

Yapay Sinir Ağları Kullanarak Proje Hazırlamak

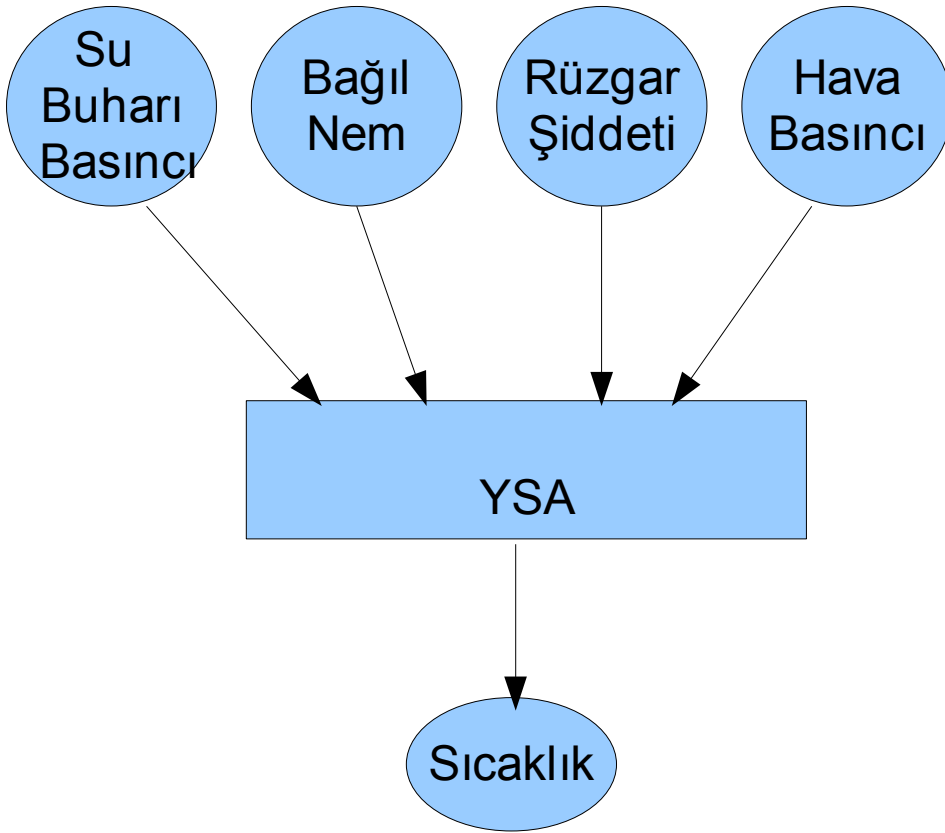
BlueKid
Derin Deli Mavi

Giriş:

Bu Belgenin hazırlanış amacı; Tez, Ödev, Proje,... vb bir iş için Yapay Sinir Ağlarını kullanmak isteyen ve Fakat tek satır program yazmak istemeyenlere yol göstermektir. Konuyu anlatmak için örnek bir proje hazırlanacaktır.

Örnek Proje Konumuz : Eksik veri tamamlama, İstanbul Göztepe meteoroloji istasyonunda belirli bir tarihten sonra sıcaklık parametresine ait verilerde eksiklik olduğu tespit ediliyor. Diğer parametreler (su buharı basıncı, bağıl nem, rüzgar şiddeti, hava basıncı) yardımıyla YSA kullanılarak Sıcaklık parametresinin tahmin edilmesi gerekiyor.

Proje Konumuzu “Yapay Sinir Ağları İlkeleri” adlı Zekai ŞEN Hocanın kitabından seçtik.



Projemiz için şekildeki gibi bir YSA dizayn etmemiz gerekiyor. Yani

Giriş değerlerimiz

- Su Buharı Basıncı,
- Bağıl Nem,
- Rüzgar şiddeti,
- Hava Basıncı

Çıkış Değerimiz

- Sıcaklık

Veri Hazırlama :

Veri hazırlama işlemi için Bir Hesap Tablolama programına ihtiyacınız olacak Biz bu proje de Open Office programlarını tercih ettik. Projenin konusunu aldığımız kitapta 50 adet veri vardı. Bu veriyi iki parçaya ayırıyoruz. İlk kısmı 33 adettten oluşan *Eğitim verisi*, kalanları Eğiteceğimiz YSA yı test etmek için kullanacağımız *Test Verisi oluyor*.

