 257 | 169 |
| Duty-free-exports | 173 | 232 |

**Özellik Sıralama(Feature Ranking)**



Chi Square Dağılımı metoduyla özelliklerin sınıflamada ayırt edicilikleri sıralanmıştır.

Sıralama şöyledir: Physician-fee-freeze, Adoption-of-the-budget-resolution, El-salvador-aid, Education-spending, Aid-to-nicaraguan-contras, crime, mx-missile, Superfund-right-to-sue, Duty-free-exports, Anti-satellite-test-ban, Religious-groups-in-schools, Handicapped-infants, Synefuels-corporation-cutback, İmmigration, Water-project-cost-sharing

**Sınıflama(Classification)**

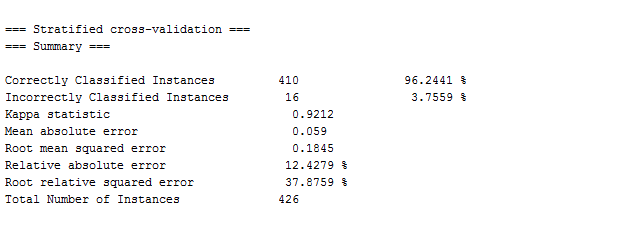
Üç farklı sınıflandırma algoritması kullanarak veri setimizin algoritmalar arasındaki başarımını ölçeceğiz.Kullanacağımız teknikler

* Decision Tree
* Bayes
* KNN

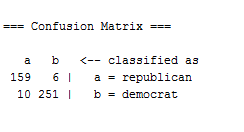
Test olarak 10 fold cross validation tekniği kullanılmıştır.

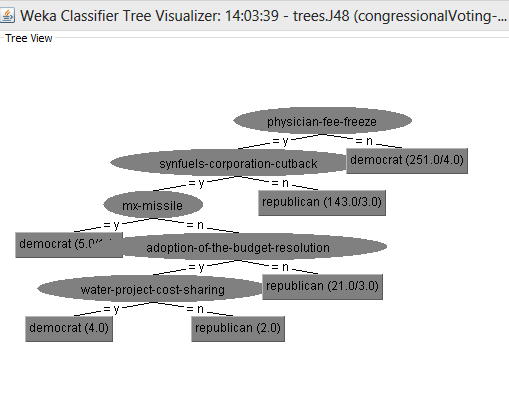
**Decision Tree Sınıflandırma**

Weka’da J48(Decision Tree) sınıflandırma tekniği seçilir.



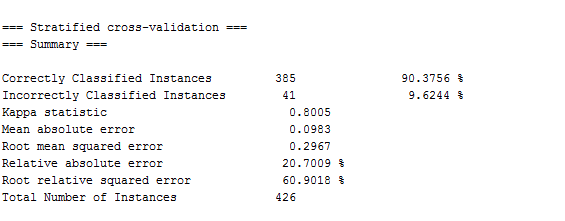
Tekniğimizde %96 başarıyla örneklerimizi sınıflandırma yapmış bulunuyor

 Sol tarafta da sınıf karışıklık matrisimiz bulunmaktadır.Yani  
 decision tree(karar ağaç,j48) algoritması hangi başarıyla sınıflandır-   
 ma yapmış görüyoruz.

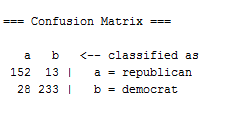


J48-decision tree grafiği

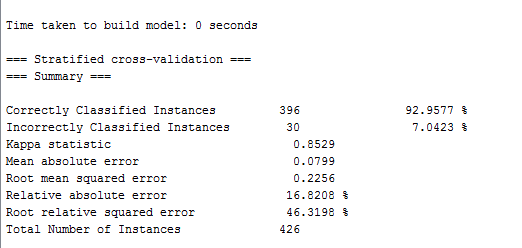
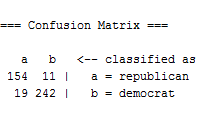
**Bayes Sınıflandırma**

****

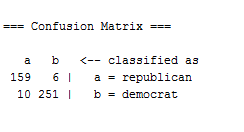
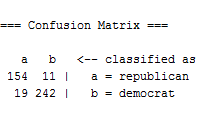
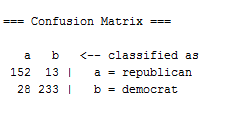
Bayes sınıflandırma tekniği ile %90 başarı ile sonuçlanmıştır.Karışıklık matrisi aşağıdaki gibidir.

