



BİLECİK ŞEYH EDEBALI
ÜNİVERSİTESİ

**BİLECİK
ŞEYH EDEBALI ÜNİVERSİTESİ**

**Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Ana Bilim Dalı**

**AŞIRI ÖĞRENME MAKİNELERİ İLE DİNAMİK SİSTEM
MODELLEME**

**Ahmet BAKIRCI
Yüksek Lisans Tezi**

**Tez Danışmanı
Prof. Dr. Cihan KARAKUZU**

**BİLECİK, 2019
Ref. No: 10285573**



BİLECİK ŞEYH EDEBALI
ÜNİVERSİTESİ

**BİLECİK
ŞEYH EDEBALI ÜNİVERSİTESİ**

**Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Ana Bilim Dalı**

**AŞIRI ÖĞRENME MAKİNELERİ İLE DİNAMİK SİSTEM
MODELLEME**

**Ahmet BAKIRCI
Yüksek Lisans Tezi**

**Tez Danışmanı
Prof. Dr. Cihan KARAKUZU**

BİLECİK, 2019



BİLECİK ŞEYH EDEBALI
ÜNİVERSİTESİ

**BİLECİK
SEYH EDEBALI UNIVERSITY**

**Graduate School of Sciences
Computer Engineering Department**

**DYNAMIC SYSTEM MODELING WITH EXTREME
LEARNING MACHINES**

**Ahmet BAKIRCI
Master's Thesis**

**Thesis Advisor
Prof. Dr. Cihan KARAKUZU**

BİLECİK, 2019



**BİLECİK ŞEYH EDEBALI ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**YÜKSEK LİSANS
JÜRİ ONAY FORMU**

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunun 31/07/2019 tarih ve 41-17 sayılı kararıyla oluşturulan jüri tarafından 21/08/2019 tarihinde tez savunma sınavı yapılan Ahmet BAKIRCI' nın "Aşırı Öğrenme Makineleri ile Dinamik Sistem Modelleme" başlıklı tez çalışması Bilgisayar Mühendisliği Ana Bilim Dalında YÜKSEK LİSANS tezi olarak oy birliği ile kabul edilmiştir.

JÜRİ

ÜYE

(TEZ DANIŞMANI) : Prof. Dr. Cihan KARAKUZU

ÜYE: Dr. Öğr. Üyesi Harun ÖZBAY

ÜYE: Dr. Öğr. Üyesi Emrah DOKUR

ONAY

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunun
.../.../..... tarih ve/..... sayılı kararı.

TEŐEKKÖR

Bu tez alıőmamda bana bilgi ve tecrübeleri ile yardımcı olan yol gösteren, rehberlik eden ve adeta bana mükemmel bir kaynak olan saygıdeęer danıőman hocam; Prof. Dr. Cihan KARAKUZU'ya, alıőmam boyunca bana sürekli destek olan aileme sonsuz teőekkürlerimi sunarım.

BEYANNAME

Bilecik Şeyh Edebali Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kılavuzu'na uygun olarak hazırladığım bu tez çalışmasında, tez içindeki tüm verileri akademik kurallar çerçevesinde elde ettiğimi, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun olarak sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

..../..../ 2019

Ahmet BAKIRCI

AŞIRI ÖĞRENME MAKİNELERİ İLE DİNAMİK SİSTEM MODELLEME

ÖZET

Sinir ağları (SA) birçok mühendislik ve bilim problemiyle ilgili çözüm üretmek için kullanılırlar. Bu mimarilerin eğitim için kullanılan algoritmaları, genellikle yinelemeli algoritmalarlardır. İleri beslemeli sinir ağlarında eşik ve bağlantı ağırlık değerlerinin belirlemek için türev tabanlı yinelemeli algoritmalar kullanılır. Türev tabanlı yinelemeli algoritmaların eğitim süresinin yavaş olması yeni arayışların başlamasına sebep olmuştur. Bu yavaşlığın üstesinden gelmek için aşırı öğrenme makineleri (AÖM) kavramı önemli bir gelişim göstermiştir. AÖM tek gizli katmanlı ileri beslemeli ağlar için geliştirilmiş bir öğrenme algoritmasıdır. AÖM öğrenme algoritması eğitim süresi açısından büyük bir üstünlük sunsa da genelleme yeteneği olarak aynı başarıyı gösterdiği söylenemez. İşte bundan dolayı geleneksel aşırı öğrenme makinelerinin bir araya gelerek oluşturduğu Meta-AÖM adı verilen yeni bir öğrenme algoritması geliştirilmiştir.

Bu çalışmamızda, Meta-AÖM ile eğitilen yapay sinir ağlarının dinamik sistem modelleme başarımlarını incelenmiştir. Literatürden seçilen yedi farklı dinamik sistem için bu çalışmada kullanılmak üzere eğitim ve test veri kümeleri hazırlanmıştır. Eğitim kümeleri kullanılarak, her bir dinamik sistem için AÖM eğitim yöntemiyle Meta-AÖM sistem tanımlama modeli elde edilmiştir. Her bir dinamik sistem için eğitim evresi ile oluşturulan Meta-AÖM modellerinin test veri kümeleri ile de genelleme başarımları elde edilmiştir. Modelleme Meta-AÖM için farklı ağ parametreleriyle eğitim ve test veri kümeleri benzer şekilde kullanılarak tekrarlanmış, elde edilen sonuçlar ile istatistiksel karşılaştırma yapılmıştır. Elde edilen sonuçlara dayalı olarak, önerilen yaklaşımın başarımlarındaki değişiklikleri gösterilmiştir. Sonuç olarak, Meta-AÖM'i yapılandırmak için hücre/düğüm sayısı ve gruptaki AÖM sayısı seçimi hakkında genel bir değerlendirme yapılarak öneriler sunulmuştur.

Anahtar Kelimeler: Aşırı Öğrenme Makineleri; Meta-AÖM; İleri Beslemeli Ağlar; Dinamik Sistem; Sistem Modelleme

DYNAMIC SYSTEM MODELING WITH EXTREME LEARNING MACHINES

ABSTRACT

Neural networks (NN) are used to find solutions to many engineering and science problems. The algorithms used for training these architectures are usually iterative algorithms. Derivative-based iterative algorithms are used to determine threshold and linkage weight values in feedforward neural networks. Derivative based iterative algorithms have a slow training period, which led to new searches. To overcome this slowness, the concept of extreme-learning machines (ELM) has made significant progress. ELM is a learning algorithm developed for single hidden layer feedforward networks. Although the ELM learning algorithm offers a great advantage in terms of training time, it cannot be said to have the same performance as generalization ability. This is why a new learning algorithm called Meta-ELM has been developed by combining traditional extreme-learning machines.

In this study, dynamic system modeling performance of artificial neural networks trained with Meta-ELM was investigated. Training and test data sets were prepared for seven different dynamic systems selected from the literature to be used in this study. Meta-ELM system identification model was obtained for each dynamic system by using ELM training method. For each dynamic system, generalization achievements have been obtained with the test datasets of Meta-ELM models created with training phase. Modeling For the Meta-ELM, training and test datasets were repeated with different network parameters and the results were compared with the statistical results. Based on the results obtained, the performance changes of the proposed approach are shown. As a result, a general evaluation was made about the selection of the number of cells / nodes and number of ELMs in the group in order to construct the Meta-ELM and suggestions were made.

Key Words: Extreme Learning Machines; Meta-ELM; Feedforward Networks; Dynamic System; System Modeling

İÇİNDEKİLER

	Sayfa No
TEŞEKKÜR	
BEYANNAME	
ÖZET	I
ABSTRACT	II
İÇİNDEKİLER	III
ŞEKİLLER DİZİNİ	IV
ÇİZELGELER DİZİNİ	V
SİMGELER ve KISALTMALAR DİZİNİ	VI
1. GİRİŞ	1
1.1. Literatür Taraması	1
1.2. Tez Çalışması Kapsamı	4
2. TEK KATMANLI İLERİ BESLEMELİ YAPAY SİNİR AĞI	6
2.1. Rasgele Düğümlere Sahip Tek Gizli Katmanlı İleri Beslemeli Ağ.....	8
3. AŞIRI ÖĞRENME MAKİNELERİ ve META-AÖM	10
3.1. AÖM Modeli	11
3.2. Meta-AÖM Modeli.....	13
3.2.1. Meta-AÖM algoritması	14
4. META AÖM İLE DİNAMİK SİSTEM TANIMLAMA	18
4.1. Dinamik Sistemler	18
4.2. Meta-AÖM ile Modelleme	20
4.2.1. Dinamik sistem modelleme örnek 1	20
4.2.2. Dinamik sistem modelleme örnek 2	24
4.2.3. Dinamik sistem modelleme örnek 3	28
4.2.4. Dinamik sistem modelleme örnek 4	32
4.2.5. Dinamik sistem modelleme örnek 5	36
4.2.6. Dinamik sistem modelleme örnek 6	40
4.2.7. Dinamik sistem modelleme örnek 7	44
5. SONUÇ ve ÖNERİLER	48
KAYNAKLAR	49
ÖZ GEÇMİŞ	

ŞEKİLLER DİZİNİ

	Sayfa No
Şekil 3.1. AÖM için Meta-öğrenme modeli öbek yapısı.....	14
Şekil 3.2. Üçer hücreli TKİBSA1ı iki giriş-tek çıkışlı bir Meta-AÖM ağ mimarisi	15
Şekil 4.1. DS #1 için Meta-AÖMnin en iyi modelleme başarımı.....	21
Şekil 4.2. DS #1 için Meta-AÖMnin en kötü modelleme başarımı	22
Şekil 4.3. DS #2 için Meta-AÖMnin en iyi modelleme başarımı.....	25
Şekil 4.4. DS #2 için Meta-AÖMnin en kötü modelleme başarımı	26
Şekil 4.5. DS #3 için Meta-AÖMnin en iyi modelleme başarımı.....	29
Şekil 4.6. DS #3 için Meta-AÖMnin en kötü modelleme başarımı	30
Şekil 4.7. DS #4 için Meta-AÖMnin en iyi modelleme başarımı.....	33
Şekil 4.8. DS #4 için Meta-AÖMnin en kötü modelleme başarımı	34
Şekil 4.9. DS #5 için Meta-AÖMnin en iyi modelleme başarımı.....	37
Şekil 4.10. DS #5 için Meta-AÖMnin en kötü modelleme başarımı	38
Şekil 4.11. DS #6 için Meta-AÖMnin en iyi modelleme başarımı.....	41
Şekil 4.12. DS #6 için Meta-AÖMnin en kötü modelleme başarımı	42
Şekil 4.13. DS #7 için Meta-AÖMnin en iyi modelleme başarımı.....	45
Şekil 4.14. DS #7 için Meta-AÖMnin en kötü modelleme başarımı	46

ÇİZELGELER DİZİNİ**Sayfa No**

Çizelge 4.1. Meta-AÖM öğrenmeli dinamik sistem modelleme için literatürden seçilen dinamik sistemler (DS)	19
Çizelge 4.2. DS #1 için elde edilen deneysel başarımlar metrikleri	23
Çizelge 4.2. DS #2 için elde edilen deneysel başarımlar metrikleri	27
Çizelge 4.4. DS #3 için elde edilen deneysel başarımlar metrikleri	31
Çizelge 4.6. DS #4 için elde edilen deneysel başarımlar metrikleri	35
Çizelge 4.7. DS #5 için elde edilen deneysel başarımlar metrikleri	39
Çizelge 4.8. DS #6 için elde edilen deneysel başarımlar metrikleri	43
Çizelge 4.8. DS #7 için elde edilen deneysel başarımlar metrikleri	47

SİMGELER ve KISALTMALAR DİZİNİ**Simgeler**

H :Gizli Katmanın Çıkış Matrisi

L :Gizli Katmandaki Nöron Sayısı

N : Veri Seti Sayısı

β :Çıkış Ağırlık Vektörü

$G(w_i, b_i, x)$:Aktivasyon Fonksiyonu

N_h : Nöron Sayısı

M :Topluluk Sayısı

Kısaltmalar

AÖM: Aşırı Öğrenme Makineleri

Meta-AÖM: Meta-Aşırı Öğrenme Makineleri

TKİBSA: Tek Katmanlı İleri Beslemeli Sinir Ağı

SA: Sinir Ağları

NN: Neural Network

RMSE: Root Mean Square Error

DS: Dinamik Sistem

MbAÖM: Meta-bilişsel Aşırı Öğrenme Makineleri

HGA-AÖM: Hata Geribildirim Aşamalı Aşırı Öğrenme Makineleri

1.GİRİŞ

Sürekli deęişim ve dönüşüm içerisinde olan teknoloji hayatımızın deęişmez bir parçası haline gelmiştir. Teknolojinin hızla gelişmesiyle birlikte birçok yeni konu ve yöntem ortaya çıkmıştır. Bunların insanlar tarafından anlaşılması hem maliyet hem de zaman açısından oldukça zorlaşmıştır. Bu sebepten dolayı bilişim teknolojilerine ihtiyaç duyulmuştur. Bu ihtiyaca yönelik kullanılan yapay sinir aęları, sürekli gelişme göstermektedir. İnsan beynini taklit ederek ortaya çıkan yapay sinir aęları birçok algoritmaya ilham kaynaęı olmuş, farklı algoritmaların ortaya çıkmasına ve algoritmaların zamanla kendini geliştirmesine zemin hazırlamıştır.

Yapay sinir aęları ileri beslemeli yapay sinir aęları ve geri beslemeli yapay sinir aęları olarak ikiye ayrılır. İleri beslemeli yapay sinir aęlarında nöronlar girişten çıkışa doğru katmanlar halindedir. Gelen bilgiler sırasıyla giriş katmanına, ara katmana ve çıkış katmanına iletilir. İşlenen veri dış dünyaya çıkar. Geri beslemeli aęlarda ise sadece kendinden sonra gelen hücrenin katmanına girdi olarak verilmez. Kendinden önceki katmanda veya kendi katmanında bulunan herhangi başka bir hücreye de girdi olarak bağlanabilir.

Yapay sinir aęlarını tıpkı bir insan beyni gibi eğitmek günümüzde çok kolay hale gelmiştir. İnsan beyninin tecrübeyle elde ettiği bilgiyi, yapay sinir aęları kullanarak makinalara öğretmek bilişim teknolojilerinin temel konusu haline gelmiştir.

Tek katmanlı ileri beslemeli bir sinir aęının makine öğrenmesi için aşırı öğrenme makineleri algoritması geliştirilmiştir. AÖM birçok problemin çözümünde geleneksel ileri beslemeli aęlara göre daha iyi başarımlar göstermektedir. Tek gizli katmanlı ileri beslemeli yapay sinir aęlarını kullanan aşırı öğrenme makineleri sınıflandırma ve regresyon problemlerini çözmek amacıyla önerilmiştir.

1.1 Literatür Taraması

Literatürde aşırı öğrenme makineleri yöntemi ile yapılan çalışmalar şu şekilde özetlenebilir.

Guang-Bin Huang and Chee-Kheong Siew (Huang ve Siew,2004), tarafından iyi bir sınıflandırma başarımları elde edebilmek için, tek gizli katmanlı ileri beslemeli yapay sinir aęları için aşırı öğrenme makineleri adı verilen öğrenme algoritmasını önermiştir. Aşırı öğrenme makinalarında giriş aęlıkları rasgele atanır. Giriş katmanındaki veriler

ağırlıklarla çarpılarak gizli katmandaki nöronlara iletilir. Burada algoritmik işlemlere ve aktivasyon fonksiyonuna tabi tutularak elde edilen veriler ile çıkış verileri kullanılarak çıkış ağırlıkları hesaplanır.

Shizhong Liaon ve Chang Feng (Liaon ve Feng,2014) klasik AÖMden daha iyi genelleme başarımı elde etmek adına Meta-AÖM olarak adlandırılan algoritmayı önermiştir. Meta-AÖM bir grup temel AÖM birleşiminden meydana gelmiştir.

Zou vd. (Zou ,Yao ,Zhang ,Guan ,2018) makalesinde geliştirilmiş Meta-AÖMden bahsedilmiştir. Geliştirilmiş Meta-AÖMnin çıkış sebebi şöyle açıklanabilir; Liao ve diğerleri (Neurocomputing 128: 81-87, 2014), aşırı öğrenme makinelerini eğiterek daha iyi bir genelleme başarımına sahip Meta-AÖM yaklaşımını önermişti. Ancak Meta-AÖMdeki sorunlardan biri, eğitim hatası minimuma indiği zaman bitmekteydi. Geliştirilmiş Meta-AÖM iki aşama içerir. İlk olarak, hata geribildirim aşamalı aşırı öğrenme makinelerinde (HGA-AÖM) her temel AÖM, bir eğitim verisi alt kümesi üzerinde eğitilmiştir. Daha sonra, üst AÖM, klasik AÖM gizli düğümler olarak öğrenir.

Mehmet Özçalıcı (Özçalıcı, 2017) tarafından hisse senedi fiyatlarının doğru bir şekilde tahmin edilebilmesi amacıyla çalışma yapılmıştır. Mehmet Özçalıcı bu çalışmada 14 farklı aktivasyon fonksiyonu kullanmıştır. Ayrıca çalışmada Goodyear, Amazon ve WalMart isimli şirketlere ve SP500 Endeksine ait geçmiş fiyat ve işlem hacmi bilgilerini kullanmak suretiyle 12 adet teknik göstere hesaplanmış ve modellerin girdi değişkeni olarak kullanılmıştır. Çıktı değişkeni olarak ta bir sonraki güne ait kapanış fiyatları kullanılmıştır.

Ömer Faruk Ertuğrul (Ertuğrul, 2016) tarafından aşırı öğrenme makineleri ile biyolojik sinyallerin gizli kaynaklarına ayrıştırılması ile ilgili çalışma yapılmıştır. Bu çalışmada tek gizli katmanlı ileri beslemeli yapay sinir ağlarını eğitmek için kullanılan aşırı öğrenme makineleri (AÖM) yöntemi ile tek bir sinyal birden fazla birbirinden bağımsız gizli kaynağa ayrıştırılmıştır. Bu amaçla EEG, EMG, ECG sinyalleri ile ivmeölçer, magnetometre ve jiroskop algılayıcılarından alınan zaman sinyalleri gizli kaynaklarına ayrıştırılmıştır. 2 adet epileptik ve 2 adet normal EEG sinyali 16 ayrı gizli kaynağa kadar ayrıştırılmıştır. Dört örnekte de elde edilen başarı oranları önerilen metodun gizli kaynak ayrıştırmada başarıyla kullanılabileceğini göstermiştir.

