

SOM DESTEKLİ RBF YAPAY SİNİR AĞLARI ile ANKARA İLİNİN DOĞALGAZ TÜKETİM TAHMİNİ

Oğuz KAYNAR¹, Metin ZONTUL², Ferhan DEMİRKOPORAN¹

¹Cumhuriyet Üniversitesi, İİBF Yönetim Bilişim Sistemleri Bölümü, 58140 Sivas

²İstanbul Aydın Üniversitesi,MMF Yazılım Mühendisliği Bölümü, İstanbul

Alınış: 17 Şubat 2009

Kabul Ediliş: 18 Haziran 2010

Özet: Doğal gaz tüketim tahmininde klasik zaman serilerinin yanı sıra son zamanlarda yapay sinir ağları başarıyla kullanılmaktadır. Bu çalışmada doğal gaz tüketimini tahmin amacıyla iyi bir fonksiyon yaklaşırıcı olarak bilinen Radyal Tabanlı Sinir Ağları kullanılmıştır. Veri setinin büyük olması durumunda RBF ağlarının eğitiminde gizli katmanda kullanılacak nöron sayısının ne olacağı ve bu katmanda yer alan nöronlara ait radyal fonksiyonların merkezlerinin nasıl belirleneceği önemli problemlerden birisini oluşturmaktadır. Bu amaçla RBF ağlarının eğitiminde SOM yapay sinir ağları ile K-means kümeleme algoritmalarından yararlanılmıştır. Her iki yöntemle de ayrı ayrı eğitilen RBF ağları yardımıyla tahminler yapılarak sonuçlar karşılaştırılmıştır.

Anahtar Kelimeler: Yapay Sinir Ağları, RBF, SOM, K-means

Natural Gas Consumption Forecasting for Ankara City by SOM Supported Rbf Artificial Neural Network

Abstract: Recently, Artificial Neural Network has been widely used in natural gas consumption forecasting in addition to classical time series analysis. In this study, RBF (Radial Basis Function) neural network which is a well known function approximation is used to predict natural gas consumption. In the case of having large dataset, how many neurons will be used in the hidden layer and how the centers that are belong to neurons located in this layer are determined are important issues. To overcome these problems, SOM artificial neural network and K-means clustering algorithm have been utilized in training RBF. The predictions are made by means of RBF that is trained with both methods and results are compared.

Keywords: Artificial Neural Network, RBF, SOM, K-means

1. GİRİŞ

Doğalgaz temini noktasında tamamen dışa bağımlı bir politika izleyen ve doğalgaz stok kapasitesi toplam tüketiminin ancak %5'ini karşılayan ülkemiz için tüketilen doğalgaz miktarının tahmini, son derece önemlidir. Talebin doğru bir şekilde tahmini, sektöre yapılacak yatırımları ve gaz alımı ile ilgili anlaşmaları, dolayısıyla sektörün gelişimini etkileyecek unsurlardan birini oluşturmaktadır [1]. Doğal gaz tüketim tahmininde klasik zaman serilerinin yanı sıra son zamanlarda yapay sinir ağları başarıyla kullanılmaktadır. Geçmiş dönemlere ilişkin gözlem değerleri yardımıyla geleceğe yönelik tahminler yapmayı amaçlayan zaman serisi modelleri tıp, mühendislik, işletme, ekonomi ve finans gibi birçok alanda yaygın bir şekilde kullanılmaktadır. Zaman serileri yardımıyla tahmin yapmak için değişik yöntemler kullanılarak oluşturulan farklı modeller bulunmaktadır. Bu modeller arasında en çok bilinen ve yaygın olarak kullanılan ARIMA modelleridir. Seriyi oluşturan veriler arasında doğrusal bir ilişkinin olduğunu varsayan ve bu doğrusal ilişkiyi modelleyebilen ARIMA modelleri durağan ya da çeşitli istatistiksel yöntemlerle durağan hale getirilen zaman serilerine başarıyla uygulanabilmektedir. Oysa uygulamada karşılaşılan birçok zaman serisi sadece doğrusal ilişki içermemektedir. Yapısı gereği hem doğrusal hem de doğrusal olmayan ilişkileri modelleyebilen yapay sinir ağları (YSA) son yıllarda zaman serilerinin analizinde kullanılan alternatif yöntemlerden biri olmuştur. Yapay sinir ağlarının en önemli avantajı, veri setine ilişkin fonksiyonel yapının tam olarak belirlenmediği durumlarda, veriden hareketle birçok değişik formdaki fonksiyonel yapıyı başarıyla modelleyebilmesidir. Genel fonksiyon yaklaşırıcısı olarak da bilinen yapay sinir ağları, istatistiksel yöntemlerin

aksine veri seti üzerinde her hangi bir ön varsayıma gerek duymaz. Yapay sinir ağlarının zaman serilerinin tahmininde kullanıldığı çalışmalara ilişkin geniş bir derleme Zhang vd. tarafından yapılmıştır [2].