| Ham Eğitim Verileri | | | | | |
|---------------------|---------------------|--------------------------|------------------|-----------------------|-----------------|
| Min | 1000 | 3 | 50 | 1 | 0 |
| Max | 1025 | 13 | 100 | 6 | 15 |
| VeriNo | <i>Hava Basıncı</i> | <i>Su Buharı Basıncı</i> | <i>Bağıl Nem</i> | <i>Rüzgar Şiddeti</i> | <i>Sıcaklık</i> |
| 1 | 1008,0 | 5,8 | 82,3 | 4,6 | 3,6 |
| 2 | 1004,1 | 5,3 | 68,3 | 1,5 | 2,9 |
| 3 | 1005,3 | 5,6 | 77,0 | 2,4 | 2,4 |
| 4 | 1012,5 | 4,9 | 65,0 | 1,6 | 3,3 |
| 5 | 1010,2 | 5,3 | 71,7 | 3,9 | 4,2 |
| 6 | 1012,0 | 5,0 | 64,7 | 5,3 | 3,5 |
| 7 | 1016,9 | 6,1 | 74,7 | 4,3 | 3,3 |
| 8 | 1021,6 | 6,4 | 82,5 | 3,6 | 2,3 |
| 9 | 1020,1 | 6,7 | 84,3 | 3,2 | 2,3 |
| 10 | 1016,0 | 6,7 | 95,3 | 4,1 | 2,4 |
| 11 | 1017,1 | 6,6 | 81,3 | 3,7 | 5,1 |
| 12 | 1014,5 | 7,8 | 74,0 | 4,9 | 6,5 |
| 13 | 1012,4 | 7,6 | 80,0 | 3,5 | 6,3 |
| 14 | 1013,7 | 7,9 | 79,3 | 4,3 | 6,2 |
| 15 | 1014,9 | 7,8 | 83,3 | 3,5 | 6,3 |
| 16 | 1012,4 | 8,1 | 82,3 | 3,4 | 5,1 |
| 17 | 1007,2 | 6,8 | 92,0 | 3,1 | 3,1 |
| 18 | 1011,9 | 5,9 | 89,0 | 2,8 | 3,2 |
| 19 | 1013,5 | 6,8 | 77,3 | 2,7 | 4,5 |
| 20 | 1012,7 | 7,1 | 80,7 | 2,9 | 5,5 |
| 21 | 1013,4 | 7,0 | 77,7 | 3,2 | 5,1 |
| 22 | 1014,9 | 7,3 | 78,3 | 4,5 | 4,7 |
| 23 | 1015,5 | 7,1 | 83,0 | 3,8 | 4,7 |
| 24 | 1018,8 | 7,1 | 72,3 | 3,6 | 5,3 |
| 25 | 1013,3 | 10,1 | 92,7 | 2,1 | 7,8 |
| 26 | 1016,6 | 7,3 | 77,7 | 1,2 | 5,7 |
| 27 | 1014,2 | 7,8 | 80,0 | 1,7 | 8,1 |
| 28 | 1012,5 | 10,8 | 70,7 | 2,0 | 12,6 |
| 29 | 1002,3 | 11,8 | 72,3 | 1,8 | 11,7 |
| 30 | 1004,0 | 9,7 | 91,0 | 1,5 | 8,2 |
| 31 | 1006,9 | 9,6 | 88,7 | 2,7 | 12,9 |
| 32 | 1012,0 | 8,7 | 81,7 | 3,6 | 8,9 |
| 33 | 1016,3 | 8,5 | 73,6 | 3,0 | 9,2 |

Evet Tablomuzu hazırladık. mavi sütunlar giriş değerlerini kırmızı sütun ise çıkış değerini gösteriyor. Sütunların üzerinde minumum ve maksimum değerleri yazıyor. Yani elimizdeki veriler bu aralık içinde. YSA' nın bu aralıkların çok ötesindeki durumlar için mantıklı tahminde bulunmasını beklemeyin. Dikkat ettiyseniz tablonun üzerinde Ham Eğitim Verisi yazıyor. Biz bu verileri doğrudan YSA üzerinde kullanamıyoruz maalesef. YSA lar katmanlar arasında geçişlerde Eşik fonksiyonlarından geçerler. Ve bu eşik fonksiyonlarının çıkış aralıkları (0,1) Ya da (-1 , 1) aralığında olur. Ham verileri YSA üzerinde kullanmak için verilerin bu aralıklara çekilmesi gerekiyor. Formül Gayet Basit

$$\text{Normalize Edilmiş Veri} = (\text{Ham Veri} - \text{Min}) / (\text{Max} - \text{Min})$$