Necmettin Sezgin (Sezgin,2016) tarafından epileptik EEG işaretlerin aşırı öğrenme makineleri ile sınıflandırılması ile ilgili çalışma yapmıştır. Bu çalışmada epilepsi tanısı konulmuş hastalardan alınan EEG işaretleri, nöbet öncesi, nöbet anı ve nöbet sonrası olarak sınıflandırılmıştır.

Fernández vd. (Fernández, Salinas, Torres, 2018) finansal zaman serilerini tahmin etmek için Meta-AÖM yöntemi kullanılmıştır. Yayınladıkları makalede farklı zamanlardaki zaman serilerinin tahmin edilmesi sorununa dikkat çekilmiştir. Makalede, öncelikle bir tahmin yöntemi önermek için altta yatan süreçte bulunan ilgili özellikler ölçülmüştür. Stok zaman serilerinin davranışını karakterize etmek için Sample Entropy ve Hurst Exponent seçilmiştir. Deneysel sonuçlar göz önüne alındığında, ölçülen ilgili özellikleri (rastgelelik, uzun süreli hafıza ve ölçekleme özellikleri) yakalayabilen dâhili tahmin ediciler dâhil, bunları içermeyen yöntemlerin öngörülmesinin doğruluğunu başarıyla geliştirdiği görülmüştür.

Alçın vd. (Alçın, Şengür, İnce, 2014) tarafından İleri-Geri Takip (İGT) algoritması tabanlı seyrek aşırı öğrenme makinesi isimli çalışmada aşırı öğrenme makinelerinin sakıncalarını çözüm bulmak için İGT algoritması kullanılmıştır. Bu sakıncaları şöyle açıklayabiliriz; aşırı öğrenme makineleri çıkış ağırlıklarının hesaplanmasında en küçük kareler minimizasyonu kullandığı için aşırı örtüşme meydana gelmektedir. Diğer bir sakıncası ise aşırı öğrenme makinelerinin doğruluğunun gizli katman nöron sayısına bağlı olmasıdır.

Kaynar vd. (Kaynar, Arslan, Görmez, Işık, 2018) makalelerinde saldırı tespit sistemleri için üretilen KDDCup99 veri seti üzerinde çeşitli seçim algoritmaları kullanılarak yeni veri setleri elde edilmiştir. Elde edilen bu yeni veri setleriyle orijinal boyuttaki veri seti ile karşılaştırmak için en yakın k komşu, destek vektör makineleri ve aşırı öğrenme makineleri kullanılarak farklı modeller geliştirilmiştir. Bu modeller içerisinden yüksek başarı oranları, diğer sınıflama yöntemlerine oranla son derece hızlı olması ve eğitim algoritmasının basit olmasından dolayı aşırı öğrenme makineleri öne çıkmıştır.

Ramasamy (Ramasamy, Savitha & Suresh, Sundaram & J. Kim, H., 2013) göre MbAÖMnin iki bileşeni vardır: bilişsel bir bileşen ve bir meta-bilişsel bileşen. MbAÖMnin bilişsel bileşeni, üç katmanlı bir aşırı öğrenme makinesi (AÖM) sınıflandırıcısıdır. Bilişsel bileşenin gizli katmanındaki nöronlar, q-Gaussian aktivasyon

fonksiyonunu kullanırken, giriş ve çıkış katmanlarındaki nöronlar doğrusaldır. MbAÖMnin meta-bilişsel bileşeni, bir meta-bilişsel çerçevede ne öğrenileceğini, ne zaman öğrenileceğini ve nasıl öğrenileceğini belirleyen kendi kendini düzenleyen bir öğrenme mekanizmasına sahiptir. Eğitim setindeki örnekler birer birer sunulduğundan, meta-bilişsel bileşen bilişsel bileşenden monitory sinyallerini alır ve örnek için uygun öğrenme stratejilerini seçer. Böylece numuneyi siler, yeni bir nöron eklemek için numuneyi kullanır veya numuneye dayalı çıktı ağırlıklarını günceller veya numuneyi ilerde kullanmak üzere ayırır. Bu nedenle, geleneksel AÖMden farklı olarak, MbAÖMnin mimarisi önceden belirlenmiş değildir, bunun yerine ağ eğitim süreci boyunca kurulur. Bir nöron eklerken, MbAÖM numuneye göre merkezleri seçer ve Gauss fonksiyonunun genişliği rastgele seçilir. Çıktı ağırlıkları, menteşe kaybı işlevine bağlı olarak en küçük kareler tahmini kullanılarak tahmin edilir. Menteşe kaybı hata fonksiyonu, arka olasılıkların orta-kare hatadan daha iyi tahmin edilmesini kolaylaştırır ve bu nedenle MbAÖM sınıflandırıcısının geliştirilmesi tercih edilir. Ağ parametrelerini güncellerken, çıkış ağırlıkları öz yinelemeli en küçük kareler tahmini kullanılarak güncellenir. MbAÖMnin başarımı, UCI makine öğrenme deposundaki bir dizi kriter sınıflandırma problemi üzerinde değerlendirilmektedir. Başarım çalışması sonuçları, AÖM çerçevesindeki meta-bilişin, AÖMnin karar verme yeteneğini önemli ölçüde arttırdığını vurgulamaktadır.

1.2. Tez Çalışması Kapsamı

Yukarıda AÖM ve Meta-AÖM ile yapılan literatür çalışmaları özetlenmiştir. Yapılan özetten de anlaşılacağı üzere Meta-AÖM algoritması dinamik sistemlerin modellenmesinde henüz kullanılmamıştır. Bu tez çalışmasında Meta-AÖM algoritmasının dinamik sistem modelleme probleminde kullanımı üzerinde çalışılmıştır. Çalışmada Meta-AÖM algoritması ile AÖM algoritması kıyaslanmıştır. Bu amaçla yapılan çalışmaların anlatıldığı bu tezin organizasyonu şu şekildedir: Birinci bölümde yapay sinir ağlarından bahsedilmiştir. Ayrıca AÖM ve Meta-AÖM ile ilgili literatür çalışması yapılmıştır. İkinci bölümde tek gizli katmanlı ileri beslemeli yapay sinir ağları anlatılmıştır. Üçüncü bölümde AÖM ve Meta-AÖM tanıtılmıştır. Dördüncü bölümde Meta-AÖM makinelerinin dinamik sistem modelleri üzerindeki davranışı incelenmiştir. Son bölümde elde edilen sonuçların yorumu ve ileriye dönük yapılabilecek çalışmalar üzerinde durulmuştur.

2. TEK GİZLİ KATMANLI İLERİ BESLEMELİ YAPAY SİNİR AĞI

Geliştirilen yapay sinir ağlarının ilk ve en basit çeşidi ileri beslemeli sinir ağıdır. Tek gizli katmanlı ileri beslemeli yapay sinir ağları, giriş katmanı, gizli katman ve çıkış katmanı olmak üzere 3 temel katmandan oluşmaktadır. Giriş katmanında bulunan düğümler ağa bilgi sağlayan düğümlerdir. Burada herhangi bir hesaplama yapılmaz. Gizli katmandaki düğümlerin dış dünyayla doğrudan bir bağlantısı yoktur. Burada matematiksel işlemler yapılır ve giriş düğümleri ile çıkış düğümleri arasındaki bilgi aktarımına yardımcı olurlar. Çıkış katmanı ise elde edilen değerlerin dış dünyaya aktarıldığı katmandır. İleri beslemeli sinir ağlarında tek bir giriş katmanı ve tek bir çıkış katmanı bulunur. Tercihe bağlı olarak sıfır ya da daha fazla sayıda gizli katman bulunabilir. Gizli katmanlarda birçok bir veya daha fazla nöron bulunabilir. Nöron sayısı problemin seviyesine göre belirlenir. Diğer katmanlar için nöron sayısını çok basit bir şekilde belirleyebiliriz. Mesela giriş katmanı için nöron sayısını sistemin girdisi olan veri setine göre karar verebiliriz. Sistemimizin iyi bir şekilde eğitilmesini istiyorsak sınıflandırma yapmak istediğimiz verileri ona göre gruplara ayırabiliriz. 100 tane veri setimiz varsa 100 tane giriş nöronu kullanırsak minimum hataya yaklaşmış oluruz. Ama şöyle bir sakıncası olacaktır; eğitim süresi artacaktır. Bunun yerine eğitim setlerimizi gruplayarak dahi iyi sonuçlar alabiliriz. Çıkış katmanındaki nöronların belirlenmesine gelecek olursak, çıkış katmanındaki nöronların sayısı giriş katmanındakine benzer yöntemle bulunabilir. Ne kadar farklı sınıf gösterilmesini istiyorsak ona göre çıkış nöronu kullanabiliriz. Gizli katmandaki nöronlara gelince, eğer bir yapay sinir ağında gizli katman bulunmuyorsa girişler çıkışlara direk bağlıdır, yani bu ağ doğrusal fonksiyonlar üzerinden ayırım yapma ve karar verme yeteneğine sahiptir. Birden fazla gizli katmanın bulunması durumunda ise problem oldukça karmaşık bir yapıya dönüşmekte, dolayısıyla ağın tasarımı oldukça zor bir hal almaktadır. Genel olarak ileri beslemeli ağlarda tek katman kullanılmakta olup giriş katmanından gelen veriler işlenerek çıkış katmanına iletilmektedir.

Son yıllarda ileri beslemeli yapay sinir ağları birçok girdi verileri kullanarak sonuçları tahmin etme üzerine yoğunlaşmıştır. Mesela bir hastanın tahlillerin bakarak, tahlildeki değerleri giriş verisi olarak alıp teşhisi ise çıkış verisi olarak kullanılmaktadır. Böylece birçok hastanın verisi bir veri setinde kullanılarak işlenir. Diyelim ki 50 tane hastamız var, her bir hastamızın 100 tane verisi var. Bu verilere göre 3 farklı teşhisimiz

olsun. Buna göre bir sinir ağı yapılandırmak istersek 50 tane giriş nöronu kullanılırız, bir tane gizli katman kullanılırız, çıkış katmanı olarak da 3 nöron kullanmamız gerekecektir. Her bir giriş nöronunda da 100 tane verimizi işlemiş olacağız. Gizli katmandaki nöron sayısını giriş katmanındaki nöron sayısına göre belirlememiz isabetli olur. Eğer gizli katmandaki nöron sayısını çok yüksek verirse ağın eğitimi için geçen zamanı artar. Çok düşük verirse hata oranı artar. Bunun için giriş nöronların sayısına ortalama bir değer vermek isabetli olur. Böylelikle birçok hastanın verileri kullanılarak yapay sinir ağları eğitilmekte ve buna göre sonuç alınmaktadır.

Aşırı öğrenme makineleri (AÖM), sınıflandırma problemleri için önerilen bir öğrenme algoritmasıdır. Aşırı öğrenme makinelerinin temel mantığında, giriş ağırlıkları ve eşik değerleri rasgele atanır ve algoritma olarak da en küçük kareler yöntemi kullanılır. Rasgele olarak atanan giriş ağırlıkları ve eşik değerleri aşırı öğrenme makinelerinin sınıflandırma yeteneğini artırır.

Sınıflandırma yeteneğini artırmak için şimdiye kadar aşırı öğrenme makineleri hakkında birçok çalışma yapıldı. Bunlar temel olarak, bağımsız tahmin edicilerin ortalamasının (yani modeller), tipik bireysel tahmin ediciden daha az varyansa sahip olduğu istatistiksel bir sonuca dayanır. Modellerin birleşimi, tek modelden daha iyi başarımlar göstermektedir. Sun (Sun,2008) ve Lan (Lan,2009) tüm veri setinde birden fazla AÖM kullanmış ve bunların ortalamasını da son tahminci olarak kullanmıştır. Deneysel sonuçlar, AÖMnin meydana getirdiği grubun sınıflandırma başarımının tek bir AÖMnin sınıflandırma başarımından daha iyi olduğunu göstermiştir. Heeswijk vd.(Heeswijk,Miche,Lindh-Knuutila, Hilbers,Honkela,Oja Lendasse, 2009) ,(Heeswijk, Miche,Oja,Lendasse,2011) ayrıca her bir AÖMnin ağırlığını yinelemeli olarak ayarlayarak, LOO hatası temelinde ağırlıkları belirleyerek ya da bir doğrulama alt kümesinde en küçük kare çözümü kullanarak daha iyi sonuçlar almıştır. Zhao vd.(Zhao,Shen,Miao,Gay,2008), AÖMnin ağırlıklarını eşitlik (2.1)'deki minimize edecek şekilde belirlemişlerdir.

$$\sum_{j=1}^J \sum_{(x_i, t_i) \in L_j} \left(t_i - \sum_{k=1}^K a_k M_k^{(-j)}(x_i) \right)^2, \quad (2.1)$$

Buradaki t_i sayısal değeri, j 'nci kattaki veri setindeki beklenen çıktıyı gösterir. Yukarıda belirtilenlerin tümü, orijinal AÖM sınıflandırma başarımını iyileştirmek için

ortaya atılmıştır. Bununla birlikte, büyük ölçekli problemler için, tüm veri setinde ard arda AÖM eğitimi olmasından dolayı yüksek hesaplama maliyeti ortaya çıkmıştır. Bu yüksek hesaplama maliyetinin Liao ve Feng(Liao, Feng, 2014), Collobert vd. (Collobert, Bengio, Bengio, 2002) ilham alarak AÖMyi birleştiren Meta-öğrenme modelini önermiştir. Bu model, AÖMyi veri kümesinin bir parçası üzerinde eğitir. Önerilen model AÖMnin ağırlıklarını AÖMnin gizli düğümlerde yaptığı gibi analitik olarak belirler. Bu şekilde, “Meta-AÖM” olarak adlandırılan hiyerarşik bir AÖM mimarisi elde edilir. Birkaç regresyon veri setiyle yapılan deneysel sonuçlar Meta-AÖMnin iyi performans elde ettiğini ve çoklu AÖMnin eğitilmesinden kaynaklanan hesaplama maliyetini düşürdüğünü göstermektedir.

2.1. Rasgele Düğümlere Sahip Tek Gizli Katmanlı İleri Beslemeli Ağ

L adet gizli katman hücrelerine sahip tek gizli katmanlı ileri beslemeli ağın çıkışı eşitlik (2.2)’deki gibi hesaplanır.

$$f_L = \sum_{i=1}^L \beta_i g_i(x) = \sum_{i=1}^L \beta_i G(w_i, b_i, x), \quad x \in R^d, \quad \beta_i \in R^m, \quad (2.2)$$

Buradaki g_i gizli katmanın çıkış fonksiyonu olan $G(w_i, b_i, x)$ ’i gösterir. Aktivasyon fonksiyonu olan g , ilave düğümler için g_i olarak tanımlanır ve eşitlik (2.3)’deki gibi gösterilir.

$$g_i = G(w_i, b_i, x) = g(w_i \cdot x + b_i), \quad w_i \in R^d, \quad b_i \in R, \quad (2.3)$$

Radyal temelli fonksiyonlar için gizli düğümler, g_i eşitlik (2.4)’teki gibi tanımlanır.

$$g_i = G(w_i, b_i, x) = g(b_i \|x - w_i\|), \quad w_i \in R^d, \quad b_i \in R^+, \quad (2.4)$$

N tane isteğe bağlı farklı örnekler (x_i, t_i) için $x_i = [x_{i1}, x_{i2}, \dots, x_{id}]^T \in R^d$ ve $t_i \in R$ olmak üzere, L gizli nöronlu standart TKİBSA matematiksel olarak eşitlik (2.5)’deki gibi modellenmiştir.

$$\sum_{i=1}^L \beta_i g_i(x_j) = \sum_{i=1}^L \beta_i G(w_i, b_i, x_j) = o_j, \quad j = 1, 2, \dots, N., \quad (2.5)$$

$L \geq N$ olduğu zaman, TKİBSA N tane örneği sıfıra yakın hatayla hesaplar.

$\sum_{j=1}^N \|o_j - t_j\| = 0$, dır; (w_i, b_i) ve β_i eşitlik (2.6)' da gösterildiği gibi hedef çıkış değerini verecek şekilde belirlenmiştir.

$$\sum_{i=1}^L \beta_i G(w_i, b_i, x_j) = t_j, \quad j = 1, 2, \dots, N., \quad (2.6)$$

Yukarıdaki denklemler için, N tane eşitlik genel olarak eşitlik (2.7)'deki gibi yazılır.

$$H\beta = T, \quad (2.7)$$

Burada H matrisi eşitlik (2.8)'de, β ve T vektörleri eşitlik (2.9)'da gösterilmiştir.

$$H = \begin{bmatrix} h(x_1) \\ \vdots \\ h(x_N) \end{bmatrix} = \begin{bmatrix} G(w_1, b_1, x_1) & \Lambda & G(w_L, b_L, x_1) \\ \vdots & \vdots & \vdots \\ G(w_1, b_1, x_N) & \Lambda & G(w_L, b_L, x_N) \end{bmatrix}_{N \times L}, \quad (2.8)$$

$$\beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_L \end{bmatrix}_{L \times 1} \quad \text{ve} \quad T = \begin{bmatrix} t_1 \\ \vdots \\ t_N \end{bmatrix}_{N \times 1}, \quad (2.9)$$

H, TKİBSA'nın gizli katman çıkış matrisi olarak adlandırılır. H'nin her bir sütunu, x_1, x_2, \dots, x_N girişlerine göre gizli düğüm çıkışıdır.

$h(x) = [G(w_1, b_1, x), \dots, G(w_L, b_L, x)]$ gizli katman fonksiyonu olarak adlandırılır.

3.AŞIRI ÖĞRENME MAKİNELERİ VE META-AÖM

Matematiksel bir bakış açısına göre, ileri beslemeli sinir ağlarının genelleştirme yetenekleri üzerine yapılan araştırmalar iki yöne odaklanmıştır: Kompakt girdi kümeleri üzerinde evrensel yaklaşım ve belirli bir kümede yaklaştırma. Birçok araştırmacı, standart ileri beslemeli sinir ağlarının evrensel yaklaşım yeteneklerini araştırdı. Gerçek uygulamalarda, sinir ağları sınırlı eğitim kümesi ile eğitilirler. Sınırlı bir eğitim kümesinde fonksiyon yaklaşımı için, L gizli nörona ve doğrusal olmayan aktivasyon fonksiyonlarına sahip tek gizli katmanlı ileri beslemeli bir sinir ağı modeli için aşırı öğrenme makinaları (AÖM) Huang (Huang,2014) tarafından önerilmiştir.