Doğalgaz tüketiminin yapay sinir ağları ile tahminine ilişkin literatürde birçok çalışma yer almaktadır. Kaynar ve diğerleri [1] yaptıkları çalışmada Ankara iline ilişkin günlük ve haftalık doğalgaz tüketim serilerinden faydalanarak ARIMA ve MLP yapay sinir ağları ile tahminde bulunmuşlar ve MLP yapay sinir ağlarının ARIMA modellerinden daha iyi sonuç verdiğini belirtmişlerdir. Ivezic [3] çalışmasında geçmiş günlere ait sıcaklık ve doğal gaz tüketim verilerini kullanarak Sibiryaya için günlük ve haftalık doğalgaz tüketim miktarlarını tahmin etmiştir. Garcia ve Mohagheg [4] ise çalışmalarında 2020 yılına kadar Amerika Birleşik Devletleri'ndeki doğal gaz üretim miktarını tahmin eden YSA modeli geliştirmiştir. Bolen [5] yaptığı tez çalışmasında ABD'nin sıvılaştırılmış gaz ithalini tahmin etmek için yapay sinir ağlarından faydalanmıştır. Khotanzad [6] ise çalışmasında iki aşamalı yapı önermiştir. Birinci aşamada iki adet paralel çalışan adaptif yapay sinir ağı kullanarak ayrı ayrı elde ettiği günlük doğal gaz tüketim tahminlerini ikinci aşamada birleştirme ünitesinden geçirerek son tahmin değerini elde etmiştir. Brown ve diğerleri [7] ise çalışmalarında YSA kullanarak geliştirdikleri günlük doğal gaz tüketim modelini lineer regresyon modelleriyle karşılaştırmışlar ve YSA kullanılarak elde edilen tahmin değerlerinin daha iyi sonuçlar verdiğini göstermişlerdir. Viet ve Mandziuk [8] Polonya'nın belirli bir bölgesinde doğal gaz tüketiminin tahmini için gerçekleştirdikleri çalışmada YSA ve Bulanık YSA modellerini kullanmışlar ve uzun dönemli aylık tahminlerin orta dönemli haftalık tahminlere oranla daha iyi sonuçlar verdiğini göstermişlerdir. Ayrıca çalışmalarında, YSA kullanılarak gerçekleştirilen tahmin sonuçlarının klasik istatistiksel yöntemlerle elde edilen tahmin sonuçlarından daha iyi olduğunu belirtmişlerdir.

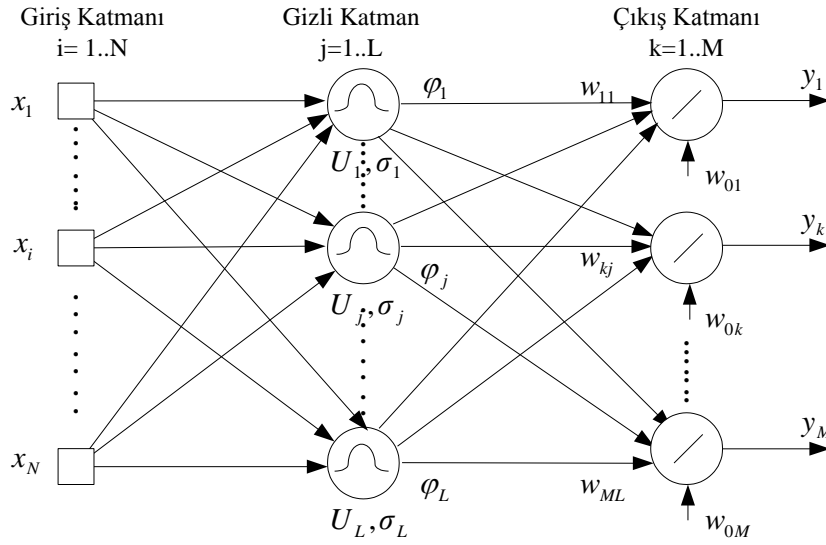
Bu çalışmada ise radyal tabanlı yapay sinir ağları kullanılarak Ankara iline ilişkin doğal gaz tüketim tahmini çalışması yapılmıştır. RBF ağlarının gizli katmanında yer alan radyal fonksiyonlara ait merkez vektörlerin belirlenmesi için kendinden düzenlemeli haritalar (Self Organization Map-SOM) ve K-means kümeleme yöntemlerinden faydalanılmıştır. Her iki yöntemle de ayrı ayrı eğitilen RBF modellerinden elde edilen tahmin sonuçları karşılaştırılmıştır.

