

ELEKTRİK DAĞITIM ŞEBEKELERİNDE TALEP TAHMİNİ

Mehmet ÖZEN¹

e-posta: ozenmehmet.92@gmail.com

Ömer GÜL¹

e-posta: enerjikalitesi@gmail.com

¹İstanbul Teknik Üniversitesi, Elektrik-Elektronik Fakültesi, Elektrik Mühendisliği Bölümü, 34469, Maslak, İstanbul, Türkiye

ÖZET

Elektrik dağıtım şebekelerinde talep tahmini, dağıtım şirketleri tarafından belirli bir zaman aralığı için farklı senaryolar çerçevesinde geleceğe dönük olarak yapılan dağıtılan elektrik enerjisi tahminidir. Talep tahminleri kısa, orta ve uzun dönemli olarak üç dönemde incelenir. Bu çalışmada ise kısa dönemli tahminler üzerinde durularak bölgeye(indirici merkezlerinin toplam yüküne) ve ayrı ayrı indirici merkezlere dayalı olarak iki farklı tahmin modeli oluşturulmuştur. Bu iki model karşılaştırılarak tahmin hatalarının düşürülmesi amaçlanmıştır. Bu çalışmalar ise kısa dönemli tahminlerde oldukça başarılı sonuç veren Yapay Sinir Ağları(YSA) ile yapılmıştır. Çalışmanın sonucunda ise indirici merkezlere dayalı olarak yapılan tahminlerin daha başarılı olduğu görülmüştür.

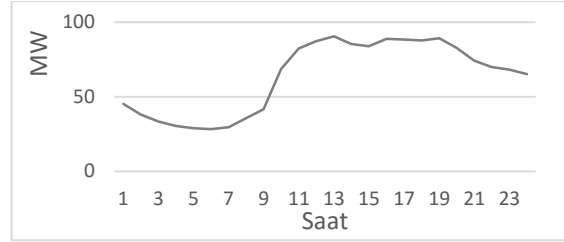
Anahtar kelimeler: talep tahmini, yük tahmini, yapay sinir ağları

I. GİRİŞ

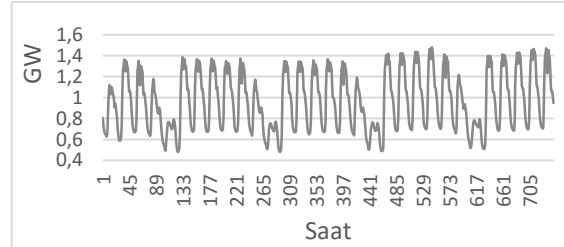
Kısa dönemli tahminler 1 saatten 1 haftalık süreci içine almaktadır. Bu tahminler sayesinde elektrik güç sistemlerinin verimli ve sağlıklı çalışması sağlanmaktadır. Ayrıca güç sistemlerinin işletilmesinde kısa dönemli yük tahminleri oldukça önemlidir. Gün içerisinde elektrik enerji ihtiyacının sürekli olarak değişmesinden dolayı devreye giren ya da devreden çıkan generator sayısı değişmektedir.

Enerji ihtiyacının belirlenmesi talep tahmin modelleriyle yapılmaktadır. Güç sistemlerinin tahminlere göre işletilmesi ile kullanıcılara ekonomik ve kaliteli enerji sağlanabilir. Verimli enerji planlaması milyon dolarları kurtarabilir ve bu da ülkenin ekonomisinin büyümesinde önemli rol oynamaktadır[1].

İyi bir tahmin modeli oluşturmak için öncelikle yük karakteristiklerini incelemek gerekmektedir. Yük karakteristikleri yıla, mevsime, güne, saate ve hava koşullarına bağlı olarak değişim göstermektedir. Buna örnek olarak Şekil 1 ve Şekil 2 gösterilebilir.



Şekil 1: Bir İndirici Merkezin Günlük Yük Eğrisi



Şekil 2: Toplam Yükün Aylık Yük Eğrisi

Yük eğrilerinin denklemini Bilge aşağıdaki gibi ifade etmiştir[2].

$$L = L_n + L_w + L_s + L_r \quad (1)$$

Bu denklemde bulunan, L toplam sistem yükünü, L_n yıl boyunca görülmekte olan yükün normal bölümünü, L_w yükün mevsime ve hava koşullarına bağlı kısmını, L_s özel bir olaydan dolayı yük davranışındaki sapmayı ve L_r yükün değişimindeki açıklanamayan kısmı temsil

etmektedir. Bu denklemde bulunan bileşenlerin her bireri kendi içinde non-lineer olabilir ve bu da analitik olarak tahmin yapmayı zor ve karmaşık bir hale getirmektedir.