Bu Formülü kullanarak Kolayca Kullanabileceğimiz verilere ulaşıyoruz

| VeriNo | Hava Basıncı | Su Buharı Basıncı | Bağıl Nem | Rüzgar Şiddeti | Sıcaklık |
|--------|--------------|-------------------|-----------|----------------|----------|
| 1 | 0,3200 | 0,2800 | 0,6460 | 0,7200 | 0,2400 |
| 2 | 0,1640 | 0,2300 | 0,3660 | 0,1000 | 0,1933 |
| 3 | 0,2120 | 0,2600 | 0,5400 | 0,2800 | 0,1600 |
| 4 | 0,5000 | 0,1900 | 0,3000 | 0,1200 | 0,2200 |
| 5 | 0,4080 | 0,2300 | 0,4340 | 0,5800 | 0,2800 |
| 6 | 0,4800 | 0,2000 | 0,2940 | 0,8600 | 0,2333 |
| 7 | 0,6760 | 0,3100 | 0,4940 | 0,6600 | 0,2200 |
| 8 | 0,8640 | 0,3400 | 0,6500 | 0,5200 | 0,1533 |
| 9 | 0,8040 | 0,3700 | 0,6860 | 0,4400 | 0,1533 |
| 10 | 0,6400 | 0,3700 | 0,9060 | 0,6200 | 0,1600 |
| 11 | 0,6840 | 0,3600 | 0,6260 | 0,5400 | 0,3400 |
| 12 | 0,5800 | 0,4800 | 0,4800 | 0,7800 | 0,4333 |
| 13 | 0,4960 | 0,4600 | 0,6000 | 0,5000 | 0,4200 |
| 14 | 0,5480 | 0,4900 | 0,5860 | 0,6600 | 0,4133 |
| 15 | 0,5960 | 0,4800 | 0,6660 | 0,5000 | 0,4200 |
| 16 | 0,4960 | 0,5100 | 0,6460 | 0,4800 | 0,3400 |
| 17 | 0,2880 | 0,3800 | 0,8400 | 0,4200 | 0,2067 |
| 18 | 0,4760 | 0,2900 | 0,7800 | 0,3600 | 0,2133 |
| 19 | 0,5400 | 0,3800 | 0,5460 | 0,3400 | 0,3000 |
| 20 | 0,5080 | 0,4100 | 0,6140 | 0,3800 | 0,3667 |
| 21 | 0,5360 | 0,4000 | 0,5540 | 0,4400 | 0,3400 |
| 22 | 0,5960 | 0,4300 | 0,5660 | 0,7000 | 0,3133 |
| 23 | 0,6200 | 0,4100 | 0,6600 | 0,5600 | 0,3133 |
| 24 | 0,7520 | 0,4100 | 0,4460 | 0,5200 | 0,3533 |
| 25 | 0,5320 | 0,7100 | 0,8540 | 0,2200 | 0,5200 |
| 26 | 0,6640 | 0,4300 | 0,5540 | 0,0400 | 0,3800 |
| 27 | 0,5680 | 0,4800 | 0,6000 | 0,1400 | 0,5400 |
| 28 | 0,5000 | 0,7800 | 0,4140 | 0,2000 | 0,8400 |
| 29 | 0,0920 | 0,8800 | 0,4460 | 0,1600 | 0,7800 |
| 30 | 0,1600 | 0,6700 | 0,8200 | 0,1000 | 0,5467 |
| 31 | 0,2760 | 0,6600 | 0,7740 | 0,3400 | 0,8600 |
| 32 | 0,4800 | 0,5700 | 0,6340 | 0,5200 | 0,5933 |
| 33 | 0,6520 | 0,5500 | 0,4720 | 0,4000 | 0,6133 |

YSA işlemleri için Pek tabii ki favori programımız [FANNTool](#) ' u kullanacağız. FANN kütüphanesinin ve dolayısıyla FANNTool'un veri dosyaları için kullandığı format gayet basit.

Veri dosyaları Basit bir Text dosyası, mesela *Not Defteri* programıyla kolayca hazırlayabilirsiniz

- Dosyanı ilk satırında Veri sayısı Giriş Parametre Sayısı Çıkış Parametre sayısı aralarında boşluk bırakılarak yazılır
- Dosyanın İkinci satırında İlk Verinin Giriş Parametre değerleri aralarında boşluk bırakılarak yazılır
- Dosyanın Üçüncü satırında İlk verinin Çıkış parametre değerleri aralarında boşluk bırakılarak yazılır.
- Sonraki satırlarda 2. ve 3. satırlar gibi veriler bitinceye kadar sırayla yazılır.

Aşağıda Eğitim veri dosyasının ilk satırlarını görüyorsunuz

```
33 4 1
0.3200 0.2800 0.6460 0.7200
0.2400
0.1640 0.2300 0.3660 0.1000
0.1933
```

...