2. YAPAY SİNİR AĞLARI

Yapay sinir ağları, biyolojik sinir ağlarından esinlenilerek ortaya çıkarılan ve biyolojik sinir ağlarına benzer bazı performans özellikleri içeren bir bilgi işleme sistemidir [9]. Basit bir şekilde insan beyninin çalışma şekli taklit eden YSA'lar veriden öğrenebilme, genelleme yapabilme, sınırsız sayıda değişkenle çalışabilme vb. birçok önemli özelliğe sahiptir. YSA'nın çalışmasına esas teşkil eden en küçük birimler yapay sinir hücresi ya da işlem elemanı olarak isimlendirilir. Yapay sinir ağı, Şekil 1'de gösterilen bir dizi sinir hücresinin ileri sürümlü ve geri beslemeli bağlantı şekilleri ile birbirine bağlanmasından oluşur. Günümüzde, belirli amaçlarla ve değişik alanlarda kullanılmaya uygun birçok yapay sinir ağı modeli (MLP, RBF, LVQ, Hopfield, Recurrent, SOM, ART vb.) geliştirilmiştir. Bu ağ yapıları içerisinde zaman serisi analizinde en yaygın kullanım alanı bulanlar çok katmanlı ileri beslemeli yapay sinir ağları (Multiple Layer Perceptron-MLP) ile çalışmamızda da kullanılan radyal tabanlı fonksiyon (Radial Basis Function) ağlarıdır.

2.1 Radyal Tabanlı Fonksiyon Ağları

RBF ağları bir giriş katmanı, aktivasyon fonksiyonu olarak ağı ismini veren radyal fonksiyonların kullanıldığı tek bir gizli katman ve çıkış katmanı olmak üzere 3 katmanlı mimariden oluşmaktadır (Şekil 1). Bir RBF ağının çalışma ilkesi, giriş verisine bağlı olarak gizli katmanda uygun genişlik ve merkez değerlerine sahip RBF'leri belirleyerek, çıkış katmanında bu fonksiyonların ürettiği çıktıların uygun ağırlık değerleriyle doğrusal birleşimlerini oluşturup giriş-çıkış arasındaki ilişkiyi belirleme süreci olarak açıklanabilir. İlk olarak çok değişkenli interpolasyon problemlerinin çözümünde Powel [10] tarafından kullanılan radyal tabanlı fonksiyonların yapay sinir ağlarında kullanımı Broomhead and Lowe tarafından gerçekleştirilmiştir [11]. MLP'ye oranla daha kısa zamanda eğitilmesi [12] ve lokal minimumlara takılmaksızın en iyi çözüme yakınsamaları [13] gibi özelliklerinden dolayı RBF ağları tahmin, eğri uydurma ve fonksiyon yaklaşırma gibi problemlerin çözümünü içeren uygulamalarda MLP'ye alternatif bir sinir ağı olarak kullanılmaya başlanmıştır [14], [15].



Şekil 1. RBF Yapay Sinir Ağı

Giriş katmanındaki veriler MLP'den farklı olarak ağırlık değerleri ile çarpılmadan direkt olarak gizli katmana iletilirler. Gizli katmanda yer alan Radyal fonksiyonların merkezlerini gösteren U_j referans vektörleri ile giriş vektörü arasındaki uzaklık değerine bağlı olarak gizli katman çıkışında eşitlik 1'de gösterildiği şekliyle bir değer üretilir.

$$\varphi_j = e^{-\left[\frac{\|X_i - U_j\|^2}{\sigma_j^2} \right]} \quad (1)$$

Eşitlikte gösterilen X_i değeri o an ağa sunulan girdi vektörünü, U_j referans vektörünü, $\|\cdot\|$ uzaklık fonksiyonunu, σ_j ise Gaussian fonksiyonun genişlik değerini göstermektedir.

Birçok uzaklık ölçüsü tanımlanmasına rağmen uzaklık ölçüsü olarak genelde Öklid uzaklığı kullanılır. Giriş vektörü, referans vektörüne ne kadar çok benziyorsa fonksiyonun üreteceği değer o oranda büyük olacaktır. Gizli katmanda kullanılmak üzere çok sayıda radyal tabanlı fonksiyon (Gaussian, Multi-Quadric, Generalized Multi-Quadric, Thin Plate Spline) önerilmesine karşın uygulamalarda en çok Gaussian fonksiyon tercih edilmiştir. Chen ve diğerleri ağda kullanılacak Radyal fonksiyon seçiminin ağın performansı üzerinde önemli bir etkisinin olmadığını belirtmişlerdir [15]. Gizli katman çıkışında elde edilen φ_j değerleri daha sonra w_{kj} ağırlık değerleri ile çarpılıp toplanarak eşitlik 2'de verildiği gibi ağın çıkışı hesaplanır.

$$y_k = \sum_{j=1}^L \varphi_j w_{kj} + w_{k0} \quad (2)$$

RBF ağlarının eğitimi, gizli katmanda yer alan radyal fonksiyona ait merkez vektörü (U_j) ile genişlik değerinin (σ_j) ve gizli katman ile çıkış katmanı arasında yer alan ağırlık (w_{kj}) parametrelerinin belirlenmesi sürecini kapsar. Referans vektörü ve referans vektörünün genişliğini belirlemek üzere literatürde birçok yöntem önerilmiştir. Bunlardan bazıları aşağıda sıralanmıştır:

Merkez vektörlerini veri seti içerisinde seçmek: Bu yöntemde gizli katmanda kullanılacak veri seti sayısı, ağa sunulan örnek sayısına eşitlenerek her bir örnek veri seti aynı zamanda merkez vektör olarak kullanılır. Kısaca giriş uzayındaki her bir nokta için bir radyal fonksiyon belirlenir. Bu durum gerçek RBF (Exact RBF) olarak adlandırılır. Bu durumun iki sakıncası vardır. Bunlardan ilki, veri setinin büyük olması durumunda hesap karmaşası yaratacak boyut problemidir. Diğeri ise ağdaki verilerin gürültülü olması durumunda ağ bu gürültülü değerleri de öğreneceğinden eğitim verisi için sıfır hata ile öğrenme gerçekleşirken test verisi için sistemin performansı aynı oranda iyi olmayacaktır. Bu nedenle hesap karmaşasını azaltmak ve ağı aşırı öğrenmesinden kaynaklanan ezberleme sorununu gidermek için gizli katmandaki nöron sayısı giriş veri setindeki örnek sayısından daha küçük seçilir. Bu durumda merkez vektörleri giriş vektörlerinden rastgele seçilerek belirlenir.

Merkez vektörlerinin danışmasız öğrenme ile belirlenmesi: Bu yöntemde giriş vektörleri, bilinen K-ortalamar ve kendinden düzenlemeli ağlar SOM (Self Organization Map) yöntemlerinden biri kullanılarak belirli sayıda kümeler ayrılır ve her bir kümenin merkezi radyal fonksiyonların merkezi olarak atanır. Tüm radyal fonksiyonlar için σ_j çok küçük ya da çok büyük seçilmemek kaydıyla kullanıcı tarafından belirlenebildiği gibi eşitlik 3'teki şekilde de hesaplanabilir [16]. D_{\max} kümeler arasındaki maksimum uzaklığı, M ise küme sayısını göstermektedir.

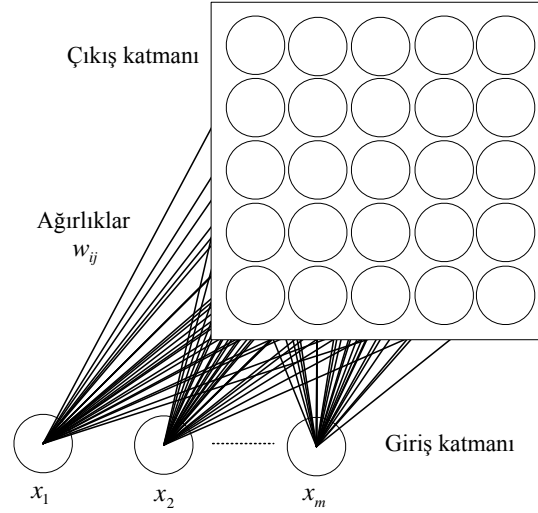
$$\sigma = d_{\max} / \sqrt{2M} \quad (3)$$

Merkez vektörleri ve σ değeri belirlendikten sonraki aşama gizli katmanla çıkış katmanı arasında yer alan w_{kj} değerlerinin belirlenmesidir. Gizli katman çıkışları (ϕ_j) belirlendikten sonra w_{kj} değerlerinin hesaplanması MLP ağlarında olduğu gibi geri yayılım algoritması yardımıyla iteratif olarak ya da en küçük hata kareleri yöntemiyle eşitlik 4'te verilen denklemin çözümü sonucunda tek aşamada elde edilir.

$$\phi W^T = T \quad (4)$$

2.2 Kendinden Düzenlemeli Yapay Sinir Ağları (Self Organization Map-SOM)

İlk olarak Teuvo Kohonen [17] tarafından ortaya atılan SOM ağları, kümeleme amacıyla kullanılmasının yanı sıra giriş verileri arasındaki topolojik ilişkiyi de koruyarak giriş uzayında yer alan çok boyutlu verilerin çıkış uzayında tek ya da iki boyutlu haritalarla görselleştirilmesini de sağlayan yapay sinir ağı modelidir. Son yıllarda sınıflama, kümeleme, boyut indirgeme ve görselleştirme aracı olarak popülerliğini artıran SOM ağlarının kullanımına ilişkin çalışmalar Kaski vd. ile Oja vd. tarafından derlenmiştir [18], [19]. Şekil 2'de görüleceği üzere SOM ağları Kohonen katmanı olarak da adlandırılan bir çıkış katmanı ile verilerin ağa sunulduğu bir giriş katmanından oluşmaktadır. Çıkış katmanında yer alan her bir nöron girişteki nöronlara referans vektörleri (code-book) olarak adlandırılan ağırlık vektörleriyle bağlıdır. Bu vektörler ağa sunulan giriş vektörüyle aynı boyuta sahiptir. Ağı giriş katmanındaki nöron sayısı giriş vektörü tarafından belirlenirken, çıkışında yer alan nöronların sayısı ise isteğe bağlı olarak belirlenir. Çıkış katmanındaki bu nöronların farklı şekiller ve boyutlardaki dizilimlerinden farklı topolojilere sahip haritalar oluşturulur. Doğrusal, dikdörtgen, altıgen ve kübik yapıda olabilen bu dizilimlerden en çok kullanılanı dikdörtgen yapıdaki haritalardır. Nöronların farklı dizilimleri sadece topolojik komşuluğu etkiler ve bu nöronlar arasında giriş nöronlarına olan bağlantı gibi herhangi bir fiziksel bağlantı yoktur. Topolojik komşuluk sadece ağı eğitimi sırasında yarışmayı kazanan referans vektörle birlikte komşu vektörlerin de ağırlıklarının değiştirilmesinde etkilidir.