Geleneksel olarak, ileri beslemeli ağların tüm parametrelerinin ayarlanması gerekir ve bu nedenle farklı parametre katmanları arasında bağımlılık vardır. Yapay sinir ağlarında öğrenme süreci, geri yayılım algoritması kullanılarak, bilinen çıkış değerleri arasındaki hatayı sürekli azaltarak her bir katmandaki ağırlık değerlerinin adım adım güncelleyerek gerçekleştirilmektedir. Yıllar boyunca gradyan temelli ileri beslemeli ağlar kullanılmıştır. Bununla birlikte gradyan temelli ileri beslemeli ağların öğrenme adımlarının çok yavaş olduğu ve de kolayca yerel minimumlara yaklaşabileceği görülmüştür. Dolayısıyla daha iyi öğrenme başarımı elde etmek için giriş ağırlıkları ve eşik değerlerinin rastgele üretildiği, ayrıca çıkış ağırlıklarının hesaplanmasında analitik yöntemlerin kullanıldığı AÖM gradyan beslemeli ileri beslemeli ağlarından daha iyi başarımlar göstermiştir. Aşırı öğrenme makinalarında giriş katmanındaki ağırlıklar rastgele atanarak, gizli katmanla çıkış katmanı arasındaki ağırlıklar analitik bir denklem sistemi yardımıyla çok daha hızlı ve etkin bir şekilde belirlenebilmektedir. Aşırı öğrenme makinalarının üstünlükleri aşağıdaki gibi sıralanabilir:

- ✓ Aşırı öğrenme makinaları hızlı öğrenme yeteneğinin yanı sıra geleneksel geri yayılım algoritması ile öğrenen ileri beslemeli ağlara göre daha iyi genelleme başarısına sahiptir.

- ✓ Aşırı öğrenme makinalarının öğrenme süreci son derece hızlıdır. Saniyeler seviyesinde, hatta bazen saniyeden bile daha kısa olmaktadır.

- ✓ Geleneksel türev tabanlı öğrenme algoritmaları sadece türevi alınabilen aktivasyon fonksiyonlarını kullanma yeteneğine sahipken aşırı öğrenme makinalarının

kullandığı öğrenme algoritmaları kesikli ve türevi alınamayan birçok aktivasyon fonksiyonunu kullanabilmektedir.

3.1. AÖM Modeli

$L > N$ olduğu zaman, Eşitlik (2.7) deki doğrusal sistem, sıfır hata ile bir çözüm sunar. Ancak bu yüksek maliyete neden olur. Bir TKİBSA $L < N$ durumunda, eşitlik (3.1) ile eğitildiği durumda, ε 'nın çok küçük değeri için eşitlik (3.2)'deki maliyet fonksiyonu en aza iner.

$$\|H\beta - T\| < \varepsilon, \quad (3.1)$$

$$C = \sum_{j=1}^N \left(\sum_{i=1}^L \beta_i G(w_i, b_i, x) - t_j \right)^2, \quad (3.2)$$

$\hat{w}_i, \hat{b}_i, \hat{\beta} (i = 1, K, L)$ gibi özel değerlerin bulunmasını eşitlik (3.3)'teki denklem sağlar.

$$\|H(\hat{w}_1, K, \hat{w}_L, \hat{b}_1, K, \hat{b}_L) \hat{\beta} - T\| = \min_{w_i, b_i, \beta} \|H(w_1, K, w_L, b_1, K, b_L) \beta - T\|, \quad (3.3)$$

AÖM kuramına göre, özellik eşlemelerinin neredeyse tamamı doğrusal olmayan parçalı sürekli fonksiyonlar AÖMnin evrensel yaklaşım yeteneğini karşılar. Bölüm 2.1 ve bölüm 3.1'de gösterildiği üzere, giriş ağırlıkları ve gizli katman eşik değerlerini ayarlamak geleneksel fonksiyon yaklaşımı kuramlarının aksine, sadece aktivasyon işlevi tamamen farklılaşabiliyorsa rastgele atanabilir. Sabit giriş ağırlıkları w_i ve gizli katmandaki eşik değerleri b_i olmak bir TKİBSA eğitilerek eşitlik (3.4)'teki $\hat{\beta}$ değeri bulunur. Bu eğitim süreci Algoritma 1'de verilmiştir.

$$\|H\hat{\beta} - T\| = \min_{\beta} \|H\beta - T\|, \quad (3.4)$$

Algoritma 1. AÖM Algoritması

Giriş: $N = \{(x_i, t_i) \mid x_i \in R^d, t_i \in R, i = 1, K, N\}$:Eğitim Verisi

$G(w_i, b_i, x)$:Aktivasyon fonksiyonu

Çıkış: β : Çıkış ağırlık vektörü

1. İlk olarak gizli nöronların parametre değerleri rasgele üretilir.

$$(w_i, b_i), i = 1, K, L.$$

-- **for** $i = 1 : L$ **do**

-- w_i, b_i : rasgele atanmış değerler

--**end**

2. İkinci adımda gizli katmanın çıkış matrisi hesaplanır.

--**for** $i = 1 : L$ **do**

-- **for** $j = 1 : N$ **do**

$$-- H(i, j) = G(w_i, b_i, x_j);$$

-- **end**

--**end**

3. Son adımda çıkış ağırlık vektörü olan β hesaplanır.

$$-- \beta = H^+T;$$

Çoğu durumda, gizli düğümlerin sayısı, farklı eğitim örneklerinin sayısından çok daha azdır, $L < N$, AÖM eşitlik (3.2)'deki maliyet fonksiyonunu en aza indirgeyerek çıkış ağırlığı olan β 'yi öğrenir. Buradaki β değeri eşitlik (3.4)'teki özel $\hat{\beta}$ değeridir. AÖM burada eşitlik 3.5 teki β değerini öğrenir.

$$\hat{\beta} = H^+T, \quad (3.5)$$

Buradaki H^+ , H matrisinin tersi olan genelleştirilmiş Moore-Penrose matrisidir.

Eşitlik (3.5)'teki $\hat{\beta}$ çözümü, tüm çözümler üzerinde minimum norma sahip eşitlik (2.7)'deki lineer sistemin en küçük kareler çözümlerinden biridir. $\hat{\beta}$ sadece eğitim hatasını en aza indirgemez, aynı zamanda ağırlıklarda en küçük boyuta sahiptir. Böylece $\hat{\beta}$ diğer en küçük kareler çözümüne göre en iyi sınıflandırma başarımı gösterir.

AÖM algoritması üç aşamalı bir öğrenme algoritmasıdır (Bknz. Algoritma 1). 2. adımda, for döngüsü yerine matris tarafından verimli bir şekilde kullanılabilir. (for-

döngüsü, MATLAB ortamında çok zaman alır). Ardından, AÖMnin çıkış fonksiyonu eşitlik (3.6)'da gösterilebilir:

$$f(x) = h(x) \cdot \beta, \quad (3.6)$$

Buradaki $h(x)$ değeri eşitlik (3.7)'de gösterilmiştir.

$$h(x) = [G(w_1, b_1, x_1), K, G(w_L, b_L, x_1)]., \quad (3.7)$$

3.2. Meta-AÖM Modeli

Son yıllarda, sınıflandırma başarımı artırmak için AÖMler ile ilgili birçok çalışma yapılmıştır. Bu çalışmalarda, birçok AÖM tekrar tekrar eğitilmiştir ve tek bir AÖMden daha iyi bir sınıflandırma başarımı yakalamıştır. Bununla birlikte, büyük ölçekli problemler için, çoğu çalışmanın tüm veri kümesinin tekrar tekrar eğitilmesinden dolayı hesaplama maliyeti yüksek olmaktadır. Bu maliyet sebebiyle, farklı bir yaklaşım arayışına giren Liao ve Feng (Liao, Feng, 2014), birçok AÖMyi bir araya getirerek Meta-AÖM olarak adlandırılan bir ağ mimarisi önerdi.

Meta-öğrenme çok sayıda öğrenmenin sonuçlarını birleştirmek için kullanılan genel bir tekniktir ve öğrenmeler tarafından üretilen bilgilerden oluşan bir öğrenme olarak tanımlanabilir. Meta-AÖM modeli, birkaç temel AÖM ve bir Meta-Öğreniciye sahip olan bir mimariden elde edilir. Meta-Öğrenici Şekil 3.1'de gösterildiği gibi temel AÖMlerin çıktılarından öğrenir. Her bir temel AÖM, bir temel tahmin üretir ve Meta-öğrenici de bir meta-tahmin üretir.

Meta-AÖM bir grup klasik AÖMyi gizli düğüm olarak alan yeni bir AÖM ağıdır. Şekil 3.2 de görüleceği üzere Meta-AÖM bir grup tek gizli katmanlı ileri beslemeli sinir ağının bir birleşimidir. Bu mimaride eğitim şu şekilde yapılır:

-Eldeki tüm giriş-çıkış örnekleri verisi M adet alt gruba bölünür.

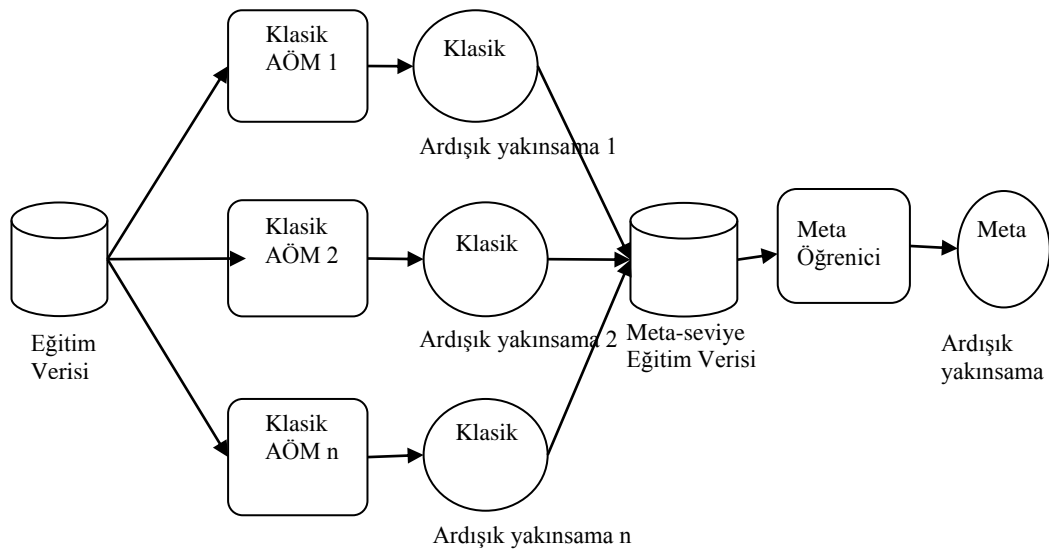
-Her bir TKİBSA'nın eşik ve bağlantı ağırlıkları rastgele atanır.

-Her TKİBSA birer alt grup veri ile Algoritma 1 kullanılarak eğitilir. Böylece Her bir TKİBSA'nın çıkış bağlantı ağırlıkları ile belirlenir.

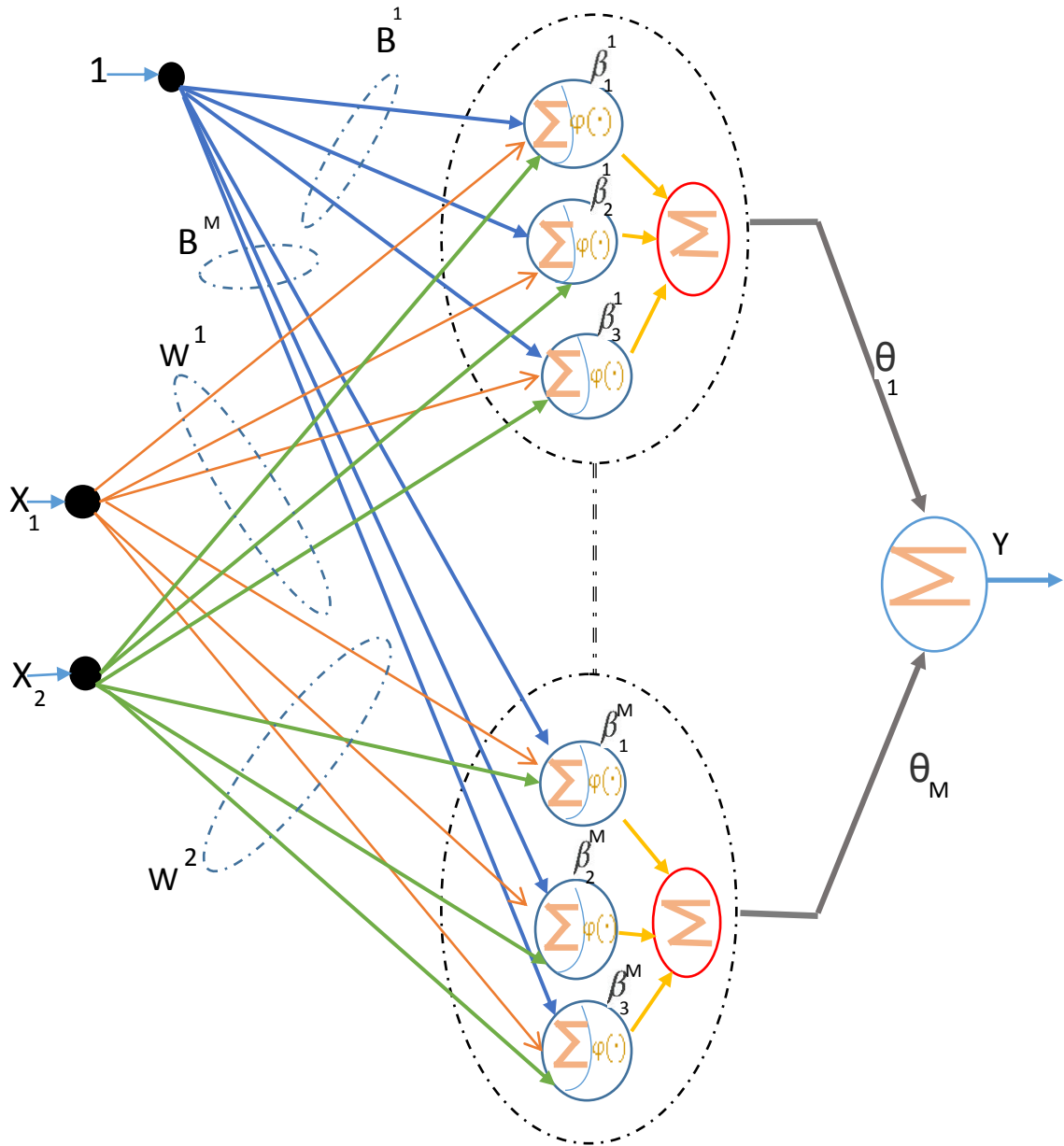
-Son olarak eğitilen TKİBSA'lar ve tüm veri kullanılarak Meta-AÖMnin çıkış bağlantı ağırlıkları yine Algoritma 1 ile belirlenir.

3.2.1 Meta-AÖM Algoritması

Meta-öğrenme, çok sayıda öğrencinin sonuçlarını birleştirmek için kullanılan genel bir tekniktir ve bu öğrenciler tarafından üretilen bilgilerden bir öğrenme olarak tanımlanabilir. Meta-AÖM modeli, birkaç temel AÖM ve bir Meta-Öğrencisi olan temel bir mimariyle elde edilir ve Meta-Öğrencisi, Şekil 3.1'de gösterildiği gibi, temel AÖMlerin çıkışlarından öğrenir. Meta-Öğrenen bir meta-onaylayıcı üretir. Meta-AÖM, AÖMleri ayrı veri kümeleri altlarına ve hiyerarşik mimarinin olduğu tüm veri setinde bir “üst” AÖM’e dayandırır. Meta-AÖMnin torbalama ve artırma yöntemleri ile güçlü bir şekilde ilişkili olduğuna dikkat edilmelidir. Hem torbalama hem de yükseltme, birden fazla öğrenci oluşturur ve birleşik bir öğrenci oluşturmak için öğrencileri birleştirir. Torbalama ve güçlendirme arasındaki temel fark; torbalama, eğitim veri setlerini seçmek için yükseltme yeniden örnekleme kullanır ve paralel olarak öngörücüleri üretebilir. Yükseltme, zayıf bir öğrenme algoritmasının başarımını artırmak için bir yöntem olarak Schapire (Schapire,1990) tarafından sunulmuştur. Yükseltme yöntemi olarak Adaboost, öngörücüyü ardışık olarak oluşturur ve önceden oluşturulmuş öngörücüler temelinde eğitim örneklerinin ağırlıklarını değiştirir. Ancak, Meta-AÖM hiyerarşik bir öğrenme modelidir. Eğitim veri setini rastgele ayarladıktan sonra, Meta-AÖM tüm veri kümesini ayrı alt kümelere ayırır, alt kümeler üzerinde tahminçiler oluşturur ve AÖMnin yaptığı gibi tahminçilerin ağırlıklarını analitik olarak belirler.



Şekil 3.1. Meta-AÖM için öğrenme modeli öbek yapısı.



Şekil 3.2. Üçer hücreli TKİBSAlı iki giriş-tek çıkışlı bir Meta-AÖM ağ mimarisi

Önerilen model eşitlik (3.8)'deki maliyet fonksiyonunu en az indirmektedir:

$$C = \sum_{i=1}^N \left(\sum_{m=1}^M \beta_m(x_i) AÖM(x_i) - t_i \right)^2, \quad (3.8)$$

Burada N eğitim verilerimizin sayısı, M Meta-AÖM modelindeki klasik AÖMnin sayısı, $AÖM_m(x_i)$, x_i girişi için m 'inci klasik AÖMnin çıkışı ve $\beta_m(x_i)$ ise x_i girişli m 'inci klasik AÖM için çıkış ağırlık vektörüdür.

Eşitlik (3.8)'deki maliyet fonksiyonu en aza indirgeyerek $\|H\hat{\beta} - T\| = \min_{\beta} \|H\beta - T\|$. denklemindeki özel $\hat{\beta}$ vektörü bulunur.