Bu şekilde hazırladığımız Eğitim veri dosyasını “Sıcaklık-Eğitim.dat” adıyla kaydediyoruz. Yukarıda anlattığımız işlemleri Test için ayırdığımız veriler içinde uygulayıp “Sıcaklık-Test.dat” adıyla kaydediyoruz.

Eğitim :

Artık eğitim ve test verilerimiz hazır olduğuna göre FannTool'u çalıştırabiliriz. FannTool'u çalıştırın ve “Training Data File” olarak “Sıcaklık-Eğitim.dat” dosyasını “Testing Data File” olarak “Sıcaklık-Test.dat” dosyasını seçin.

| | |
|--------------------|---------------------------------|
| Training Data File | C:/Sıcaklık/Sıcaklık-Eğitim.dat |
| Testing Data File | C:/Sıcaklık/Sıcaklık-Test.dat |

Bundan sonrası sizin artık edebiyat kuvvetinize kalmış :) FannTool'un size sağladığı imkanları kullanarak Envai çeşit parametre değişikliklerinin Eğitim ve test süreçlerine olan etkilerini ballandıra ballandıra anlatabilirsiniz.

Peki FannTool ile ne gibi değişik parametreleri değiştirebilirsiniz. Hepsini anlatmamız mümkün değilse de önemli bir kısımdan bahsedelim.

- Saklı katman ve katmanlardaki hücre sayısını değiştirerek YSA'nın geometrisini değiştirebilirsiniz mesela örneğimizdeki YSA'yı 3 katmanlı ve saklı katmanda 3 hücreli olarak şöyle gösterebiliriz (4 -> 3 -> 1)

| | | | | | | | |
|------------|---|----------------|---|----------------|---|----------------|---|
| # of Layer | 3 | Hid. Layer 1 : | 3 | Hid. Layer 2 : | 1 | Hid. Layer 3 : | 1 |
|------------|---|----------------|---|----------------|---|----------------|---|

of Layer : Katman sayısı = Giriş + Saklı (Minimum 1 Maksimum 3) + Çıkış

Hid Layer 1 : 1. Saklı katmandaki Nodül Sayısı

Hid Layer 2 : 2. Saklı katmandaki Nodül Sayısı (# of Layer >= 4)

Hid Layer 3 : 3. Saklı katmandaki Nodül Sayısı (# of Layer = 5)

Aslında FANN kütüphanesi 5 katmanlıdan fazlası için destek veriyor. Fakat pratikte pek faydalı olmadığından kullanmadım yani FannTool ile En fazla 3 saklı katman tanımlayabilirsiniz.

- Değişik eğitim metodlarını (Trainin Metod) kullanabilirsiniz. FANN kütüphanesinde halen 4 çeşit algoritma vardır bunlar
 - FANN_TRAIN_INCREMENTAL** : standart Geri yayılma algoritmasıdır. Her eğitim verisi sonunda ağırlıklar güncellenir.
 - FANN_TRAIN_BATCH** : standart Geri yayılma algoritmasıdır. Bütün eğitim verileri uygulanıp MSE hesaplandıktan sonra ağırlıklar güncellenir.
 - FANN_TRAIN_RPROP** : batch training metodunun çok daha gelişmişidir, özelliği gereği öğrenme hızını kullanmaz. Kendi ayar parametrelerinin değiştirilmesi de metodun işleyişini bilmeyen kullanıcılara tavsiye edilmez. Bu metod ilk olarak 1993 yılında Riedmiller ve Braun, tarafından geliştirilmiştir. FANN da kullanılan iRPROP-ise Igel ve Husken, tarafından – 2000 - geliştirilmiş bir RPROP çeşididir
 - FANN_TRAIN_QUICKPROP** : Fahlman, tarafından 1988 geliştirilmiş bir eğitim metodudur.
- Saklı ve Çıkış Katmanlarındaki Eşik Foksiyonlarını (Activation Functions)

- deęiřtirebilirsiniz
- Öğrenme Hızını ve Momentum deęerleri ile oynaya bilirsiniz

Eęitim sonunda YSA'nın MSE (Ortalama Hatanın karesi) deęeri arzu ettięimiz deęerin altına inince YSA mızı kaydediyoruz. “Sıcaklık.net”

Test :

Eęitilmiş ve Kaydedilmiş YSA Test verisi ile alıřtırılır ve sonuta ulařılan MSE deęerine bakılır. Bu adımın amacı Eęitimin doęru olup olmadıęını kontrol etmektir. Eęer Test sonucunda elde ettięimiz MSE deęeri ok yksekse YSA'mız Eęitimi dzgn deęildir. Yazının sonunda verceęimiz rnekle konu daha net olacaktır řimdilik řu kadarını syleyelim test datanız olmadan YSA nın eęitimi konusunda bir řey sylemek mmkn deęildir...