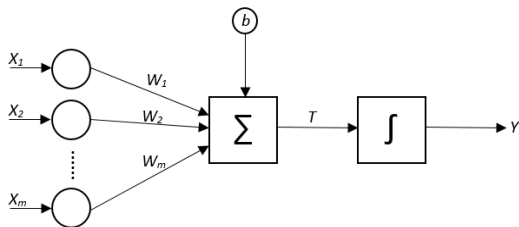
Son yıllarda yapılan çalışmalar incelendiğinde, YSA'nın kısa dönemli tahminlerde oldukça başarılı olduğu görülmüştür [3], [4], [5], [6], [7]. YSA doğrusal olmayan girdi ve çıktı vektörleri arasındaki bağlantıyı öğrenme yeteneğine sahiptir. Yani karmaşık sistemlerde bile istenen sonuçları üretebilmektedir.

II. YAPAY SINİR AĞLARI

YSA teknolojisi, bilgisayar ortamında insan beyninden ve sinir sisteminden esinlenerek oluşturulmuş modellerdir. Bu yaklaşımda amaç insan beyninin temel özelliklerinden biri olan öğrenme yeteneğini bilgisayar ortamında gerçekleştirmektir[8].

YSA bir sisteme ilişkin girdi ve çıktıları analiz ederek bunlar arasındaki ilişkiden faydalanarak kendini eğitir. Eğitimi gerçekleştirdiğinde, yeni durumlar karşısında hızlı bir şekilde karar verme yeteneğine sahip bir yapıdadır.

YSA'nın temelinde yapay sinir ağı hücresi(YSH) yatmaktadır. Biyolojik sinir hücresine benzetilerek oluşturulan bu yapı genel olarak, girişler(X), çıkışlar(Y), ağırlıklar(W), toplama fonksiyonu ve aktivasyon fonksiyonundan oluşmaktadır.



Şekil 3: Yapay Sinir Hücresi

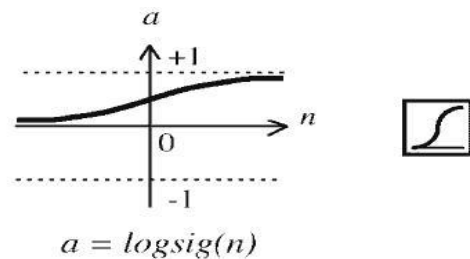
Girişler, dışarıdan ve kendinden önceki sinirlerden aldıkları verileri sinir ağına getirmekle sorumludurlar.

Ağırlıklar, girişler sayesinde alınan verilerin sinir üzerindeki etkisini belirleyen katsayılardır. Her giriş kendine ait bir ağırlığa sahiptir. Ağırlık katsayısı büyük olan girişler YSH'ye güçlü bağlanırken, ağırlığı düşük olan girişler zayıf bağlanır. Buradan anlaşılacağı üzere ağırlık katsayısı büyük olan girişlerin önemli olduğu anlamına gelirken, ağırlık katsayısı düşük olan girişlerin az önemli olduğu anlamına gelmektedir.

Girişler ile gelen bilgiler kendi ağırlıkları ile çarpılarak toplanmaktadır. Ancak bu işlem yapılırken aktivasyon potansiyeli(b) ile de çarpılmaktadır. Toplama fonksiyonu(Σ) bu işlemi yaparak toplamları(T) aktivasyon fonksiyonuna göndermektedir.

$$T = \sum_{j=1}^m (w_{1j} \cdot bx_j) \quad (2)$$

Aktivasyon fonksiyonu sonuç elde etmede yapay sinir hücresindeki son aşamadır ve girdi-çıkı değerleri arasındaki eğrisel eşleşmeyi sağlamaktadır. Aktivasyon fonksiyonunun doğru seçilmesi ağ performansını önemli ölçüde etkilemektedir[9]. Yaygın olarak kullanılan aktivasyon fonksiyonlarından logaritmik sigmoid fonksiyonu Şekil 4'te görülmektedir.



Şekil 4: Logaritmik Sigmoid Fonksiyonu

Logaritmik sigmoid fonksiyonu, lineer ve lineer olmayan kesin artan bir fonksiyon olarak tanımlanmaktadır.