En üst AÖM için gizli katman çıkış matrisi olan H eşitlik 3.9'da gösterilmiştir.

$$H = \begin{bmatrix} h(x_1) \\ M \\ h(x_N) \end{bmatrix} = \begin{bmatrix} AÖM_1(x_1) & \Lambda & AÖM_M(x_1) \\ M & \Lambda & M \\ AÖM_1(x_N) & \Lambda & AÖM_N(x_N) \end{bmatrix}_{N \times M}, \quad (3.9)$$

ve $h(x)$ değeri eşitlik (3.10)'da verilmiştir.

$$h(x) = [AÖM_1(x), AÖM_2(x), K, AÖM_M(x)], \quad (3.10)$$

Meta-AÖM modeli için çıktı fonksiyonu eşitlik (3.11)'de verilmiştir.

$$f(x) = h(x) \cdot \beta, \quad (3.11)$$

Burada $\beta = H^+T$

Meta-AÖM modeli için, Algoritma 2. de AÖMe dayalı iki aşamalı bir öğrenme algoritması verilmiştir ve Meta-AÖM algoritması olarak adlandırılmıştır. İlk aşamada, Meta-AÖM algoritması verileri M ayrık kümelere böler ve her bir kümedeki temel AÖMleri eğitir. 2. Aşamada, 1. Aşama'da elde edilen temel AÖMleri gizli düğümler olarak alarak bir "üst" AÖMyi eğitir.

Algoritma 2. Meta-AÖM algoritması

Giriş: $N = \{(x_i, t_i) \mid x_i \in R^d, t_i \in R, i = 1, K, N\}$:Eğitim Verisi

M : İlk aşamadaki AÖMnin sayısı

Çıkış: β : İkinci aşamadaki çıkış ağırlık vektörü

1. İlk aşama:

-İlk olarak eğitim verileri N/M büyüklüğünde M tane alt kümelere bölünür.

$$S = \left\{ S_i \mid S_i \subseteq N, \mathbf{Y}_{i=1}^M S_i = N, S_i \cap S_j = \emptyset, i \neq j, i, j = 1, \dots, M \right\}$$

-Her AÖM bu alt gruplardan biri üzerinde ayrı ayrı eğitim yapar.

-- **for** her $S_i \in S$ **do**

-- AÖM S_j üzerinde β_j 'i öğrenir.

--**end**

2. İkinci aşama:

-İkinci adımda gizli katmanın çıkış matrisi olan H hesaplanır.

--**for** $i = 1 : M$ **do**

-- **for** $j = 1 : N$ **do**

-- $H(i, j) = AÖM_j(x_j) = h_j(x_i) \cdot \beta_j;$

-- **end**

--**end**

3. Son adımda çıkış ağırlık vektörü olan β hesaplanır.

-- $\beta = H^+T;$

4. META AÖM İLE DİNAMİK SİSTEM TANIMLAMA

Bu bölümde, Meta-AÖM algoritmasının dinamik sistem modelleme başarımı incelenmiştir. Çalışma Çizelge 4.1’de verilen yedi adet dinamik sistemin modellenmesi üzerine yapılmıştır. Çizelge 4.1’de verilen dinamik sistemler üzerinde Meta-AÖM algoritmasının başarımlarını kıyaslamaları yapılmıştır. Meta-AÖM algoritması için farklı hücre sayısı ve grup sayısı kıyaslama amaçlı parametre olarak kullanılmıştır.

Her bir dinamik sistem, farklı hücre sayısı ve grup sayısı için rasgele oluşturulmuş başlangıç giriş ağırlıkları ve eşikler ile 100’er kez ayrı ayrı koşturulmuştur. Her bir algoritmanın 100’er kez koşması sonucu elde edilen sonuçlar istatistikî olarak irdelenmiştir.

Çizelge 4.1’de verilen örnek dinamik sistemler sırasıyla Meta-AÖM algoritması kullanılarak modellenmiş ve her bir sistem için elde edilen model sonuçları ortalama, en iyi, en kötü ve standart sapma şeklinde metriklerle ölçülerek aşağıdaki alt bölümlerde tablo olarak verilmiştir. Bu tablolarda eğitim süresi sonuçları ve modelleme başarımlarını metrikleri toplu olarak verilmiştir. Root Mean Square Error (RMSE) metriği cinsinden en iyi metrik değerler eğitim seti için koyu siyah yazı tipi ve test seti için koyu kırmızı yazı tipi ile vurgulanmıştır. En kötü metrik değerler ise eğitim seti için koyu yeşil yazı tipi ve test seti için koyu turuncu yazı tipi ile vurgulanmıştır. Ayrıca alt bölümlerde grafiksel olarak şekiller de görsel olarak başarımlarını kıyaslaması için verilmiştir. Şekillerde mavi düz çizgi ile verilenler istenen sonucu, kırmızı kesikli çizgiyle verilen ise elde edilen sonucu göstermektedir.

4.1. Dinamik Sistemler

Bu bölümde Meta-AÖM algoritmasının dinamik sistemleri modelleme davranışları incelenecektir. Çizelge 4.1’de bu çalışmada kullanılan dinamik sistemlerin matematiksel tanımları verilmiştir.

Öncelikle dinamik sistemlerin modellenmesi aşamasında eğitim veri setleri hazırlanmış ve kullanılmıştır. Eğitim seti kullanılarak aşırı öğrenme makinalarının çalıştırılması ile elde edilen sonuçlar tablolar halinde gösterilmiştir.

Çizelge 4. 1. Meta-AÖM öğrenmeli dinamik sistem modelleme için literatürden seçilen dinamik sistemler (DS).

Dinamik Sistemler	
1	$y(k) = \frac{y(k-1) \cdot y(k-2) \cdot (y(k-1) + 2.5)}{1 + y^2(k-1) + y^2(k-2)} + u(k),$ <p>(Narendra ve Parthasarathy, 1990)</p>
2	$y(k+1) = \frac{y(k)}{1 + y^2} + u^3(k),$ <p>(Narendra ve Parthasarathy, 1990)</p>
3	$y(k+1) = y(k) + u(k)e^{-3 y(k) },$ <p>(Babuska, 2012)</p>
4	$y(k+1) = \frac{24 + y(k)}{30} y(k) - 0.8 \frac{u(k)^2}{1 + u(k)^2} y(k-1) + 0.5u(k),$ <p>(Oussar, Rivals ve Dreyfus, 1998)</p>
5	$y(k+1) = 0.5 * \left(\frac{y(k)}{1 + y^2(k)} + (1 + u(k))u(k)(1 - u(k)) \right),$ <p>(Sastry ,Santharam ve Unnikrishnan, 1994)</p>
6	$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)[y(k-2)-1] + u(k)}{1 + y(k-1)^2 + y(k-2)^2},$ <p>(Chia-Feng Juang, 2002)</p>
7	$y(k+1) = 0.3 y(k) + 0.6 y(k-1) + f(k)$ $f(k) = 0.6 \sin(\pi u(k)) + 0.3 \sin(3 \pi u(k)) + 0.1 \sin(5 \pi u(k)),$ <p>(Jang, 1993)</p>

4.2. Meta-AÖM ile Modelleme

Bu çalışmada, Meta-AÖM'nin dinamik sistem tanımlama/modelleme başarımı incelenmiştir. Bu amaçla literatürden yedi referans dinamik sistem (DS) seçilmiştir. DS'lerin tanımı Çizelge 4.1'de verilmiştir. Çizelge 4.1'de verilen her bir sistem için, harici kontrol sinyali ($u(k)$) ve $y(k)$ kullanılarak eğitim ve test veri setleri hazırlanmıştır. Meta-AÖM için hücre aktivasyon fonksiyonu olarak logaritmik sigmoid transfer fonksiyonu (logsig) kullanılmıştır.

Deneyel çalışmalarda Intel (R) Core (TM) i7-6700U CPU @ 3.40 GHz işlemcili, 8 GB RAM ve 64 bit Windows 10 işletim sistemine sahip bir masaüstü bilgisayar kullanılmıştır. Modelleme deneyleri her bir DS için 100'er kez farklı nöron sayısı (N_h) ve TKİBSA topluluğu sayısı (M) ile yapılmıştır. Deneyel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir.

4.2.1. Dinamik sistem modelleme örnek 1

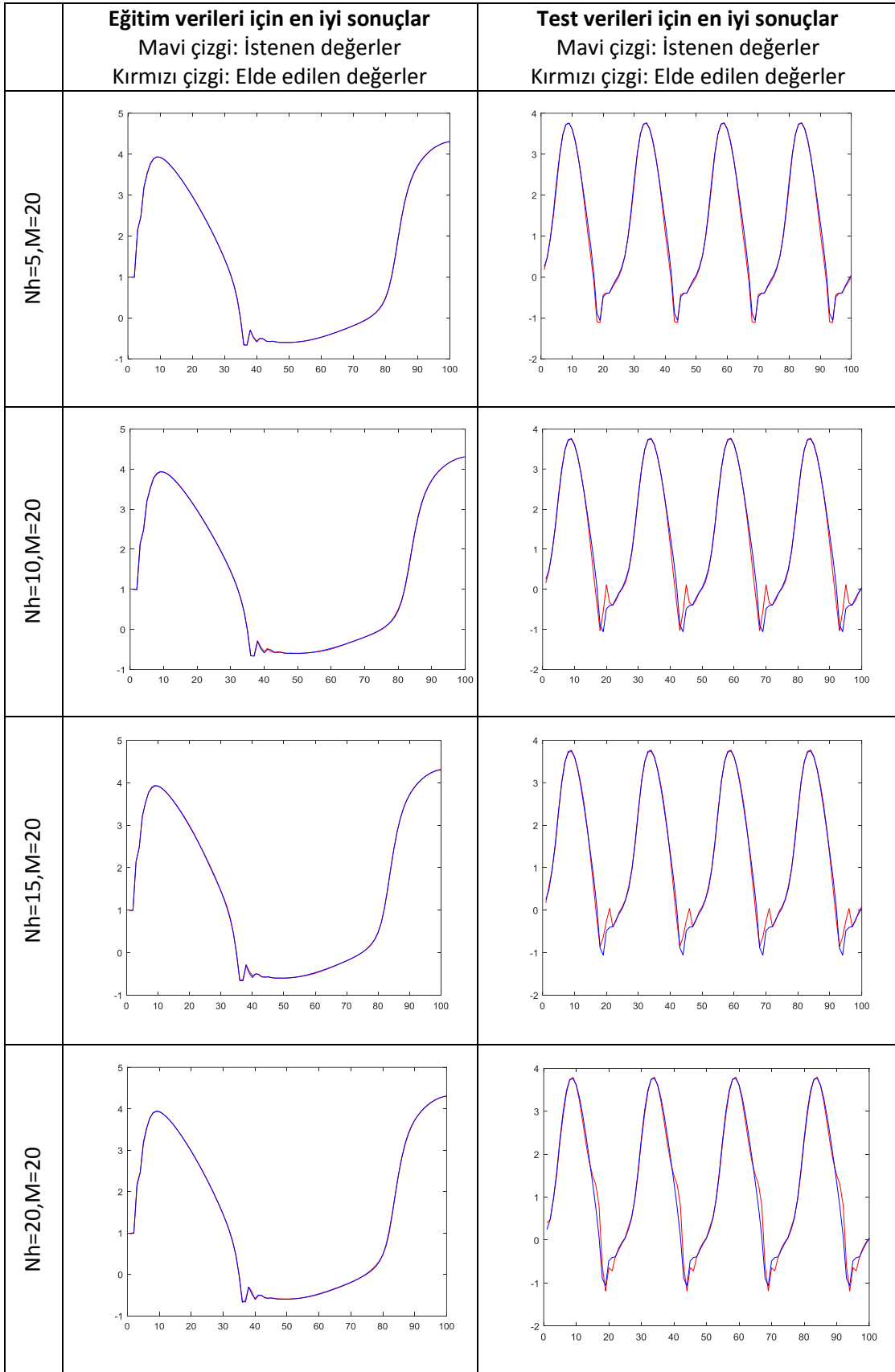
Dinamik sistemin matematik fonksiyonu eşitlik (4.1)'de (Narendra ve Parthasarathy, 1990) verilmiştir. Bu sistem için giriş olarak eşitlik (4.2) dizisi kullanarak eğitim verisi bulunmuştur. Bu sistem için elde edilen Meta-AÖM modelinin başarısını kontrol etmek için eşitlik (4.3)'teki giriş dizisi kullanılarak ve test verisi hazırlanmıştır.

$$y(k) = \frac{y(k-1) \cdot y(k-2) \cdot (y(k-1) + 2.5)}{1 + y^2(k-1) + y^2(k-2)} + u(k) \quad (4.1)$$

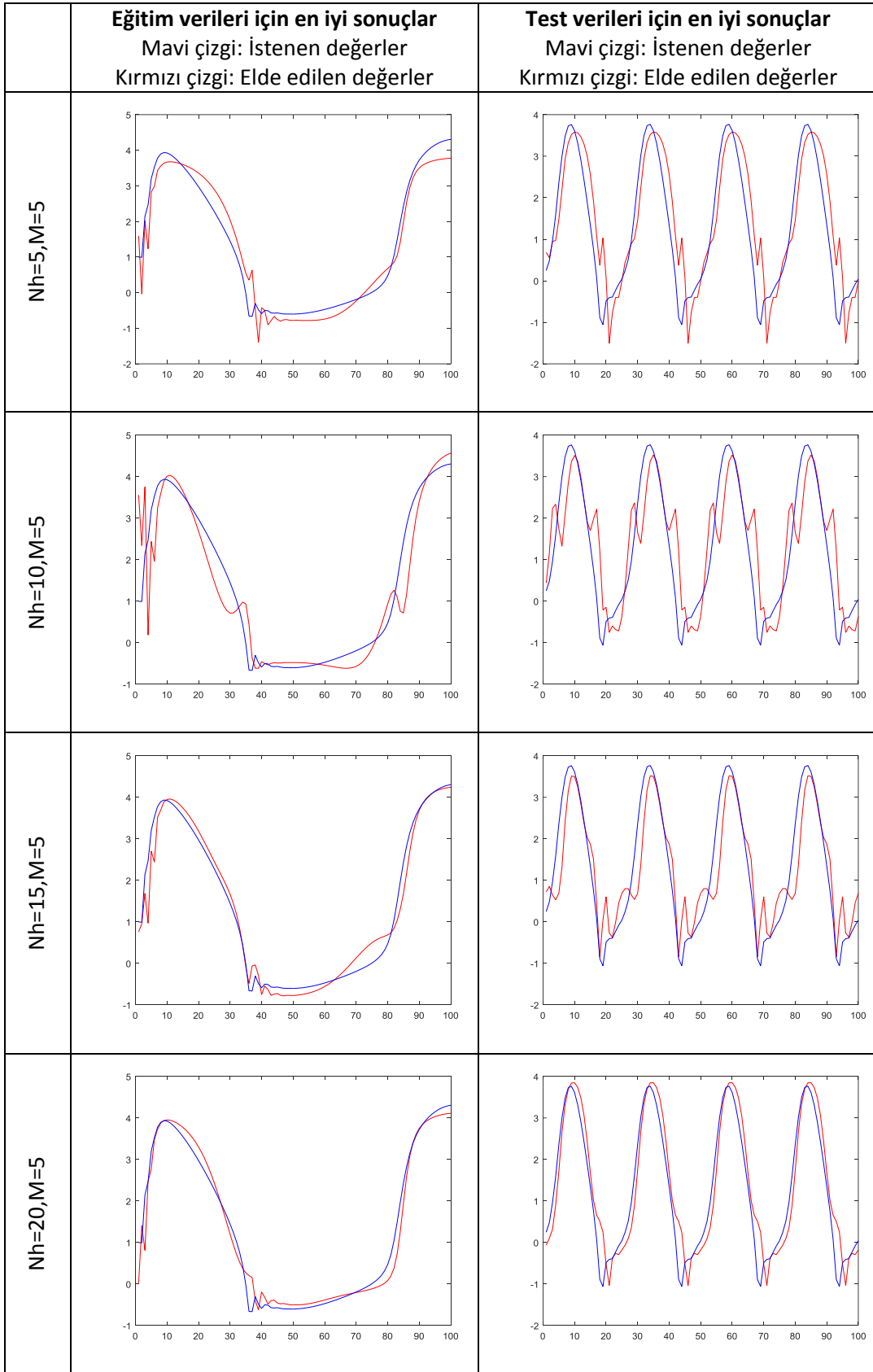
$$u(k)_{Train} = \cos \frac{2\pi k}{100} \quad (4.2)$$

$$u(k)_{Test} = \sin \frac{2\pi k}{25} \quad (4.3)$$

Deneyel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.1'de uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.2'de ise en kötü modelleme başarımı verilmiştir.



Şekil 4.1. DS #1 için Meta-AÖMnin en iyi modelleme başarımı.



Şekil 4.2. DS #1 için Meta-AÖMnin en kötü modelleme başarımı.