alıřtırma :

“Run with File“ Seeneęi ile Daha nceden eęittięimiz YSA Test veri dosyası kullanılarak alıřtırılır ve sonular alınır

```
Trained ANN Loaded...
Runing with Testing Data File...
Data No : Calculated Outputs : Real Outputs
0001 : 0.7407 : 0.7933
0002 : 0.5307 : 0.5267
0003 : 0.5106 : 0.4733
0004 : 0.4349 : 0.4400
0005 : 0.5440 : 0.4133
0006 : 0.5581 : 0.3800
0007 : 0.3825 : 0.4267
0008 : 0.6411 : 0.6067
0009 : 0.5731 : 0.5867
0010 : 0.4395 : 0.5733
0011 : 0.4461 : 0.4867
0012 : 0.3673 : 0.7467
0013 : 0.6908 : 0.7467
0014 : 0.3819 : 0.5800
0015 : 0.3846 : 0.5467
0016 : 0.5732 : 0.3867
0017 : 0.8208 : 0.5600
```

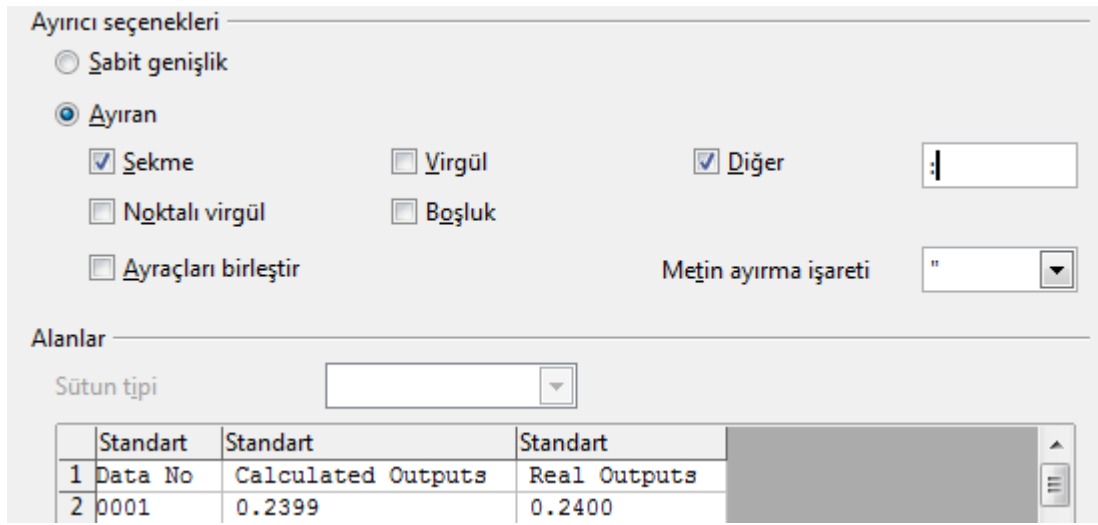
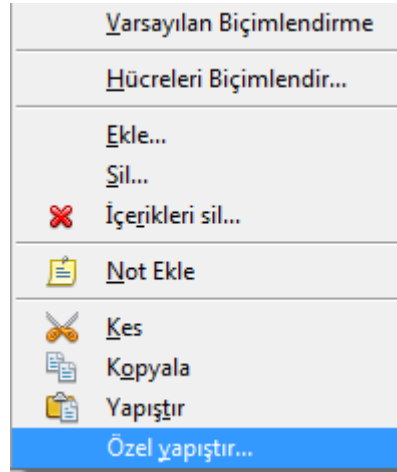
Data No : Veri sıra No

Calculated Outputs : YSA nın rettięi sonular

Real Outputs : Gerek Sonular

ıkan sonuları save log ile bir metin dosyayaasına kaydediyoruz. '.' Noktaları ',' ile deęiřtiriyoruz. Sonra verileri Tablolama programımıza aktarıyoruz. nk sonular grdęnz gibi Normalize edilmiř halde. Normalize iřleminin tersini yapıp gerek deęerleri hesaplıyoruz ve Grafikler izdiriyoruz

Text Dosyasında setięimiz veriyi. OpenOffice de veriyi aktaracaęımız kısma gelip zel yapıřtır seeneęinden



ayıran kısmına daki diğer seçeneğine ':' yazıp tamam diyerek aktarıyoruz

Bundan sonrası Veri hazırlamadaki formülün tersini uygulayarak Normalize edilmiş sonuç değerlerini Sonuç değerlerine çevirmek.

$$Sonuç = Normalize Sonuç * (Max - Min) + Min$$

Bundan sonra Hesaplanan Sonuç değerleri ile Gerçek Sonuç değerlerinin karşılaştırmasını yapmak için grafik çizdiriyoruz.