YSH'nin çıkışları ise aktivasyon fonksiyonundan alınan çıkış değeridir. Çıkışlar kendinden sonra gelen YSH'nin giriş değeri olabilir.

YSA katman sayısına göre tek ve çok katmanlı olarak sınıflandırılırken, yapılarına göre ise ileri ve geri beslemeli olarak sınıflandırılmaktadır. İleri beslemeli sinir ağlarında tek yönlü bağlantılar kullanılırken, geri beslemeli sinir ağlarında hem ileri yönlü hem de geri yönlü bağlantılar vardır. Bu yüzden geri beslemeli sinir ağlarında gecikmeler meydana gelmektedir[10].

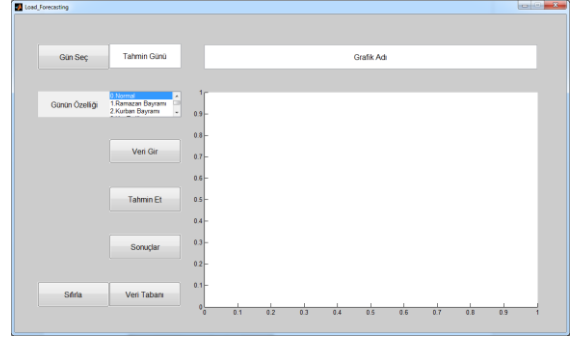
Öğrenme, çeşitli olaylar sonucu meydana gelen değişimler olarak tanımlanabilir. Yapay sinir ağlarında öğrenme ise ağırlıkların uygun girişlere atanması işlemi ile gerçekleşmektedir. Girdi ve çıktı değerlerine göre, başlangıçta rastgele atanan ağırlıklar süreç devam ettikçe değişmektedir. Bu değişimler ile uygun ağırlıklar bulunarak ağın öğrenmesi sağlanmaktadır[11].

III. YAPILAN ÇALIŞMA

Bu çalışmada İstanbul Avrupa yakası için Matlab programı ile kısa dönemli yük tahmini yapan bir yazılım hazırlanmıştır. Toplam yük tahmini yapılacağından dolayı öncelikle toplam yük verilerine göre YSA, ardından ise her bölge için ayrı ayrı YSA oluşturularak toplam yük tahmin edilmeye çalışılmıştır.

Program Arayüzünün Oluşturulması

Bu bölümde kullanımının kolay olmasını sağlamak amacıyla Matlab GUI kullanılarak Şekil 5'teki arayüz hazırlanmıştır.



Şekil 5: Hazırlanan Arayüz

Veri Setlerinin Hazırlanması

Veri setinde 2010-2013 yıllarına ait saatlik yük, saatlik sıcaklık ve günlük nem verileri kullanılmıştır. Veri tabanı Microsoft Excel'de oluşturulmuştur. Veri setinde 2010-2013 yılları arası saatlik veriler bulunduğu için toplamda veri tabanında 35064 satırlık örnek veri bulunmaktadır. Sütunlarda yıl, ay, gün, saat, saatlik sıcaklık, günlük nem, haftanın günü ve yük verileri bulunduğu için veri tabanı matrisinin sütun sayısı 8'dir. Bu veri tabanı toplam yüke dayalı tahminlerde kullanılan YSA'yı eğitmek amacıyla oluşturulmuştur.

Bölgeye dayalı yük tahmininde ise Matlab'ta kod yazılarak her bölge için ayrı ayrı veri setinin oluşturulması sağlanmıştır. Bunun amacı ise 0 değerinde yük olan veri satırının veri setine alınmamasını sağlamaktır. Çünkü bazı bölgelerde kesinti olduğunda yükler 0 değerine ulaşmaktadır ve bu da YSA'nın eğitimini etkilemektedir.

Sonuç olarak 50 adet bölge bazında oluşturulan veri seti ve toplam yük bazlı veri seti olmak üzere 51 adet veri seti hazırlanmıştır.

Normalizasyon

Ağ eğitiminin daha verimli hale getirilmesinde bazı ön işlemler uygulanmaktadır. Bu işlemler giriş değerlerini ağ için uygun hale getirip ağ kullanımı açısından daha iyi bir forma dönüştürmektedir. Normalizasyon adı altında uygulanan bu işlemler ham verilere

uygulanarak uygun veri kümesinin hazırlanmasında etkilidir. Eğer normalleştirme işlemi uygulanmadan eğitim süreci başlatılırsa eğitim uzun sürebilir ve ya ağ eğitilmeden program kapanabilir.