Şekil 2. SOM ağı

Som ağlarının eğitiminde danışmasız öğrenme algoritması kullanılır. Danışmasız öğrenme algoritmalarında ağırlık girişle çıkış arasındaki ilişkiyi öğrenebilmesi için danışmalı öğrenme algoritmalarında olduğu gibi önceden ağırlık sunulan giriş-çıkış vektör çiftlerine ihtiyaç duyulmaz. Ağın eğitimi için sadece giriş vektörleri yeterlidir. Ağın öğrenmesi, çıkış nöronlarına ait referans vektörlerinin giriş vektörüne olan uzaklıklarına bağlı olarak yarışması esasına dayanır. Yarışmayı kazanan nöronla birlikte komşu nöronların da ağırlık değerleri değiştirilerek, ağırlık vektörlerinin girişteki verilerin oluşturduğu küme merkezlerine doğru hareketi sağlanır ve eğitim sonunda kümeleri temsil edecek referans vektörleri elde edilir. Sadece kazanan nöronun değil bu nörona komşu diğer nöronların da kazananla birlikte ağırlıklarının değiştirilmesi neticesinde birbirine benzer desenlerin uzaysal kümelenmesi sağlanmaktadır. Böylece n boyutlu bir giriş uzayında birbirine benzer kümeler, 2 boyutlu çıkış uzayında birbirlerine komşu olmaktadır.

SOM ağlarının eğitim süreci yarışmacı, işbirlikçi ve adaptasyon süreçlerinden oluşur. Yarışmacı süreçte ilk olarak l tane çıkış nöronuna ait ağırlık vektörleri (w_j) rastgele değerler atanarak oluşturulur (eşitlik 5).

$$w_j = [w_{j1} \quad w_{j2} \quad \dots \quad w_{jm}] \quad j = 1, 2, \dots, l \quad (5)$$

Ardından ağa giriş (x) vektörlerinden birisi rastgele seçilerek sunulur (eşitlik 6).

$$x = [x_1 \quad x_2 \quad \dots \quad x_m] \quad (6)$$

Sonrasında her bir nörona ait referans vektörü ile giriş vektörü arasındaki uzaklık değerleri (d_j) hesaplanarak karşılaştırılır (eşitlik 7). En küçük uzaklığa sahip nöron yarışmayı kazanan nörondur ve bu nöron en iyi uyan birim (Best matching Unit, BMU) olarak da adlandırılır [16].

$$d_j = \|X - W_j\| = \sqrt{\sum_{i=1}^N (x_i - w_{ij})^2} \quad (7)$$

İşbirlikçi süreçte sadece kazanan nöronun ağırlıkları değil aynı zamanda kazanan nörona topolojik olarak komşu olan nöronların da ağırlıkları güncellenir. Giriş vektörüne yaklaşacak şekilde ağırlıkları değiştirilen bu nöronların bir sonraki adımda kazanma olasılıkları daha da artırılmış olur. Yeni ağırlık değerleri eşitlik 8 yardımıyla hesaplanır.

$$W_j(t+1) = W_j(t) + \eta(t)h_{j,i}(t)(X - W_j(t)) \quad (8)$$

Eşitlikte $W_j(t+1)$ bir sonraki adımdaki ağırlık vektörünü, $W_j(t)$ şu anki ağırlık vektörünü, X giriş vektörünü, $\eta(t)$ zamanla değişen öğrenme oranını, $h_{j,i}(t)$ hem zamana hem de kazanan nöronla komşu nöronlar arasındaki uzaklığa bağlı olarak değişen topolojik komşuluk fonksiyonunu göstermektedir. Ağırlığı değiştirilecek birimlerin kimler olacağı topolojik komşulukla ilgilidir. Topolojik fonksiyon sadece topolojik komşuların kimler olacağını belirlemekle kalmaz aynı zamanda bu komşu nöronların kazanan birime olan uzaklıklarına bağlı olarak öğrenme oranı üzerindeki etkisini de belirler. Topolojik komşuluk belirlenirken komşuluk yarıçapından faydalanılır. Kazanan nörona olan uzaklıkları komşuluk yarıçapı içerisinde kalan nöronlar topolojik komşuluk içerisindeyler. Komşuluk yarıçapını gösteren eşitlik 9'daki fonksiyon zamanla azalan bir yapıya sahiptir. Eşitlikte σ_0 başlangıç yarıçapını, λ_1 zaman sabitini, t ise o anki çevrimi göstermektedir.