Unutmadan hatırlatayım. Gerek Normalize ederken gerek tersine çevirirken kullandığımız formüller verilerimizi (0 - 1) aralığına çeker. Eğer (-1 1) aralığını kullanmak istersek

$$Normalize Edilmiş Veri = [2 * (Ham Veri - Min) / (Max - Min)] - 1$$

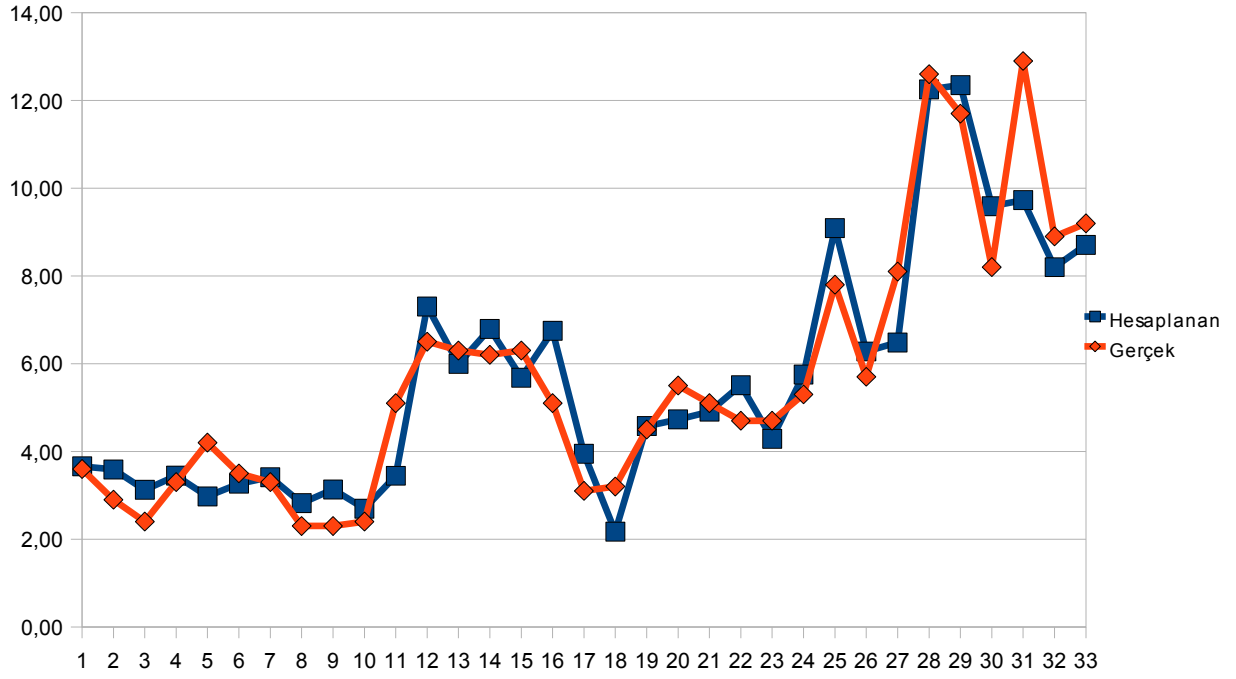
$$Sonuç = [(Normalize Sonuç + 1) * (Max - Min)] / 2 + Min$$

şeklinde olur. (Yani Umarım :) öyle oluyordur.....)