Özellikle çok katmanlı modellerde normalizasyon ağ performansı ile yakından etkilidir. Normalizasyon işlemi ile bütün girdilerin ve çıktılarının belirli bir aralıkta olması sağlanır. Normalizasyon işlemi için çeşitli formüller oluşturulabilmektedir. Uygulamada ise Matlab'ta bulunan "mapminmax" fonksiyonu kullanılarak girişlere normalizasyon işlemi uygulanmıştır. Normalizasyon işlemi uygulanan veriler -1 ile 1 arasında değerler almıştır. Çıktılara ise 3'te görülen denklem ve Matlab'ta bulunan "minmax" fonksiyonu uygulanarak değerlerin 0,1 ile 0,9 arasında olması sağlanmıştır.

$$P_n = 0,8 \cdot \frac{P - P_{\min}}{P_{\max} - P_{\min}} + 0,1 \quad (3)$$

Bu denklemde bulunan; P_n normalleştirilmiş veriyi, P_{\min} , P matrisinde bulunan minimum değerli vektörü, P_{\max} , P matrisinde bulunan maksimum değerli vektörü ve P giriş matrisini ifade etmektedir.

Ağın uygulamasında ise Şekil 7'de görüldüğü üzere girişler normalize edilerek ağa verilmiş çıktılar ise denormalize edilerek alınmıştır. Denormalize işleminin amacı ağdan alınan çıktılarının gerçek değerine dönüşmesini sağlamaktır. Denormalize işleminde ise oluşturduğumuz fonksiyonların tersi kullanılmıştır.

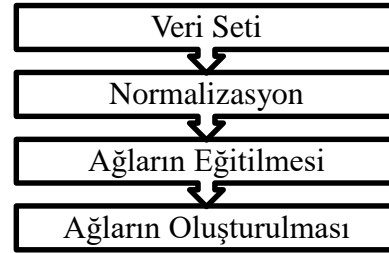
YSA'ların Oluşturulması

Veri kümesinde bulunan veriler ile ileri beslemeli çeşitli YSA'lar oluşturulmuş ve en iyi sonuç veren YSA bulunmaya çalışılmıştır. Bu çalışmada 51 ağ eğitildiği için her bir ağ için en iyi sonuç veren YSA'lar farklı yapıda olmaktadır.

İleri beslemeli ağda Levenberg-Marquardt optimizasyonu ile geliştirilmiş olan

"trainlm" eğitim fonksiyonu kullanılmıştır. Bu eğitim algoritması en hızlı eğitim algoritmasıdır[12]. Transfer fonksiyonu olarak ise "logsig" fonksiyonu seçilmiştir.

Ağların eğitilmesinde, örnek veri seti öncelikle normalizasyon işleminden geçirilmiş, ardından eğitime tabii tutulmuştur. Şekil 6'da bu işlemin basit şeması görülmektedir.



Şekil 6: Ağların Eğitilmesi

IV. UYGULAMANIN SONUÇLARI

Uygulamada 2 farklı model oluşturularak tahmin sonuçları alınmıştır;

1.model; toplam yüke dayalı olarak eğitilen ağın kullanıldığı model.

2.model; bölgelere dayalı olarak eğitilen ağların kullanıldığı model.

Uygulamada sonuçları iyileştirmek amacıyla tahmin edilecek gün ile tahmin edilecek günün bir hafta önceki tahmin verilerinden yararlanılarak katsayılar matrisi oluşturulmuştur. Tahmin sonuçları bu katsayılar ile çarpılarak daha güvenilir sonuçlar alınmıştır. Oluşturduğumuz algoritmanın akış şeması Şekil 7'de verilmiştir.

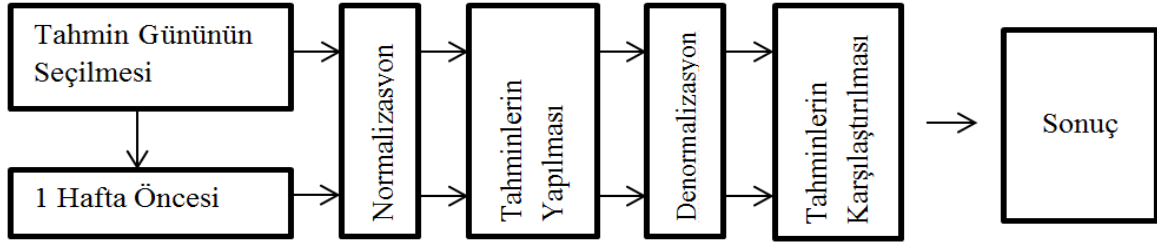
Haftanın farklı günleri için bir haftalık tahminler yapılmış ve sonuçları Şekil 8 ve Şekil 9'da gösterilmiştir. Bu eğrilerde görülen gerçek değerler ile tahmin değerleri arasındaki hata yüzdeleri 4'teki denklemle hesaplanmış ve Tablo 1'de görülen değerler karışımıza çıkmıştır.