Çizelge 4. 2. DS #1 için elde edilen deneysel modelleme başarımları metrikleri

Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,001018	0,000705	0,018095	0,001751	0,000967	0,000925	0,001310	0,000067
Eğitim RMSE	0,220127	0,074289	0,472718	0,081772	0,401460	0,098889	1,450401	0,274875
Test RMSE	0,366589	0,159197	0,856230	0,145503	0,701723	0,184996	2,801200	0,494106
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,001315	0,001214	0,001716	0,000121	0,001524	0,001483	0,001670	0,000044
Eğitim RMSE	0,099666	0,036861	0,272968	0,034883	0,096355	0,034730	0,221095	0,034510
Test RMSE	0,228831	0,067598	0,522435	0,089811	0,209967	0,083273	0,441277	0,073458
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,001817	0,001712	0,002532	0,000167	0,002189	0,002118	0,002557	0,000081
Eğitim RMSE	0,033791	0,013181	0,076362	0,009861	0,032619	0,016375	0,061473	0,009160
Test RMSE	0,171803	0,065060	0,371857	0,060073	0,156927	0,067868	0,299903	0,047225
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,002589	0,002346	0,003074	0,000263	0,003236	0,002804	0,003773	0,000374
Eğitim RMSE	0,015557	0,006962	0,034033	0,004908	0,013516	0,006846	0,026215	0,004126
Test RMSE	0,200578	0,062555	0,686896	0,074198	0,177293	0,048552	0,575966	0,077434
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,000860	0,000816	0,001288	0,000062	0,001275	0,001152	0,001626	0,000144
Eğitim RMSE	0,353000	0,107429	1,570693	0,249412	0,409743	0,135703	1,835449	0,239069
Test RMSE	0,574036	0,155989	2,448897	0,388980	0,671873	0,171709	1,855356	0,320770
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,001394	0,001360	0,001529	0,000034	0,001696	0,001622	0,001981	0,000077
Eğitim RMSE	0,110037	0,047548	0,411597	0,050044	0,089056	0,037025	0,174033	0,028218
Test RMSE	0,248600	0,102319	0,613834	0,095595	0,209961	0,092389	0,492041	0,067262
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,001960	0,001899	0,002536	0,000096	0,002298	0,002238	0,002983	0,000106
Eğitim RMSE	0,033318	0,020191	0,096606	0,010910	0,033692	0,012470	0,080227	0,010414
Test RMSE	0,161263	0,079474	0,401602	0,050756	0,162128	0,069793	0,317185	0,057102
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,002975	0,002589	0,003709	0,000348	0,003327	0,003002	0,004498	0,000380
Eğitim RMSE	0,014173	0,006017	0,030261	0,004419	0,014483	0,007272	0,028243	0,004566
Test RMSE	0,186068	0,070335	0,366933	0,060100	0,184302	0,068162	0,364926	0,070910

4.2.2. Dinamik sistem modelleme örnek 2

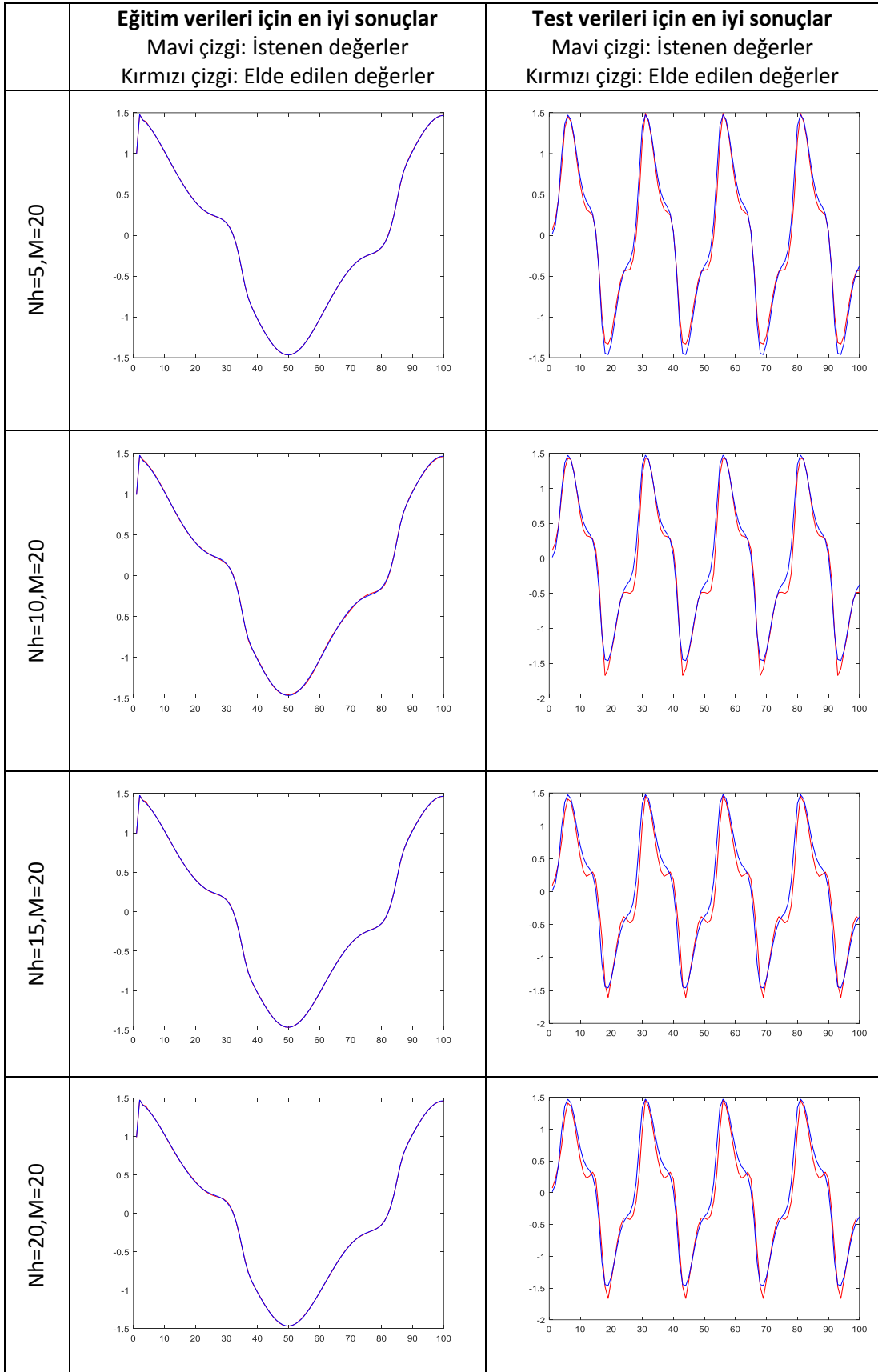
Dinamik sistemin matematik fonksiyonu eşitlik (4.4)'de (Narendra, 1990) verilmiştir. Bu sistem için giriş olarak eşitlik (4.5) dizisi kullanarak eğitim verisi bulunmuştur. Bu sistemin başarısını kontrol etmek için eşitlik (4.6)'daki giriş dizisi kullanıldı ve test verisi bulundu.

$$y(k+1) = \frac{y(k)}{1+y^2} + u^3(k) \quad (4.4)$$

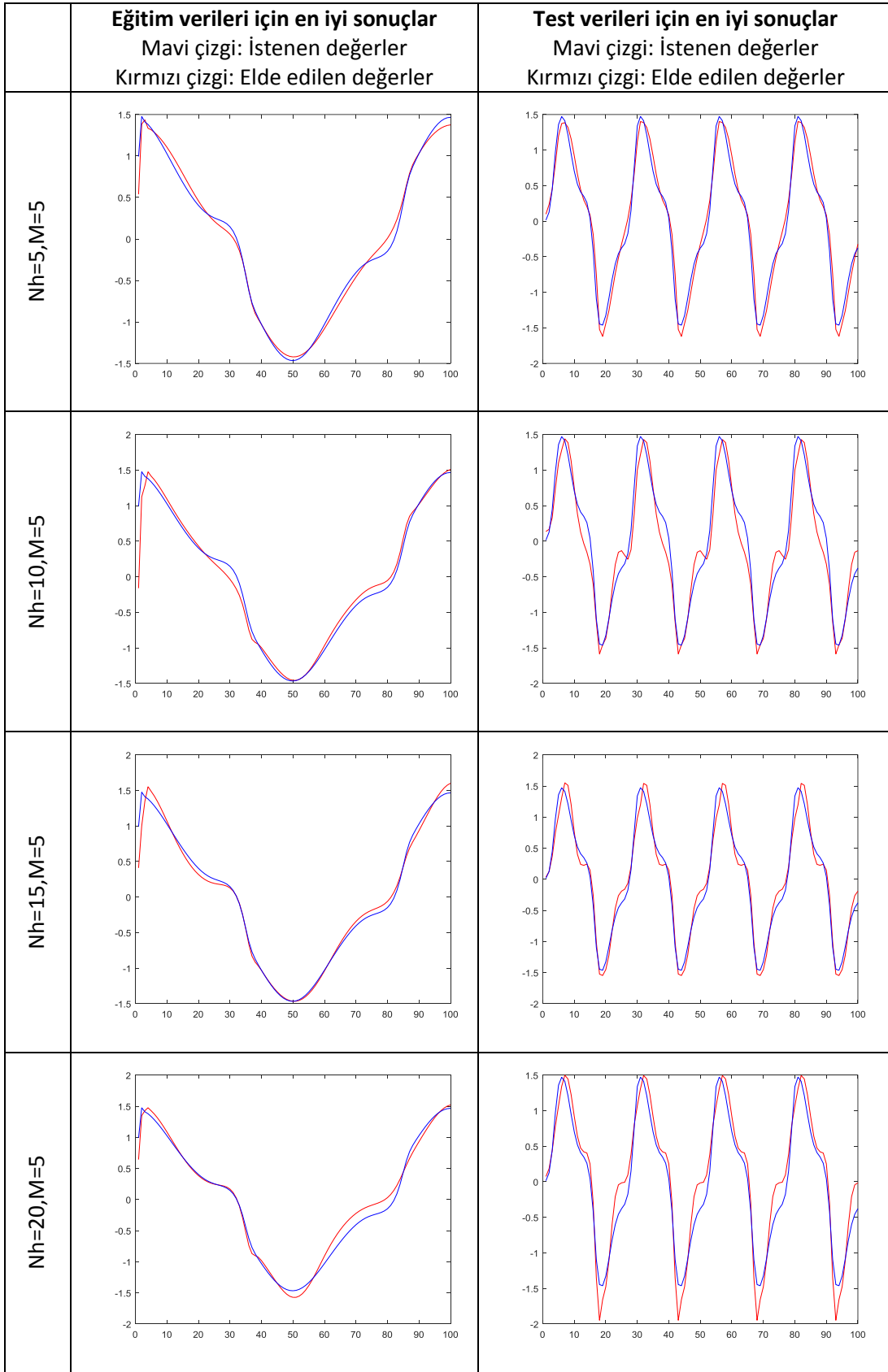
$$u(k)_{Train} = \cos \frac{2\pi k}{100} \quad (4.5)$$

$$u(k)_{Test} = \sin \frac{2\pi k}{25} \quad (4.6)$$

Deneysel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.3'te uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.4'te ise en kötü modelleme başarımı verilmiştir.



Şekil 4.3. DS #2 için Meta-AÖMnin en iyi modelleme başarımı.



Şekil 4.4. DS #2 için Meta-AÖMnin en kötü modelleme başarımı.

Çizelge 4. 3. DS #2 için elde edilen deneysel modelleme başarımları metrikleri.

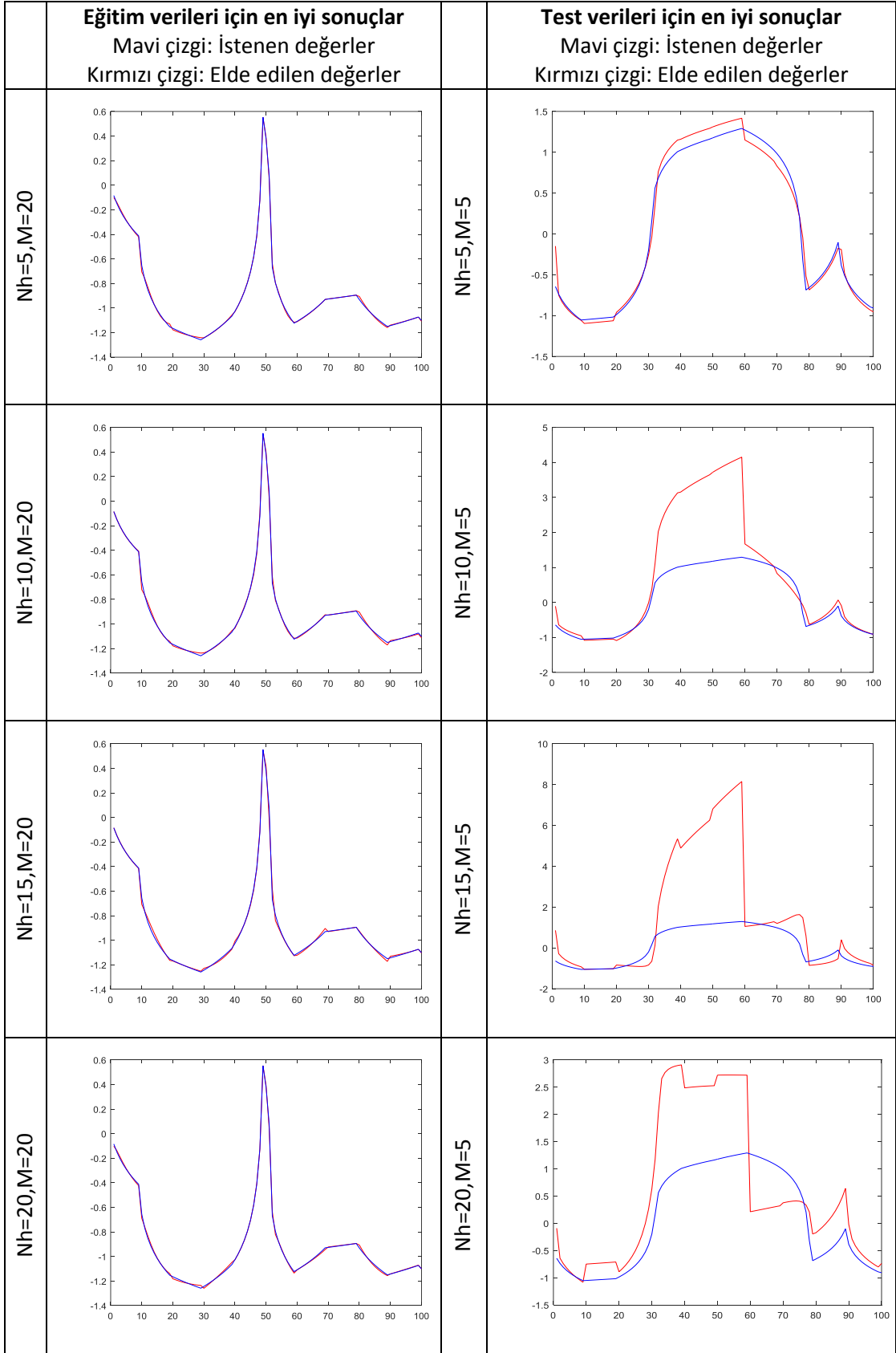
Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,001022	0,000706	0,018980	0,001852	0,000969	0,000930	0,001261	0,000061
Eğitim RMSE	0,133922	0,069575	0,311168	0,044920	0,156069	0,059960	0,491606	0,063849
Test RMSE	0,216319	0,099461	0,685672	0,094808	0,316319	0,129782	0,784363	0,116223
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,001268	0,001223	0,001798	0,000073	0,001551	0,001495	0,002428	0,000105
Eğitim RMSE	0,063764	0,031158	0,095461	0,014264	0,061667	0,023881	0,096121	0,013989
Test RMSE	0,182821	0,110481	0,307254	0,037599	0,175453	0,095805	0,354558	0,041248
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,001795	0,001716	0,002172	0,000076	0,002173	0,002117	0,002403	0,000061
Eğitim RMSE	0,022933	0,006791	0,052031	0,009368	0,026137	0,006864	0,057707	0,009600
Test RMSE	0,204191	0,078859	0,542297	0,092015	0,199582	0,091576	0,535820	0,071640
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,002715	0,002359	0,003125	0,000282	0,003173	0,002819	0,003765	0,000348
Eğitim RMSE	0,005954	0,001800	0,018231	0,002460	0,006183	0,001671	0,034381	0,004102
Test RMSE	0,212375	0,061933	0,803680	0,110059	0,205705	0,065981	1,164074	0,126786
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,000852	0,000820	0,001090	0,000049	0,001287	0,001157	0,002218	0,000166
Eğitim RMSE	0,120468	0,071553	0,246420	0,035133	0,148183	0,057821	0,450158	0,065008
Test RMSE	0,236091	0,120453	0,589996	0,079213	0,321946	0,125536	1,007057	0,138537
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,001408	0,001367	0,001886	0,000074	0,001669	0,001616	0,001956	0,000071
Eğitim RMSE	0,062039	0,027949	0,168908	0,019920	0,066069	0,026355	0,095189	0,013151
Test RMSE	0,194954	0,090189	0,433238	0,057325	0,179686	0,106348	0,273525	0,034232
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,001947	0,001895	0,002097	0,000053	0,002292	0,002230	0,002698	0,000079
Eğitim RMSE	0,025715	0,007445	0,058363	0,010889	0,025468	0,005577	0,066076	0,010954
Test RMSE	0,210133	0,091639	0,481124	0,075405	0,201493	0,086891	0,753156	0,083185
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,002811	0,002602	0,003456	0,000256	0,003429	0,003007	0,004164	0,000402
Eğitim RMSE	0,005475	0,002215	0,016405	0,002725	0,005818	0,001485	0,017500	0,002859
Test RMSE	0,193766	0,052842	0,543167	0,096584	0,184684	0,065425	0,541086	0,091781