  
Bayes karışıklık matrisi

**KNN Sınıflandırma**

******s K** KNN(K-Nearest Neighboor) sınıflama algoritması % 92,5 başarıyla sınıflandırma yapmıştır.Karışıklık matrisi aşağıdaki gibidir.

KNN karışıklık matrisi



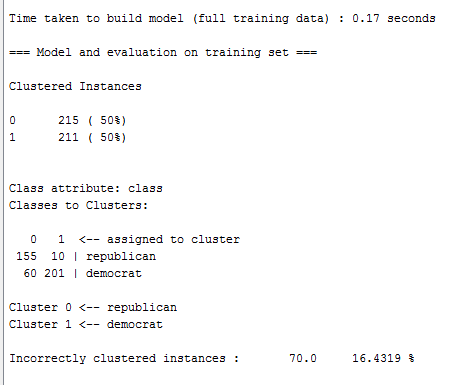
Decision Tree Bayes KNN

En başarılı sınıflandırma %96 ile decision tree(j48) sınıflandırma algoritması olmuştur.

**Kümeleme(Clustering)**

Kümeleme bir unsupervised learning tekniği yani sınıfı belli olmayan örneklere kümeleme temeline dayanan bir algoritma olup çalışmamızda üç farklı kümeleme algoritmasını veri setimize uygulayacağız ve başarı oranlarını ölçeceğiz.

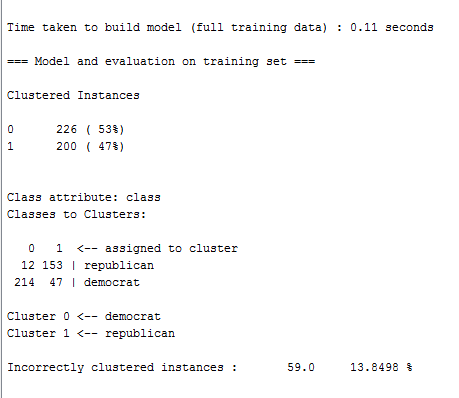
* K-Means Kümeleme
* Farthest First Kümeleme
* Estimation Maximization(EM) Kümeleme

**K-Means Kümeleme**

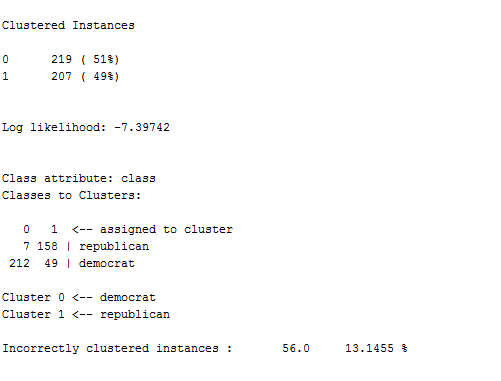
Soldaki şekilde K-means algoritması sonucu kümeleme sonuçları bulunmaktadır.Algoritmaya göre verilerin %16 sı istenildiği gibi kümelenememiştir.Aşağıdaki tabloda ise gerçekte ve hesaplanan kümelenmiş verilerin sayısı bulunmaktadır.

|  |  |  |
| --- | --- | --- |
|  | gercekte | k-means |
| Republican | 165 | 155 |
| democrat | 261 | 201 |

**Farthest First Kümeleme**

**** Algoritmaya göre verilerin % 38,96 lık kısmı istenildiği gibi kümelenememiştir. Aşağıdaki tabloda ise gerçekte ve hesaplanan kümelenmiş verilerin sayısı bulunmaktadır

|  |  |  |
| --- | --- | --- |
|  | gercekte | farthestFirst |
| Republican | 165 | 153 |
| democrat | 261 | 214 |

**Estimation Maximization(EM) Kümeleme**

Algoritmaya göre verilerin % 13,14 lık kısmı herhangi bir kümeye ait değildir. Aşağıdaki tabloda ise gerçekte ve hesaplanan kümelenmiş verilerin sayısı bulunmaktadır.

|  |  |  |
| --- | --- | --- |
|  | gercekte | em |
| Republican | 165 | 158 |
| democrat | 261 | 212 |