$$\sigma(t) = \sigma_0 \exp\left(-\frac{t}{\lambda_1}\right) \quad t = 0,1,2,\dots \quad (9)$$

Eğitim sürecinde kazanan nöron etrafındaki nöronların sayısı zamanla azaltılarak eğitim sonuna doğru yalnızca kazanan nöron kalacak şekilde ağırlıkların değiştirilmesi sağlanır. Böylelikle eğitimin başında çıkış uzayının geniş bir bölümünün etkileşime uğrayarak daha büyük kümelerin oluşması sağlanırken eğitim sonuna doğru çok daha küçük boyutlu ayrışmaların oluşması gerçekleştirilir [20].

3. K-MEANS KÜMELEME YÖNTEMİ

En bilinen kümeleme algoritmalarından olan k-means, 1967 yılında J.B. MacQueen [21] tarafından geliştirilmiştir. K-means yöntemi n adet nesneden oluşan veri setinin; $k \leq n$ olacak şekilde, giriş parametresi olarak verilen, küme benzerliği yüksek, kümeler arası benzerliği düşük olan k adet kümeye bölünmesi mantığına dayanır. Küme benzerliği, bir kümedeki nesnelerin ortalaması olan küme merkezi ile diğer nesnelere arasındaki uzaklığa göre hesaplanır [22]. K-means algoritmasında ilk olarak her biri bir küme merkezini ifade edecek şekilde rastgele k adet vektör seçilir. 2. adımda kümelenecek veriler en yakın küme merkezine atanır. 3. adımda mevcut küme üyelikleri kullanılarak küme merkezleri yeniden hesaplanır. Son adımda ise eğer yakınsama kriteri sağlanmadıysa 2. adıma geri dönlür. Tipik yakınsama kriterleri hiçbir nesnenin bir kümeden diğerine yer değiştirmemesi veya hata kriterinde önemli bir azalmanın olmamasıdır [23].

Her biri d boyutlu olmak üzere x_1, x_2, \dots, x_N şeklinde N adet vektör verilsin ve bu vektörler C_1, C_2, \dots, C_K

biçiminde K kümeye ayrılınsın. O zaman $\sum_{i=1}^k n_i = N$ olmak üzere her bir kümedeki eleman sayısı n_i ile

gösterilirse herhangi bir C_i kümesinin merkez vektörü μ_i , eşitlik 10'da verildiği şekilde hesaplanır

$$\mu_i = \frac{1}{n_i} \sum_{x_j \in C_i} x_j \quad (10)$$

Burada x_j değeri C_i kümesine ait olan j . vektördür. K-means kümeleme yönteminin değerlendirilmesinde en yaygın olarak karesel hata kriteri SSE kullanılır. En düşük hata kareleri toplamı (SSE) değerine sahip kümeleme en iyi sonucu verir [24]. C_i kümesi için hata-kare her bir C_i örneği ile onun merkezi arasındaki Öklid uzaklıkları toplamıdır. Bu hataya "küme içi değişme" adı da verilir. K adet küme için toplam hata kareleri değeri eşitlik 11 yardımıyla hesaplanır.

$$SSE = \sum_{i=1}^k \sum_{x_j \in C_i} \|x_j - \mu_i\|^2 \quad (11)$$

Burada algoritmaya kullanıcı tarafından verilen k parametresi ile N tane veriden oluşan veri seti k adet kümeye bölünerek karesel-hata fonksiyonunu azaltacak k parça belirlenmeye çalışılır [23]. Bu yüzden kare-hatanın bir önceki çevrime göre azalması beklenir.

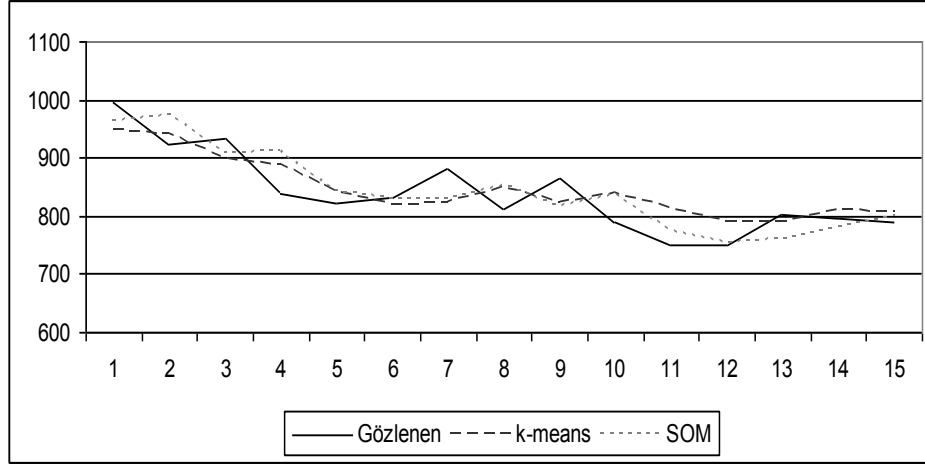
K-means yöntemi büyük veri setlerini işlemede oldukça verimli ve ölçeklendirilebilir yapıdadır. Kümeler birbirinden ayrılmış yoğun bulutlar şeklinde ise iyi sonuç verir [22]. K-means yönteminde k küme sayısının ve başlangıç bölümlemesinin belirlenmesi için herhangi etkin ve evrensel bir metodun olmaması bir dezavantaj olarak sayılabilir. Bu problem için en genel strateji rastgele başlangıç bölümlemesi ile algoritmayı birçok kez çalıştırmaktır [25]. K-means algoritması gürültüye karşı oldukça duyarlıdır ve aşırı uç değerlerden etkilenir. Az sayıda bile olsalar bu uç değerler küme ortalama değerlerini büyük ölçüde etkiler [22].