$$MAPE = \frac{|P_{gerçek} - P_{tahmin}|}{P_{gerçek}} \cdot 100 \quad (4)$$

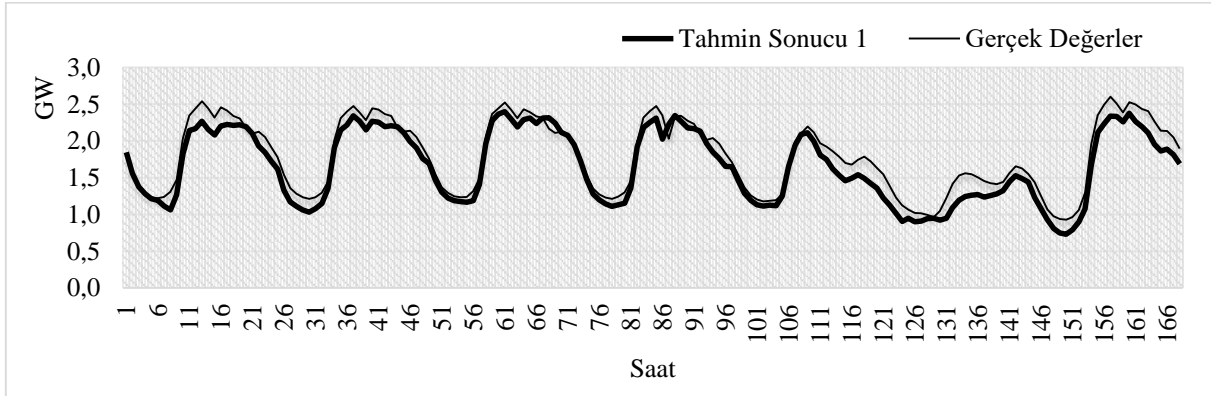
Bu denklemde bulunan; MAPE ortalama mutlak yüzdelik hatayı, $P_{gerçek}$ gerçek yük değerini ve P_{tahmin} tahmin sonucunu ifade etmektedir.

Tablo 1. Ortalama Yüzdelik Hatalar(MAPE)

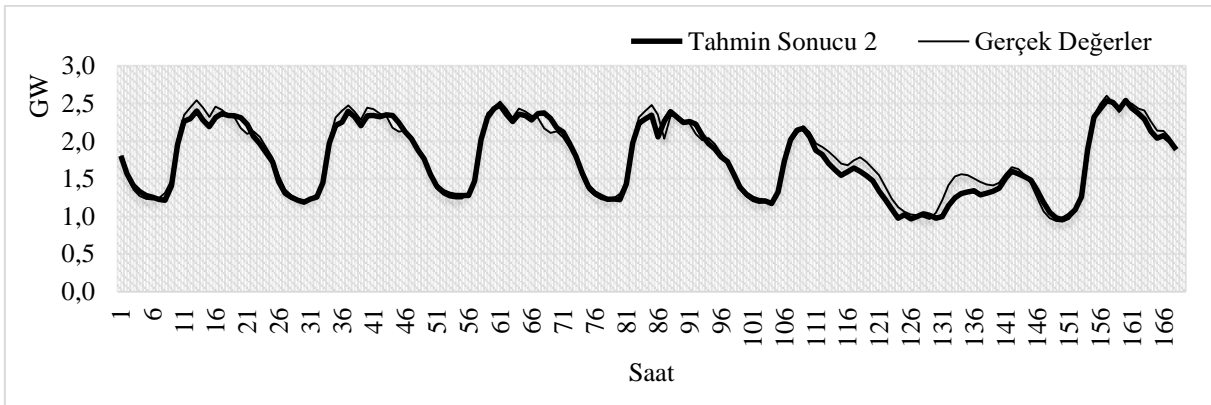
Gün	Model 1	Model 2
1	% 8,40	% 4,24
2	% 8,56	% 2,65
3	% 3,38	% 2,15
4	% 7,70	% 4,52
5	% 12,65	% 7,56
6	% 13,49	% 7,75
7	% 14,04	% 2,30
Ortalama	% 9,74	% 4,45