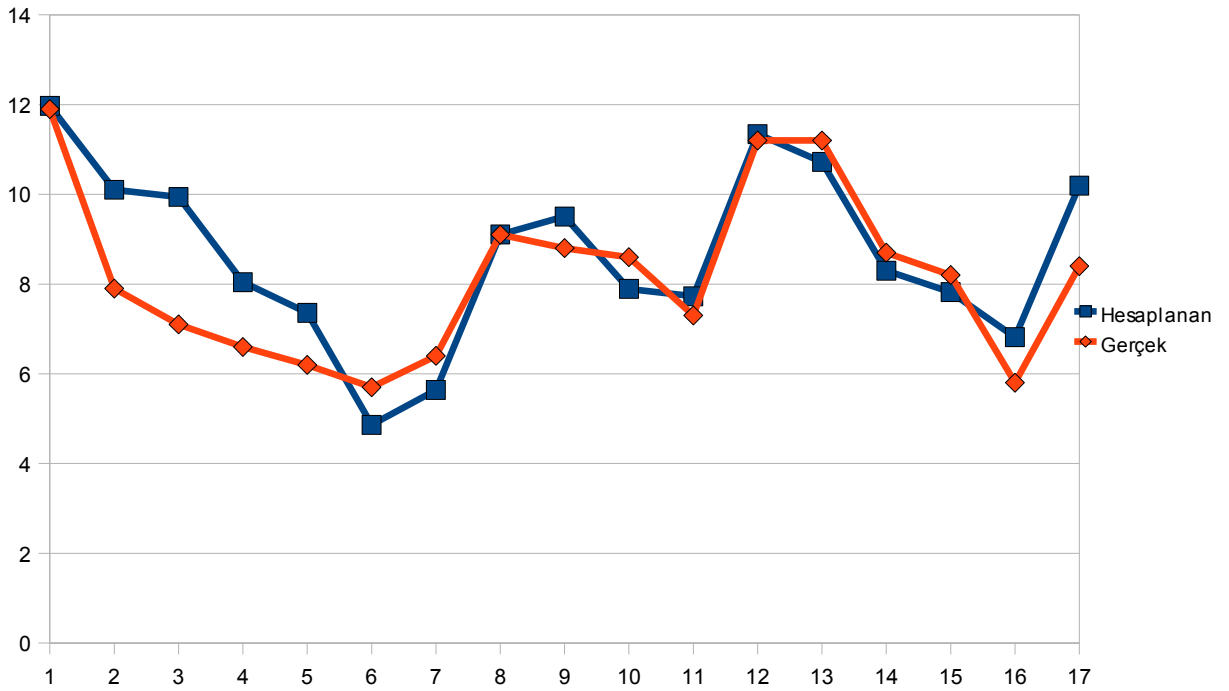
Bir başka hatırlatmada Ondalık ayracı konusunda FannTool Ondalık ayracı olarak nokta kullanıyor. Hesap tablolama programların Türkçe olanları virgül kullanıyor.' yani verileri Hesap tablosundan FannTool'a aktarırken 'Virgülleri' nokta olarak değiştirin. FannToolun ürettiği sonuçları hesap tablosunda kullanmak istediğinizde Noktalar'ı virgül ile değiştirin.

Sonuçlar :