4.2.3. Dinamik sistem modelleme örnek 3

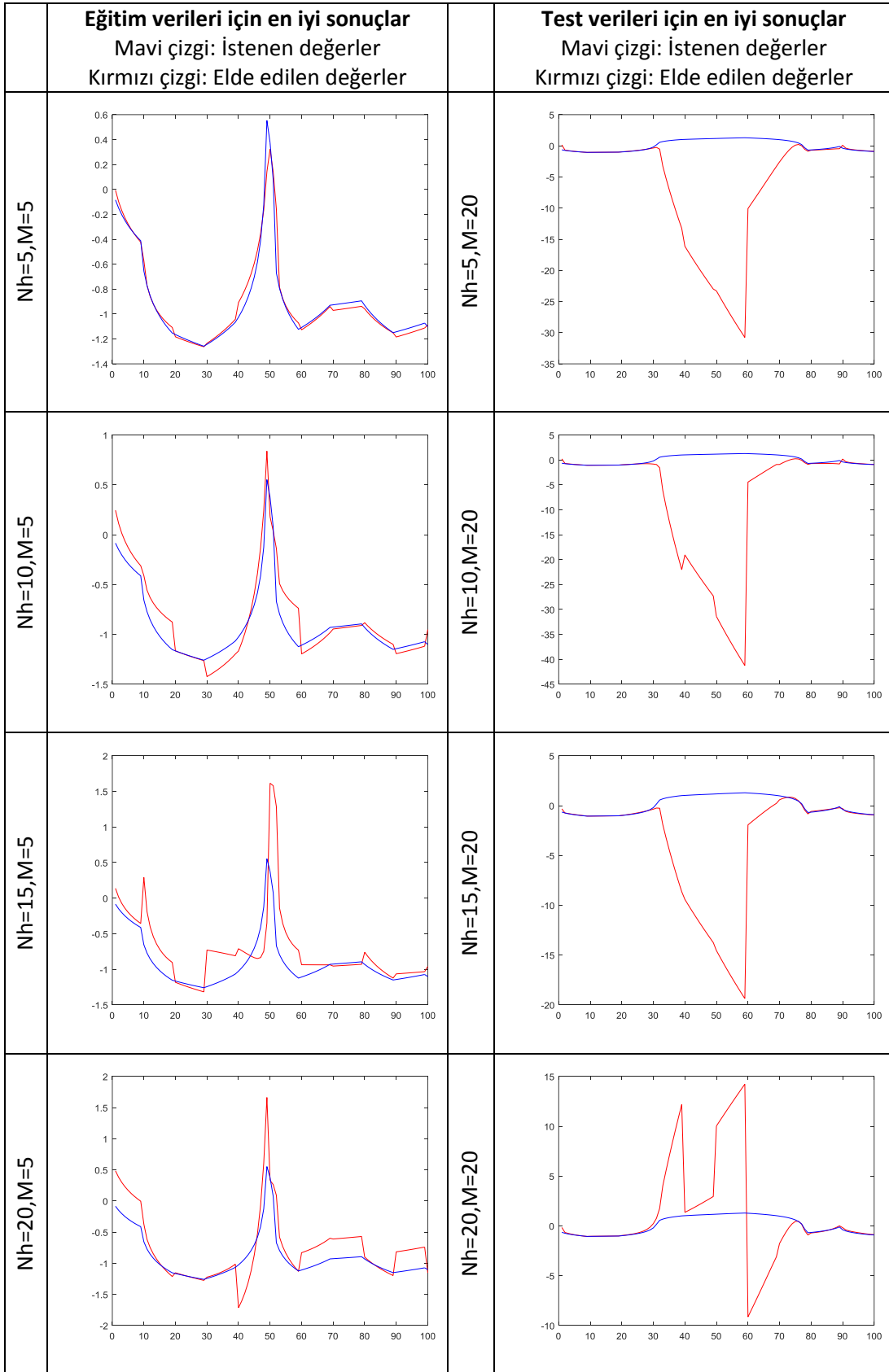
Dinamik sistemin tanımı eşitlik (4.7) de (Babuska,2012) verilmiştir. Bu sistem eğitimi ve testinde kullanmak üzere $[1 -1]$ aralığında rasgele genlikli darbelerden oluşan $u(k)$ girişleri ile eğitim ve test verisi hazırlanmıştır. Sistemde $u(k)$ ve $y(k)$ girişler olarak alınmıştır.

$$y(k+1) = y(k) + u(k)e^{-3|y(k)|} \quad (4.7)$$

Deneysel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.5'te uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.6'da ise en kötü modelleme başarımı verilmiştir.



Şekil 4.5. DS #3 için Meta-AÖMnin en iyi modelleme başarımı.



Şekil 4.6. DS #3 için Meta-AÖMnin en kötü modelleme başarımı.

Çizelge 4. 4. DS #3 için elde edilen deneysel modelleme başarımları metrikleri.

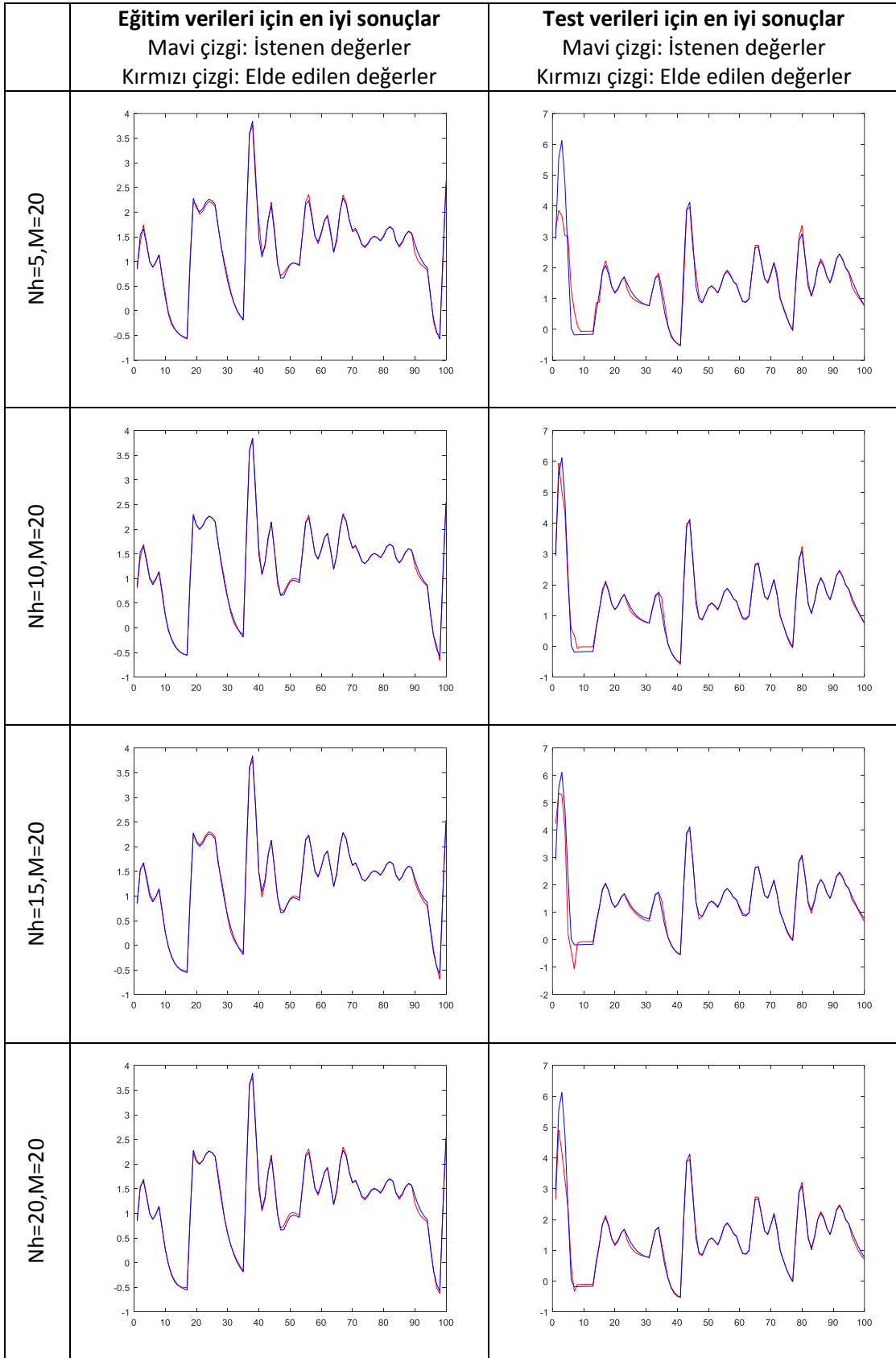
Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,000884	0,000695	0,013604	0,001321	0,000941	0,000892	0,001254	0,000057
Eğitim RMSE	0,067962	0,041848	0,097295	0,009480	0,290756	0,075980	0,743972	0,144729
Test RMSE	0,519813	0,098374	1,649616	0,323644	1,437971	0,369642	3,758256	0,716199
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,001253	0,001199	0,001823	0,000089	0,001504	0,001441	0,002012	0,000107
Eğitim RMSE	0,039146	0,027502	0,055852	0,006648	0,041461	0,030605	0,059387	0,007214
Test RMSE	1,818637	0,278395	4,590024	0,759569	1,750351	0,174211	4,133574	0,949982
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,001753	0,001689	0,002199	0,000097	0,002114	0,002043	0,002755	0,000111
Eğitim RMSE	0,023806	0,017987	0,030595	0,003153	0,022893	0,018014	0,030150	0,002546
Test RMSE	2,562533	0,347500	8,286780	1,766510	2,731629	0,434617	10,553923	1,934874
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,002598	0,002348	0,003285	0,000276	0,003137	0,002733	0,004028	0,000376
Eğitim RMSE	0,014244	0,009212	0,020002	0,002498	0,013338	0,008591	0,019477	0,002037
Test RMSE	8,443120	0,592333	30,996307	5,637400	7,612533	1,301427	18,057005	3,710338
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,000831	0,000794	0,001082	0,000048	0,001261	0,001143	0,001622	0,000128
Eğitim RMSE	0,126204	0,036200	0,641739	0,096296	0,297993	0,069514	0,737561	0,140916
Test RMSE	1,208510	0,144742	3,730760	0,689936	1,730912	0,279546	4,771748	0,838850
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,001373	0,001320	0,001566	0,000062	0,001640	0,001570	0,002288	0,000113
Eğitim RMSE	0,043126	0,028085	0,057961	0,006804	0,039632	0,025283	0,058427	0,007799
Test RMSE	1,701397	0,136417	4,252583	0,924931	1,916745	0,163656	3,704923	0,922225
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,002009	0,001881	0,002850	0,000170	0,002234	0,002142	0,003742	0,000214
Eğitim RMSE	0,022879	0,017896	0,030850	0,002888	0,023018	0,016272	0,030906	0,003202
Test RMSE	2,484913	0,472668	7,142909	1,700324	2,768515	0,353085	8,389081	1,756828
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,002875	0,002539	0,003402	0,000308	0,003211	0,002882	0,004150	0,000375
Eğitim RMSE	0,012846	0,007460	0,018103	0,001903	0,012670	0,010333	0,020248	0,001734
Test RMSE	8,723793	1,110658	37,876353	5,204231	7,738984	0,764787	25,734793	3,848416

4.2.4. Dinamik sistem modelleme örnek 4

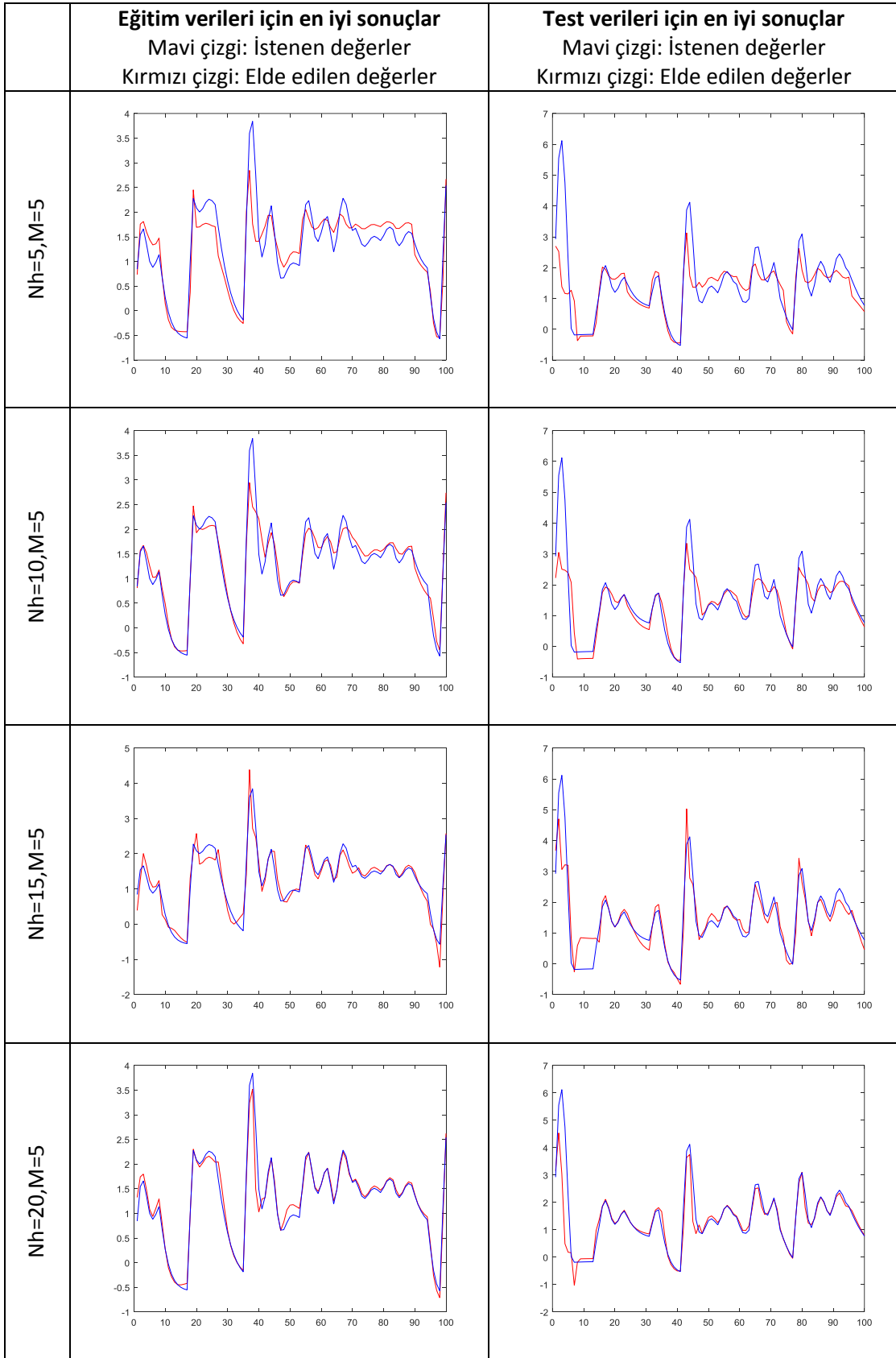
Dinamik sistemin matematiksel tanımı (4.8) de (Oussar, Rivals ve Dreyfus, 1998) verilmiştir. Bu sistem $[-5 \ 5]$ aralığında sürekli darbelerden oluşan $u(k)$ girişleri ile eğitim ve test verisi olarak tasarlanmıştır. Sistem 10 örnekleme periyodu ile çalışacak şekilde hazırlanmıştır. Sistemde $u(k)$, $y(k)$ ve $y(-1)$ girişler olarak alınmıştır. Bu sistemde 32 parametre eğitim verisi için belirlenmiştir.

$$y(k+1) = \frac{24 + y(k)}{30} y(k) - 0.8 \frac{u(k)^2}{1 + u(k)^2} y(k-1) + 0.5u(k) \quad (4.8)$$

Deneysel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.7’de uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.8’de ise en kötü modelleme başarımı verilmiştir.



Şekil 4.7. DS #4 için Meta-AÖMnin en iyi modelleme başarımı.



Őekil 4.8. DS #4 iin Meta-AÖMnin en kt modelleme baŐarımı.

Çizelge 4. 5. DS #4 için elde edilen deneysel modelleme başarımları metrikleri.

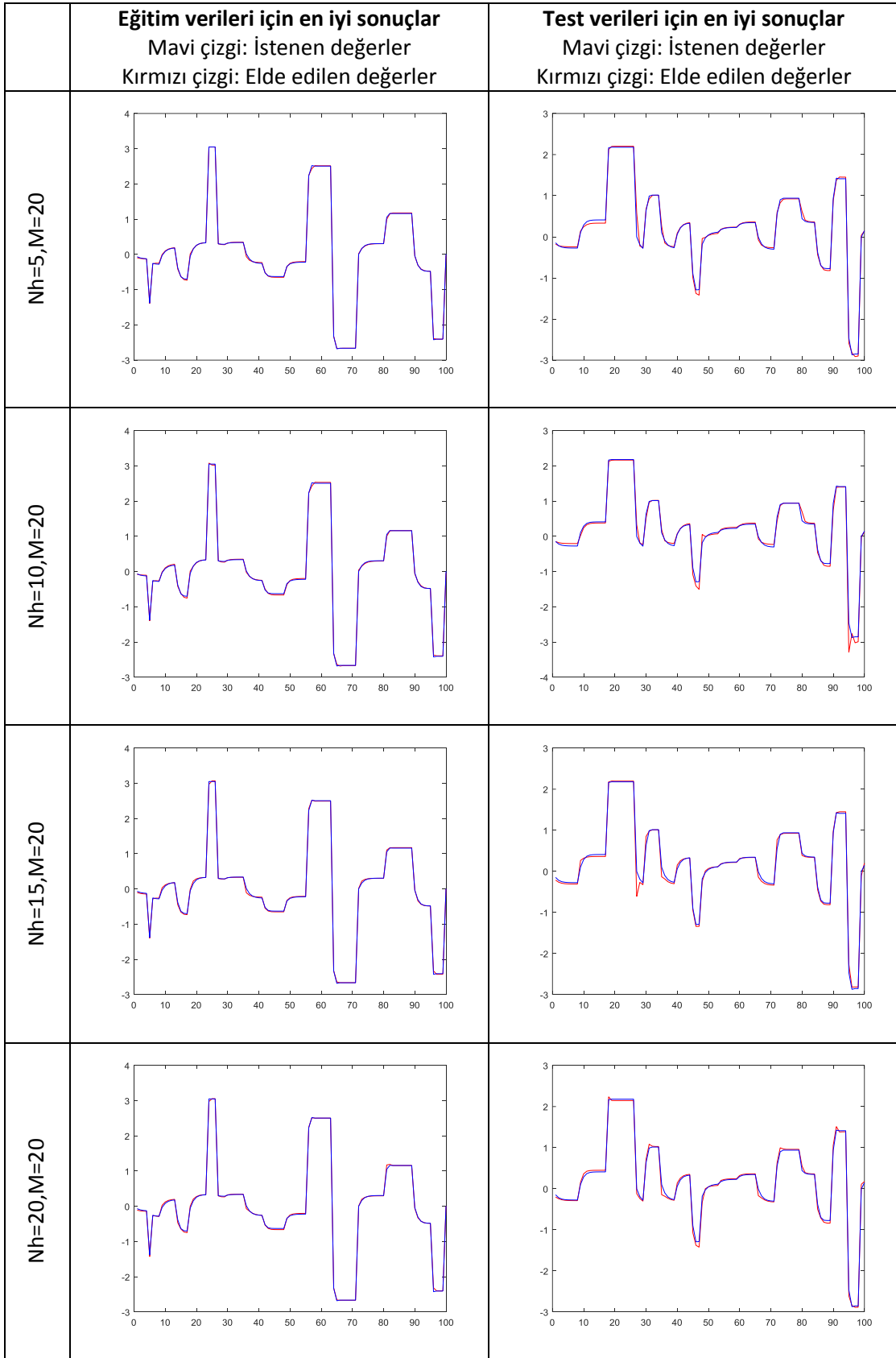
Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,000936	0,000703	0,013512	0,001308	0,001024	0,000970	0,001448	0,000100
Eğitim RMSE	0,305993	0,191441	0,449440	0,054865	0,233274	0,092497	0,543708	0,090715
Test RMSE	0,687427	0,411573	0,888943	0,086695	0,587554	0,264178	1,250204	0,178756
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,001275	0,001230	0,001462	0,000055	0,001574	0,001525	0,001796	0,000057
Eğitim RMSE	0,205350	0,084383	0,327459	0,045946	0,154631	0,078377	0,276040	0,036276
Test RMSE	0,543834	0,209183	0,745908	0,094786	0,444598	0,182506	0,747766	0,093754
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,001849	0,001740	0,002407	0,000129	0,002272	0,002170	0,002677	0,000111
Eğitim RMSE	0,113309	0,054894	0,171242	0,024654	0,092997	0,039634	0,155761	0,029351
Test RMSE	0,412592	0,198210	0,634192	0,091286	0,354748	0,150426	0,613190	0,087514
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,002815	0,002423	0,003411	0,000292	0,003272	0,002883	0,004229	0,000416
Eğitim RMSE	0,073010	0,029807	0,145359	0,022360	0,052918	0,024979	0,094247	0,013835
Test RMSE	0,382015	0,186109	0,660512	0,103259	0,306252	0,121180	0,587075	0,090419
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,000902	0,000841	0,001431	0,000096	0,001319	0,001209	0,001838	0,000127
Eğitim RMSE	0,242130	0,134256	0,503777	0,066717	0,352349	0,096537	1,161933	0,208393
Test RMSE	0,594301	0,311946	1,042961	0,139888	0,785263	0,283738	2,604200	0,403174
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,001510	0,001413	0,002182	0,000148	0,001805	0,001677	0,002300	0,000146
Eğitim RMSE	0,228079	0,089283	0,531815	0,066738	0,150111	0,071131	0,266842	0,038049
Test RMSE	0,577745	0,270122	1,112489	0,147097	0,425726	0,172980	0,658364	0,096981
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,002056	0,001936	0,002506	0,000133	0,002666	0,002296	0,005774	0,000652
Eğitim RMSE	0,102761	0,051754	0,163775	0,023913	0,083293	0,046595	0,138990	0,020866
Test RMSE	0,372099	0,169272	0,693810	0,092796	0,338589	0,163839	0,514499	0,073200
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,003010	0,002642	0,003593	0,000349	0,003768	0,003092	0,006013	0,000601
Eğitim RMSE	0,056679	0,027475	0,108767	0,016073	0,050661	0,031429	0,084717	0,010294
Test RMSE	0,310767	0,172461	0,566288	0,083873	0,292151	0,171797	0,526367	0,068541