4. UYGULAMA VE SONUÇLAR

Çalışmada, Ankara iline ilişkin Ocak 2005 ile Haziran 2006 arasındaki günlük doğalgaz tüketim değerlerinden elde edilen zaman serisi kullanılmış ve uygulamalar Matlab ortamında gerçekleştirilmiştir.

RBF ağlarının eğitiminde Gizli katmanda yer alan nöronları ve bu nöronlara ilişkin merkez vektörlerini belirlemek üzere SOM ağları ve K-means kümeleme yöntemlerinden yararlanılmıştır. SOM ağlarında çıkış katmanında 5 farklı topoloji kullanılarak giriş verileri kümelenecek ve kümeleme sonuçlarından elde edilen veriler kullanılarak farklı sayıda gizli katman nöronuna sahip RBF modelleri elde edilmiştir. Benzer şekilde giriş verileri k-means algoritması yardımıyla küme sayısı 1 den 50'ye kadar değiştirilerek farklı kümeler elde edilmiş, bu kümelerden elde edilen veriler RBF modellerinin gizli katmanlarının belirlenmesinde kullanılmıştır. Tek dönemlik tahmin yapılacağı için tüm modeller için çıkış katmanındaki nöron sayısı 1 olarak belirlenmiştir. Zaman serisi tahmin etmek amacıyla kullanılan RBF ağlarında giriş katmanındaki nöron sayısını belirlemek çıkış katmanındaki nöron sayısını belirlemek kadar kolay değildir. Çünkü serinin t zamandaki değerinin geçmiş kaç gözlem değerinden etkilendiğinin belirlenmesi kritik bir sorudur ve bu sorunun cevabı girdi işlem elemanı sayısının kaç olacağını göstermektedir. Girdi işlem elemanı sayısının belirlenmesine ilişkin çeşitli görüşler olmasına karşın genel bir kural bulunmamaktadır ve girdi işlem elemanı sayısı deneme yanılma yöntemiyle belirlenmektedir. Her iki model için de giriş nöron sayıları 1-10 arasında değiştirilerek farklı giriş değerlerine sahip RBF yapıları oluşturulmuştur. Daha sonra eğitim verilerine göre en az hata değerini veren RBF modelleri en iyi model olarak belirlenmiştir. SOM ağları yardımıyla küme merkezleri belirlenen RBF ağları içerisinde küçük hata değerine sahip model doğrusal dizilime sahip, giriş nöron sayısı 6 ve gizli katman nöron sayısı 38 olan modeldir. K-means yardımıyla küme merkezleri belirlenen ağ modelleri içerisinde ise en küçük hata değerine sahip model ise 2 giriş nöronu, 11 gizli nöron içermektedir.

Kullanılan yöntemlerin her ikisi için de en küçük hata değerini veren modeller belirlendikten sonra test verileri kullanılarak her iki model için tahmin değerleri elde edilmiştir. Tablo 1 gerçek değerler ile modellerden elde edilen tahmin değerlerini göstermektedir. Bu değerlere ilişkin grafik de Şekil 3'te gösterilmiştir. Hangi eğitim yönteminin daha iyi performans sergilediğini belirlemek amacıyla MAPE performans kriteri kullanılmıştır. MAPE kriterine göre SOM ağları ile eğitilen RBF modelinin k-means kullanılarak eğitilen RBF modellerine göre daha iyi sonuç verdiği görülmüştür.