Şekil 7: Program Algoritmasının Genel Şeması



Şekil 8: Model 1 Sonuçlarının Gerçek Değerler ile Karşılaştırılması



Şekil 9: Model 2 Sonuçlarının Gerçek Değerler ile Karşılaştırılması

V. SONUÇ

Bu çalışmada yapay sinir ağları kullanılarak kısa dönemli elektrik talep tahmini yapılmıştır. Çalışmada ağı eğitimi için saatlik yük, saatlik sıcaklık ve günlük nem verileri kullanılmıştır. Ağ eğitimleri ise bölge bazında ve toplam yük verileri bazında olmak üzere farklı veri setleri hazırlanarak yapılmıştır.

Yapılan farklı uygulamalar için haftalık ortalama hata oranı % 4,45 ile bölgeye dayalı modelde daha düşük çıkmıştır. Bu çalışmanın bir sonucu olarak bölgeye dayalı talep tahminlerinin yapılması önerilmektedir.

KAYNAKLAR

- [1] Raza, M.Q., Baharudin, Z., Nallagownden, P., Islam, B., “A Comparative Analysis of PSO LM Based NN Short Term Load Forecast with Exogenous Variables for Smart Power Generation”, Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS, Tronoh, Perak, Malaysia, 2014.
- [2] Bilge, B., “Kısa Dönem Yük Tahmini”, TEİAŞ Milli Yük Tevzi İşletme Müdürlüğü, Gölbaşı, ANKARA.
- [3] D. C. Park, M. A. El-Sharkawi, R. J. Marks II, L. E. Atlas and M. J. Damborg, “Electric Load Forecasting Using An Artificial Neural Network”, IEEE Transactions on Power Systems, vol. 6, no. 2, pp. 442–449, May 1991.
- [4] T. S. Dillon, S. Sestito and S. Leung, “Short term load forecasting using an adaptive neural network”, International Journal of Electrical Power and Energy Systems, 13(4), 186-192, August 1991.
- [5] Alireza Khotanzad, Rey-Chue Hwang, Alireza Abaye and Dominic Maratukulam, “An Adaptive Modular Artificial Neural Network Hourly Load Forecaster and its Implementation at Electric Utilities”, IEEE Transactions on Power Systems, vol. 10, no. 3, pp.1716-1722, August 1995.
- [6] J. Vermaak and E .C. Botha, “Recurrent Neural Networks for Short-Term Load Forecasting”, IEEE Transactions on Power Systems, vol. 13, no. 1, pp.126-132, February 1998.
- [7] Shu Fan, Luonan Chen and Wei-Jen Lee, “Short-Term Load Forecasting Using Comprehensive Combination Based on Multimeteorological Information,” IEEE Transactions on Industry Applications, vol. 45, no. 4, pp.1460-1466, Jul./Aug. 2009.
- [8] Efe, Ö., Kaynak, O., Yapay Sinir Ağları ve Uygulamaları, Boğazici Üniversitesi, İstanbul, 2000.
- [9] Ghafari, A., Abdollahi, H., Khoshayand, M.R., Bozchalooi, I.S., Dadgar, A., Rafiee-Tehrani, M., “Performance comparison of neural network training algorithms in modeling of bimodal drug delivery”, *International Journal of Pharmaceutics (Elsevier)*, 327 126-138, 2006.
- [10] Aşkın, D., İskender, İ., Mamızadeh, A., “Farklı Yapay Sinir Ağları Yöntemleri Kullanılarak Kutu Tip Transformatör Sargısının Temal Analizi”, Elektrik-Elektronik Mühendisliği Bölümü, Mühendislik Fakültesi, Gazi Üniversitesi, Maltepe, Ankara, 2011.
- [11] Aslan, Y., Yaşar, C., Nalbant, A., “Electrical Peak Load Forecasting in Kütahya with Artificial Neural Networks”, Elektrik-Elektronik Mühendisliği Bölümü, Mühendislik Fakültesi, Dumlupınar Üniversitesi, Kütahya, Eylül 2006.
- [12] Beale, M.H., Hagan, M.T., Demuth, H.B., Neural Network Toolbox™: User's Guide, 2014.