4.2.5. Dinamik sistem modelleme örnek 5

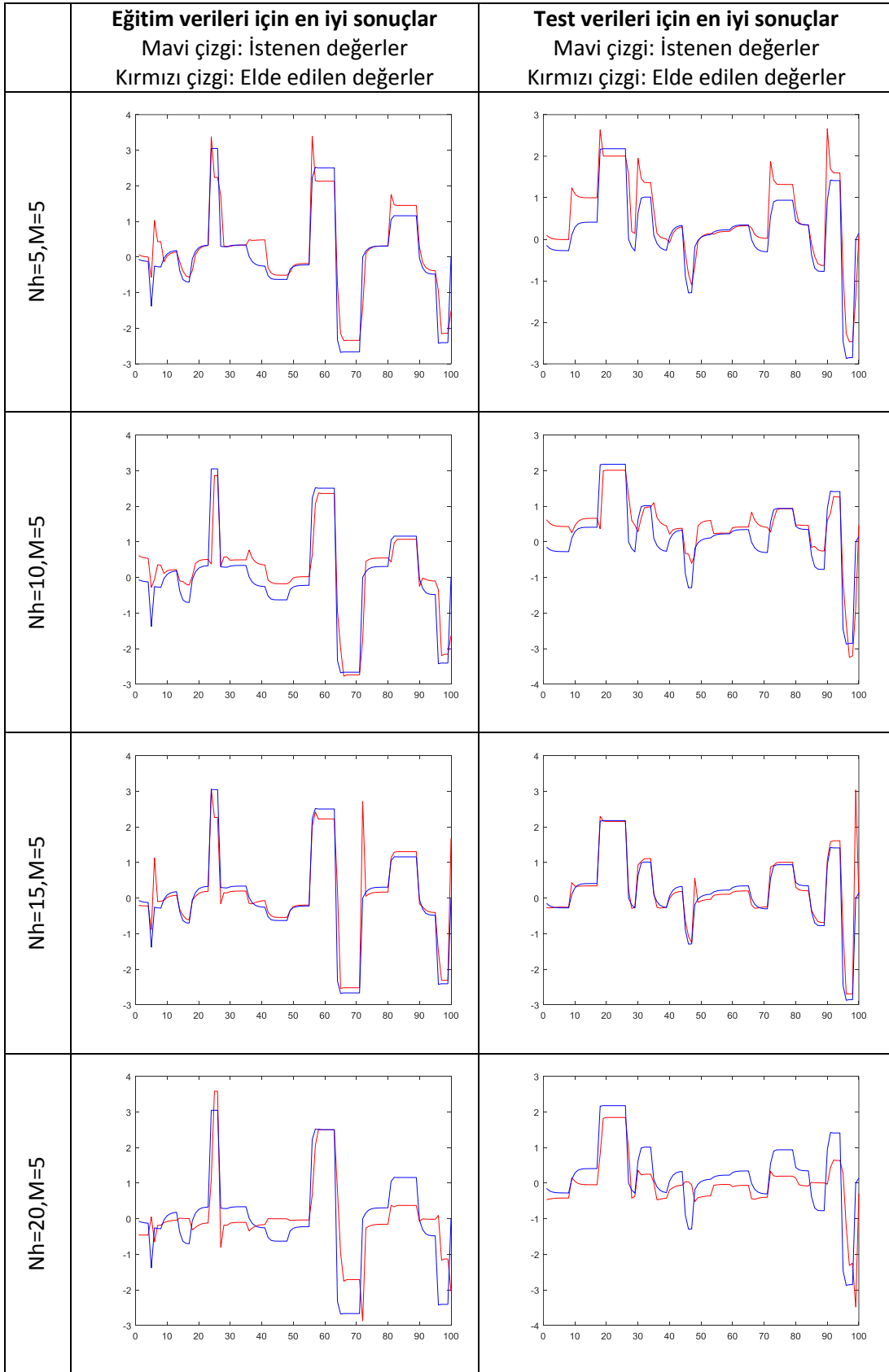
Dinamik sistemin tanımı (4.9)'da (Sastry, Santharam ve Unnikrishnan, 1994) verilmiştir. Bu sistem için $[-2 \ 2]$ aralığında sürekli darbelerden oluşan $u(k)$ girişleri ile eğitim ve test verisi hazırlanmıştır. Sistem rasgele örnekleme periyodu ile çalışacak şekilde hazırlanmıştır. Sistemde $u(k)$ ve $y(k)$ girişler olarak alınmıştır. Bu sistemde 12 parametre eğitim verisi kullanılarak belirlenmiştir.

$$y(k+1) = 0.5 * \left(\frac{y(k)}{1 + y^2(k)} + (1 + u(k))u(k)(1 - u(k)) \right) \quad (4.9)$$

DeneySEL sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.9'da uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.10'da ise en kötü modelleme başarımı verilmiştir.



Şekil 4.9. DS #5 için Meta-AÖMnin en iyi modelleme başarımı.



Şekil 4.10. DS #5 için Meta-AÖMnin en kötü modelleme başarımı.

Çizelge 4. 6. DS #5 için elde edilen deneysel modelleme başarımları metrikleri.

Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,001061	0,000736	0,016022	0,001548	0,000980	0,000929	0,001377	0,000085
Eğitim RMSE	0,483228	0,208824	0,665902	0,093124	0,598781	0,126808	1,083589	0,185085
Test RMSE	0,455064	0,202161	0,702861	0,092001	0,539965	0,103882	0,893280	0,172292
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,001289	0,001225	0,002330	0,000129	0,001574	0,001486	0,002242	0,000127
Eğitim RMSE	0,263500	0,067578	0,431618	0,087676	0,216236	0,069466	0,470845	0,078176
Test RMSE	0,285709	0,103335	0,627639	0,095981	0,234937	0,085733	0,453412	0,078406
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,001803	0,001758	0,002203	0,000073	0,002203	0,002112	0,003596	0,000167
Eğitim RMSE	0,075203	0,026013	0,203223	0,034753	0,079601	0,031329	0,187157	0,029238
Test RMSE	0,154689	0,046023	0,522901	0,083333	0,159532	0,054156	0,389752	0,064420
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,002622	0,002362	0,003373	0,000282	0,003328	0,002820	0,005027	0,000442
Eğitim RMSE	0,024322	0,008914	0,041839	0,007321	0,029710	0,014803	0,058105	0,008325
Test RMSE	0,093041	0,037319	0,234712	0,036633	0,099608	0,037602	0,284769	0,043572
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,000893	0,000828	0,001304	0,000079	0,001342	0,001167	0,001962	0,000165
Eğitim RMSE	0,516423	0,191609	0,881596	0,150658	0,789461	0,364142	1,153363	0,190609
Test RMSE	0,467931	0,158442	0,836771	0,147122	0,711443	0,273604	1,094920	0,158954
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,001434	0,001368	0,001885	0,000091	0,001727	0,001626	0,002747	0,000170
Eğitim RMSE	0,226368	0,080797	0,466078	0,091921	0,223374	0,085084	0,491134	0,091641
Test RMSE	0,254548	0,089005	0,511663	0,091062	0,236042	0,079271	0,519106	0,088165
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,002066	0,001907	0,003518	0,000227	0,002313	0,002249	0,002650	0,000074
Eğitim RMSE	0,081776	0,028010	0,182130	0,030932	0,091745	0,038945	0,269176	0,039936
Test RMSE	0,155014	0,056559	0,399493	0,069035	0,192540	0,071259	0,518282	0,092996
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,003034	0,002598	0,004067	0,000363	0,003446	0,002982	0,004180	0,000404
Eğitim RMSE	0,028104	0,007434	0,051678	0,008506	0,031095	0,013361	0,094740	0,011903
Test RMSE	0,098589	0,042592	0,226706	0,036165	0,113605	0,036006	0,330382	0,054311

4.2.6. Dinamik sistem modelleme örnek 6

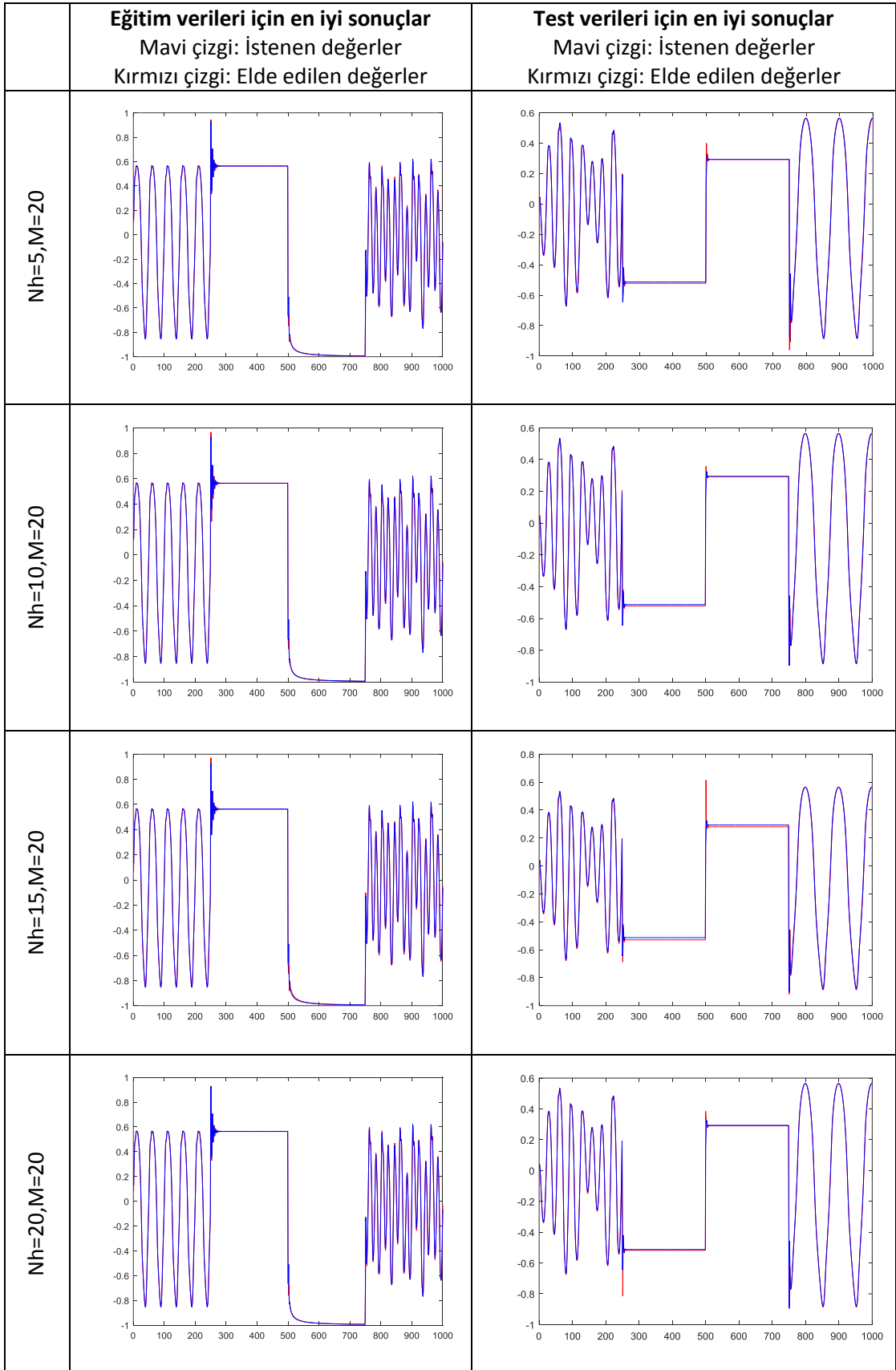
Dinamik sistemin matematik fonksiyonu eşitlik (4.10)'da (Chia-Feng Juang, 2002) verilmiştir. Bu sistem için giriş olarak eşitlik (4.11) dizisi kullanarak eğitim verisi bulunmuştur. Bu sistemin başarısını kontrol etmek için eşitlik (4.12)'teki giriş dizisi kullanıldı ve test verisi bulundu.

$$y(k+1) = \frac{y(k)y(k-1)y(k-2)u(k-1)[y(k-2)-1] + u(k)}{1 + y(k-1)^2 + y(k-2)^2} \quad (4.10)$$

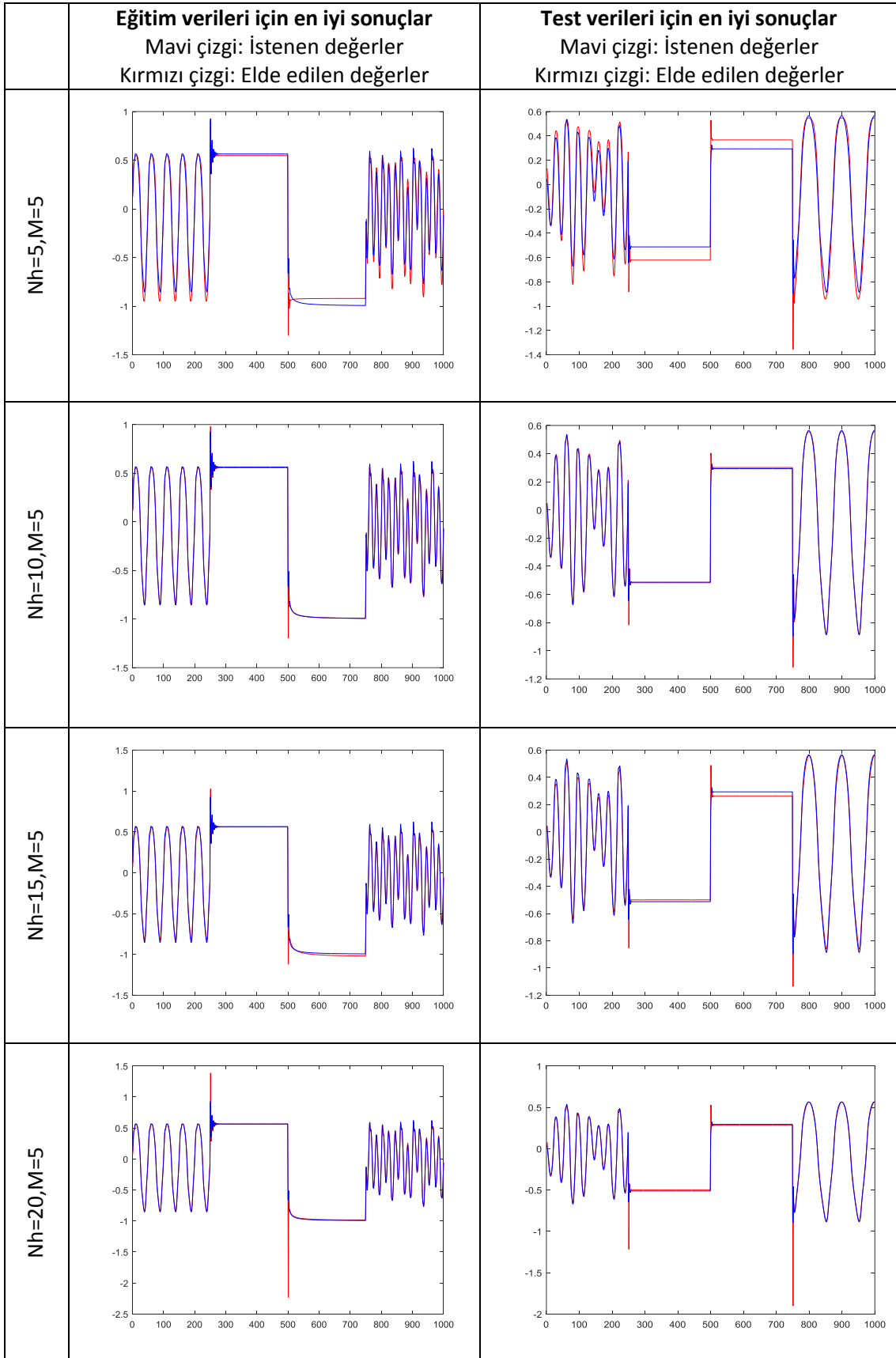
$$u(k)_{Train} = \begin{cases} \sin\left(\frac{\pi k}{25}\right) & \text{if } k < 250 \\ 1 & \text{if } 250 \leq k < 500 \\ -1 & \text{if } 500 \leq k < 750 \\ 0.3 \sin\left(\frac{\pi k}{25}\right) + 0.1 \sin\left(\frac{\pi k}{32}\right) + 0.6 \sin\left(\frac{\pi k}{10}\right) & \text{if } k \geq 750 \end{cases} \quad (4.11)$$

$$u(k)_{Test} = \begin{cases} -0.15 \cos\left(\frac{\pi k}{50}\right) + 0.5 \cos\left(\frac{\pi k}{16}\right) - 0.3 \cos\left(\frac{\pi k}{20}\right) & \text{if } k < 250 \\ -0.65 & \text{if } 250 \leq k < 500 \\ 0.35 & \text{if } 500 \leq k < 750 \\ \cos\left(\frac{\pi k}{50}\right) & \text{if } k \geq 750 \end{cases} \quad (4.12)$$

Deneysel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.11'de uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.12'de ise en kötü modelleme başarımı verilmiştir.