Şekil 3. Günlük doğalgaz tüketimi için gözlenen ve tahmin edilen değerlerin grafiği

Tablo1: Günlük doğalgaz tüketimi için gözlenen ve tahmin edilen değerler

GÖZLENEN	K-MEANS	SOM
997	947,9410	962,4948
925	941,0218	971,9239
934	897,9444	908,6014
837	887,6623	910,6936
823	841,9879	841,3320
833	820,4238	829,6549
880	823,6502	830,0659
811	847,4334	850,2236
865	821,2126	816,5574
791	838,5223	834,2291
750	811,5310	771,9517
750	789,9952	753,0933
803	788,7634	761,0143
796	808,8575	780,1118
788	806,8033	799,2404
MAPE	4,1107	3,7502

REFERANSLAR

- [1] Kaynar O, Taştan S, Demirkoparan F, “Yapay Sinir Ağları ile Doğal Gaz Tüketim Tahmini”,10. Ekonometri ve İstatistik Sempozyumu, ,Erzurum, 22-27 Mayıs 2009,106
- [2] Zhang, G., Patuwo, B. E., Hu, M. Y. ,“Forecasting with artificial neural networks: the state of the art”,

- International Journal of Forecasting, 14, 35-62, 1998
- [3] Ivezic, D., "Short-Term Natural Gas Consumption Forecast", FME Transactions VOL. 34, No 3, 165-169.,2006
- [4] Garcia, A., Mohagheg, S. D."Forecasting US Natural Gas Production into year 2020: a comparative study", Eastern Society of Petroleum Engineers Regional Conference , 2004, SPE 91413.
- [5] Bolen, M. S. "A New Methodology for Analyzing and Predicting U.S. Liquefied Natural Gas Imports Using Neural Networks", Texas A&M University, Yüksek Lisans Tezi.,2005
- [6] Khotanzad, A., Elragal, H., Lu, Tsun-Liang "Combination of Artificial Neural-Network Forecasters for Prediction of Natural Gas Consumption", IEEE Trans. Neural Netw., 11, 464-473.,2000
- [7] Brown, R. H., Matin, L., Kharout, P., etc all. "Development of Artificial Neural-network Models to Predict Daily Gas Consumption", A.G.A. Forecasting Rev., 5, 1-22.,1996
- [8] Viet, N.H. ve Mandzuik, J. ,"Neural and Fuzzy Neural Networks for Natural Gas Consumption Prediction", IEEE XIII Workshop on Neural Networks for Signal Processing, 759-768., 2003
- [9] Fausett, L., "Fundamentals of Neural Networks: Architectures, Algorithms and Applications", Prentice Hall.,3,1994
- [10] Powell MJD., "Radial basis functions for multivariable interpolation: a review. In Algorithms for Approximation", Mason JC, Cox MG (eds.) Carendon Press: Oxford; s.143-167.,1987
- [11] Broomhead DS, Lowe D., "Multivariable functional interpolation and adaptive Networks", Complex Systems 2: s.321-355.,1998
- [12] Moody, J. & Darken, C. J., "Fast learning in networks of locally-tunes processing units." ,Neural Computation 1(2), 281-294,1989
- [13] Park, J. ve Sandberg,I.W., " Universal approximations using Radial-Basis-Function Network.", Neural Computation 3(2), 246-257,1991
- [14] Bianchini, M., Frasconi, P. , Gori, M., Learning without local minima in radial basis function networks. IEEE Trans. Neural Networks 6(3), 749-755 ,1995
- [15] Chen, S., Cowan, C. F. N., Grant, P. M., "Orthogonal least squares learning algorithm for radial basis function networks." , IEEE Trans. Neural Networks 2(2), 302-309,1991
- [16] Haykin, S., "Neural Networks - A Comprehensive Foundation",321,470,Prentice Hall.1999
- [17] KOHONEN,T., " Automatic formation of topological maps of patterns in a self-organizing system.", Proceedings of Second Scandinavian Conference on Image Analysis, Espoo, Finland, 214-220, 1981.
- [18] KASKI S., KANGAS J., T. KOHONEN. "Bibliography of self-organising maps (SOM) papers: 1981-1998.", Tech. rep., Neural Computing Surveys, 1998.
- [19] Oja, M., Kaski, S., Kohonen, T. ."Bibliography of Self-Organizing Map (SOM) Papers: 1998-2001 Addendum", *Neural Computing Surveys*, 3: 1-156, 2002
- [20] ALPDOĞAN, Y. ve BİLGE, H.Ş , "Kendinden Düzenlenen Haritalar ile Ders İçeriklerinin Sınıflandırılması", Gazi Üniv. Müh. Mim. Fak. Der., Cilt 24, No 2, 303-310,2009
- [21] MACQUEEN, J. B., "Some Methods for Classification and Analysis of Multivariate Observations", Proc. Symp. Math. Statist. and Probability (5th), 1967, 281-297.
- [22] HAN, J., and KAMBER, M., "Data Mining Concepts and Techniques", Morgan Kauffmann Publishers Inc., 2006.
- [23] JAİN, A. K., MURTY, M. N., and FLYNN, P. J., "Data Clustering: A Review", ACM Computing Surveys, Volume:31, No:3, 1999.
- [24] IŞIK, M., ÇAMURCU, A. Y., "K-means, K-medoids ve Bulanık C-means Algoritmalarının Uygulamalı Olarak Performanslarının Tespiti", İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, Yıl:6, Sayı:11, 31-45,2007
- [25] XU, R., and WUNSCH, II. D., "Survey of Clustering Algorithms", IEEE Transactions On Neural Networks, Volume:16, No:3, 645-678,2005