“Sıcaklık.net” YSA sınıfın Eğitim Verileri ile Çizdirilmiş grafiği

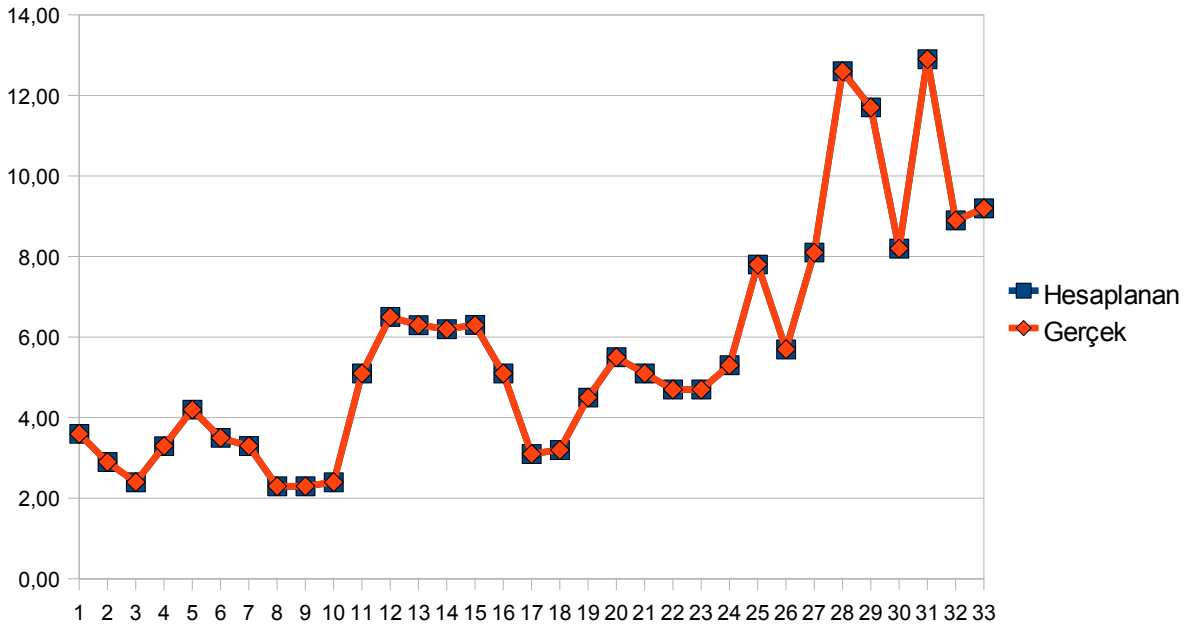


Test Verileri ile Çizdirilmiş Grafik



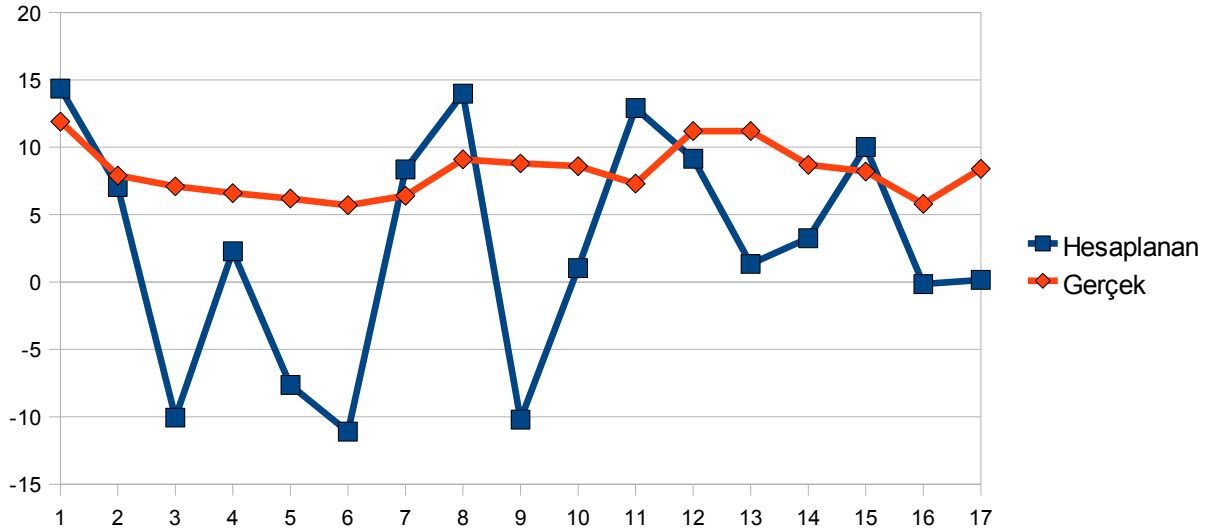
Aynı Eğitim verilerini kullanarak başka bir YSA daha eğitiyoruz, ve “Sıcaklık-Overfit.net” ismiyle kaydediyoruz. Hem de mükemmel eğitiyoruz.

“Sıcaklık-Overfit.net” YSA sınıfın Eğitim Verileri ile Çizdirilmiş grafiği



Süper bir eğitilmiş bir YSA ! hesaplanan değerler ile Gerçek değerler çıkmış. Biri de test verilerine bakalım

Test Verileriyle Çizdirilmiş Grafik



Ama Test verileriyle aynı YSA kullanıldığında elde ettiğimiz sonuç rezalet. Bunun sebebinden kısaca bahsedersek. Eğitim ilk bakışta zannedildiği gibi minimum hata değerine ulaşmak değildir. Kabul edilebilir bir hata oranıyla bir genelleme yapılabilir hale gelmektir. Tıpkı İnsanda olduğu gibi Eğitimin iyisi ezbercilikle olmaz. Bu ikinci YSA tabiri caiz ise Eğitim verilerini ezberlemiştir. Test verilerindeki kötü sonucun sebebidir budur.

Sonsöz :

Yapay Sinir Ağlarının uygulaması pek çok alanda mümkün olmasına rağmen nadiren kullanılıyorlar. Buraya kadar zahmet edip yazdıklarımızı okumuşsanız. YSA ile proje yapmanın hiçte sanıldığı kadar zor olmadığınıda anlamış olmanız lazım. Yazıda Verileri hazırladığımız Hesap Tablosu, Eğitim ve Test veri dosyaları, Eğitilmiş YSA dosyaları bu yazıyla birlikte dağıtılmaktadır. Bunları inceleyip pratik yapmanız tavsiye edilir.

Eğer anlaşılmayan bir kısım, hata veya eksiklik varsa bize bildirirseniz seviniriz. Son olarak bu konuda çalışma yapmak isteyenlere bir nebze faydamız olabilirse ne mutlu bize...

BlueKid