Şekil 4.11. DS #6 için Meta-AÖMnin en iyi modelleme başarımı.



Şekil 4.12. DS #6 için Meta-AÖMnin en kötü modelleme başarımı.

Çizelge 4. 7. DS #6 için elde edilen deneysel modelleme başarımları metrikleri

Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,005052	0,001542	0,306908	0,030499	0,003801	0,003002	0,012128	0,001409
Eğitim RMSE	0,044289	0,026851	0,068085	0,008111	0,034805	0,019935	0,058830	0,008538
Test RMSE	0,047799	0,027833	0,084787	0,010688	0,033549	0,021411	0,055729	0,006524
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,003292	0,002673	0,010272	0,001003	0,005498	0,004076	0,018109	0,001880
Eğitim RMSE	0,027418	0,019597	0,041748	0,004027	0,022836	0,013732	0,037382	0,004805
Test RMSE	0,028939	0,020049	0,051220	0,005482	0,027480	0,018614	0,042057	0,005093
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,004858	0,003838	0,031643	0,002732	0,008390	0,005864	0,025646	0,004031
Eğitim RMSE	0,017766	0,013740	0,022491	0,002225	0,017066	0,012018	0,025467	0,002745
Test RMSE	0,023465	0,018500	0,029551	0,002173	0,024399	0,016430	0,036060	0,003716
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,008340	0,005017	0,032880	0,004495	0,012225	0,007613	0,036505	0,007895
Eğitim RMSE	0,013440	0,011027	0,016763	0,001079	0,013275	0,011214	0,016742	0,001220
Test RMSE	0,022420	0,017878	0,030584	0,002676	0,022867	0,017795	0,032321	0,003040
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,002842	0,002030	0,009379	0,001491	0,005568	0,003792	0,017082	0,002594
Eğitim RMSE	0,031234	0,019033	0,061806	0,006624	0,040281	0,020955	0,080866	0,011143
Test RMSE	0,033153	0,022083	0,054465	0,006687	0,037468	0,021040	0,072321	0,009876
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,003887	0,003306	0,012820	0,001332	0,006510	0,005650	0,012988	0,001019
Eğitim RMSE	0,023787	0,015639	0,034051	0,003826	0,023009	0,014560	0,041932	0,005455
Test RMSE	0,027249	0,018982	0,040742	0,004219	0,030126	0,019597	0,085191	0,007822
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,006472	0,004896	0,020517	0,003668	0,008870	0,007745	0,011595	0,000592
Eğitim RMSE	0,017212	0,013920	0,024617	0,002139	0,017413	0,011291	0,033438	0,003848
Test RMSE	0,023749	0,018843	0,036666	0,002724	0,028824	0,019233	0,057496	0,006829
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,008720	0,006161	0,024779	0,003985	0,011211	0,009968	0,018520	0,001159
Eğitim RMSE	0,013385	0,011290	0,016413	0,001073	0,013067	0,010495	0,017291	0,001358
Test RMSE	0,022138	0,018801	0,029524	0,002062	0,025880	0,018882	0,059815	0,005648

4.2.7. Dinamik sistem modelleme örnek 7

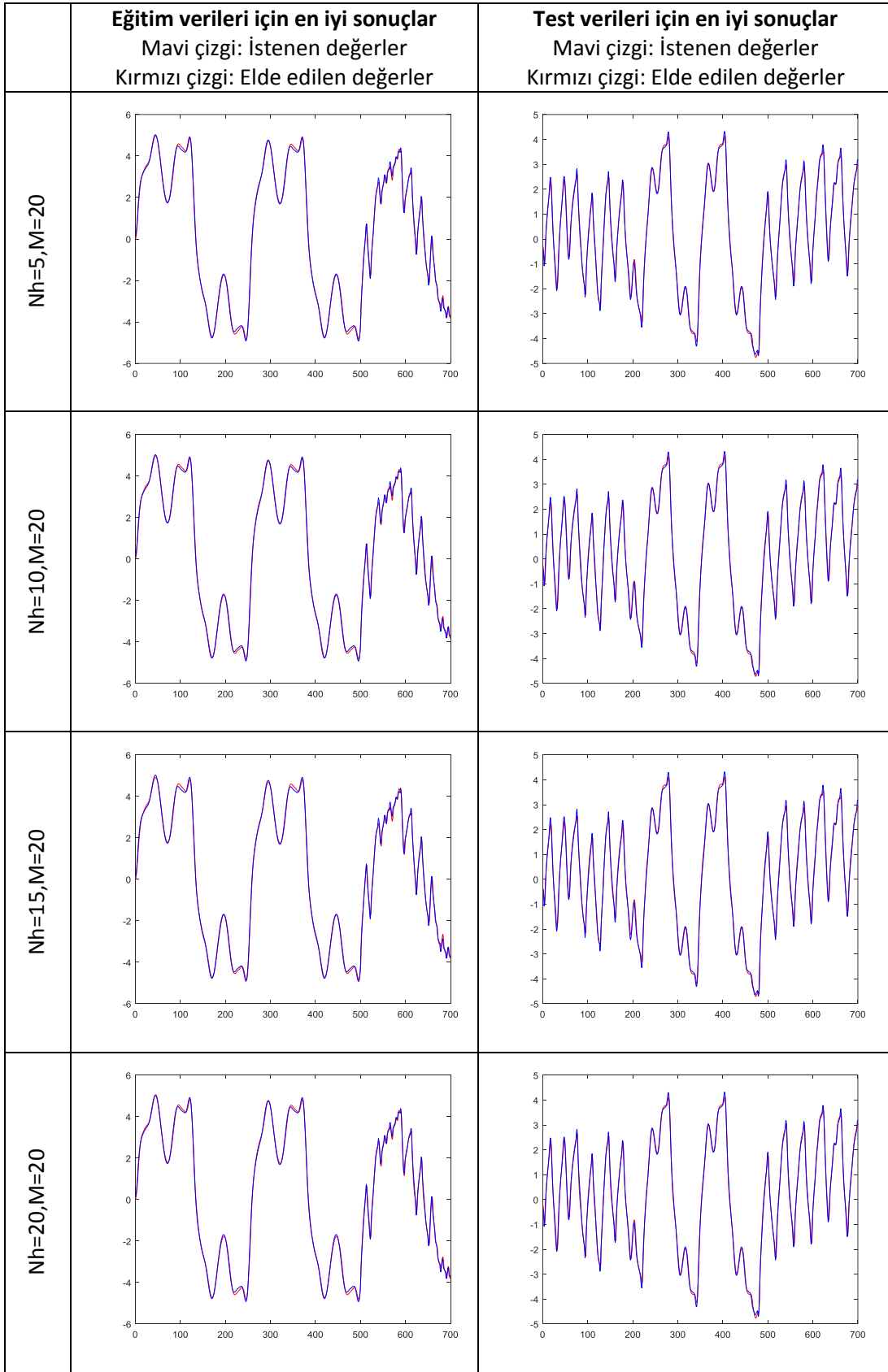
Dinamik sistemin matematik fonksiyonu eşitlik (4.13)'te (Jang 1993) verilmiştir. Bu sistem için giriş olarak eşitlik (4.14) dizisi kullanarak eğitim verisi bulunmuştur. Bu sistemin başarısını kontrol etmek için eşitlik (4.15)'teki giriş dizisi kullanıldı ve test verisi bulundu.

$$\begin{aligned} y(k+1) &= 0.3 y(k) + 0.6 y(k-1) + f(k) \\ f(k) &= 0.6 \sin(\pi u(k)) + 0.3 \sin(3\pi u(k)) + 0.1 \sin(5\pi u(k)) \end{aligned} \quad (4.13)$$

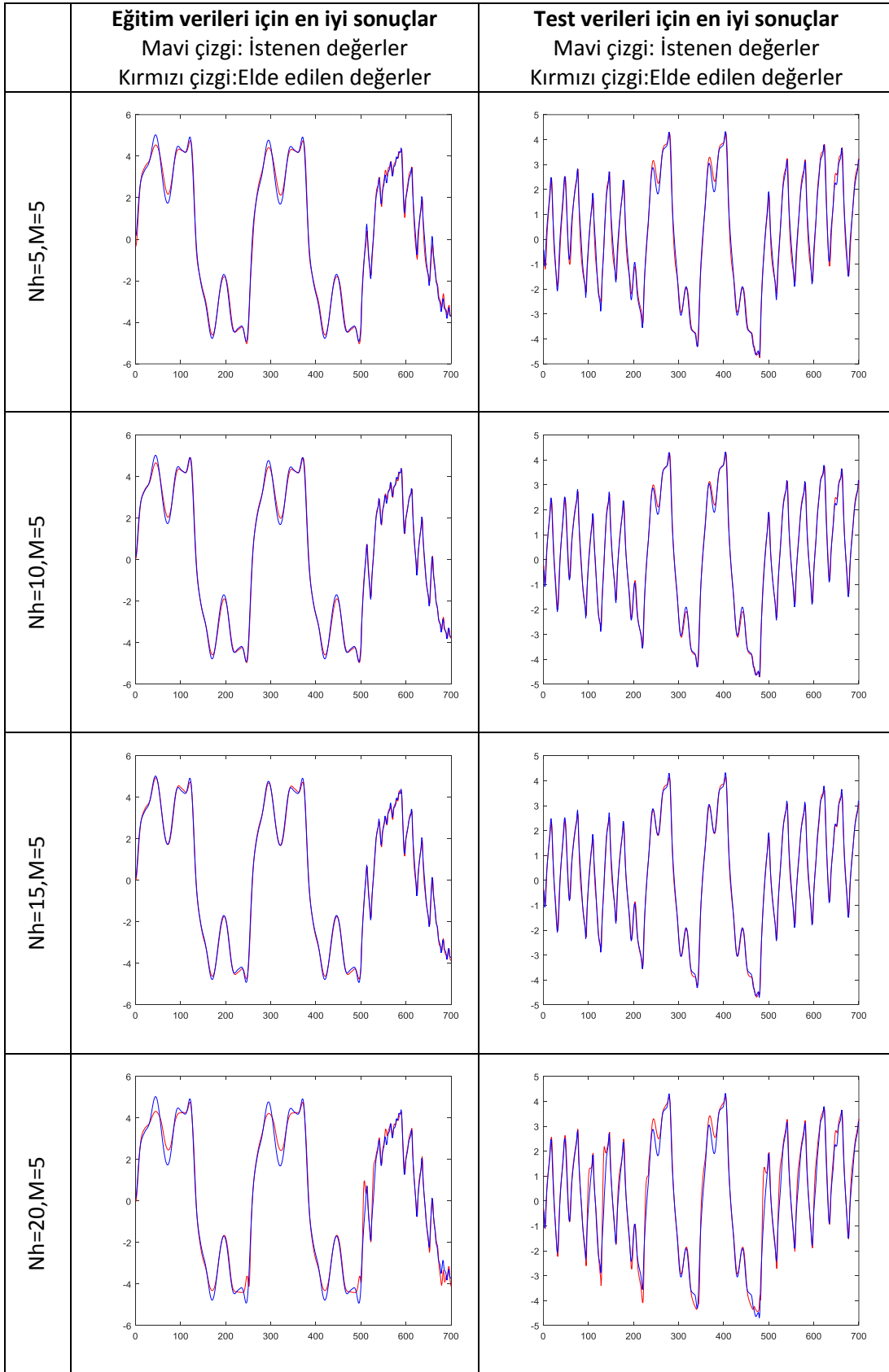
$$u(k)_{Train} = \begin{cases} \sin\left(\frac{2\pi k}{250}\right) & \text{if } k < 500 \\ 0.5 \sin\left(\frac{2\pi k}{250}\right) + 0.5 \sin\left(\frac{2\pi k}{25}\right) & \text{if } k \geq 500 \end{cases} \quad (4.14)$$

$$u(k)_{Test} = \begin{cases} -0.15 \cos\left(\frac{\pi k}{50}\right) + 0.5 \sin\left(\frac{\pi k}{16}\right) - 0.3 \cos\left(\frac{\pi k}{20}\right) & \text{if } k < 234 \\ \cos\left(\frac{2\pi k}{125}\right) & \text{if } 234 \leq k < 467 \\ 0.3 \sin\left(\frac{2\pi k}{250}\right) + 0.1 \cos\left(\frac{\pi k}{64}\right) + 0.6 \sin\left(\frac{\pi k}{20}\right) & \text{if } k \geq 467 \end{cases} \quad (4.15)$$

Deneysel sonuçlar hem eğitim süresi hem de modelleme başarımı açısından değerlendirilmiştir. Şekil 4.1'de uygulamada kullanılan algoritmaların koşturulması sonucunda eğitim ve test fazında elde edilen en iyi modelleme başarımı verilmiştir. Şekil 4.2'de ise en kötü modelleme başarımı verilmiştir.



Şekil 4.13. DS #7 için Meta-AÖMnin en iyi modelleme başarımı.



Şekil 4.14. DS #7 için Meta-AÖMnin en kötü modelleme başarımı.

Çizelge 4. 8. DS #7 için elde edilen deneysel modelleme başarımları metrikleri.

Metrik	Ortalama	En iyi	En kötü	Std. Sapma	Ortalama	En iyi	En kötü	Std. Sapma
	HÜCRE SAYISI=5 GRUP SAYISI=5				HÜCRE SAYISI=15 GRUP SAYISI=5			
Eğitim süresi [s]	0,002205	0,001312	0,042537	0,004119	0,002944	0,002470	0,004154	0,000333
Eğitim RMSE	0,239817	0,163219	0,440726	0,056404	0,151976	0,123140	0,268203	0,023405
Test RMSE	0,253489	0,174712	0,556574	0,064745	0,162815	0,133640	0,218907	0,017976
	HÜCRE SAYISI=5 GRUP SAYISI=10				HÜCRE SAYISI=15 GRUP SAYISI=10			
Eğitim süresi [s]	0,003091	0,002404	0,010592	0,001060	0,004237	0,003655	0,005455	0,000352
Eğitim RMSE	0,169071	0,128747	0,285412	0,023875	0,178206	0,127559	0,266601	0,027643
Test RMSE	0,183623	0,136775	0,359496	0,027732	0,208139	0,148810	0,314310	0,033318
	HÜCRE SAYISI=5 GRUP SAYISI=15				HÜCRE SAYISI=15 GRUP SAYISI=15			
Eğitim süresi [s]	0,004451	0,003622	0,032664	0,002876	0,005885	0,005088	0,008551	0,000565
Eğitim RMSE	0,129024	0,118187	0,176279	0,008727	0,140476	0,116352	0,191786	0,014162
Test RMSE	0,147144	0,132355	0,192209	0,010266	0,162458	0,130819	0,221718	0,019498
	HÜCRE SAYISI=5 GRUP SAYISI=20				HÜCRE SAYISI=15 GRUP SAYISI=20			
Eğitim süresi [s]	0,005640	0,004614	0,021718	0,001740	0,007497	0,006722	0,010080	0,000593
Eğitim RMSE	0,118272	0,112329	0,125346	0,002272	0,119792	0,114565	0,134494	0,003309
Test RMSE	0,137311	0,130981	0,147622	0,003509	0,138859	0,129793	0,153284	0,005150
	HÜCRE SAYISI=10 GRUP SAYISI=5				HÜCRE SAYISI=20 GRUP SAYISI=5			
Eğitim süresi [s]	0,001926	0,001575	0,002640	0,000238	0,003318	0,002960	0,004335	0,000268
Eğitim RMSE	0,179476	0,132063	0,309817	0,030839	0,200329	0,135773	0,356720	0,037922
Test RMSE	0,186666	0,139151	0,297969	0,025485	0,228337	0,152447	0,354152	0,042489
	HÜCRE SAYISI=10 GRUP SAYISI=10				HÜCRE SAYISI=20 GRUP SAYISI=10			
Eğitim süresi [s]	0,003351	0,002903	0,005042	0,000431	0,005223	0,004313	0,012353	0,001045
Eğitim RMSE	0,161752	0,126671	0,255254	0,026689	0,240327	0,148063	0,397300	0,054201
Test RMSE	0,178975	0,138315	0,294795	0,027562	0,288579	0,155605	0,606970	0,067089
	HÜCRE SAYISI=10 GRUP SAYISI=15				HÜCRE SAYISI=20 GRUP SAYISI=15			
Eğitim süresi [s]	0,004915	0,004083	0,007596	0,000639	0,007459	0,006379	0,009485	0,000712
Eğitim RMSE	0,130177	0,117737	0,172403	0,008717	0,155454	0,122553	0,251524	0,027663
Test RMSE	0,148600	0,133600	0,199831	0,010681	0,180907	0,135354	0,334517	0,038951
	HÜCRE SAYISI=10 GRUP SAYISI=20				HÜCRE SAYISI=20 GRUP SAYISI=20			
Eğitim süresi [s]	0,006266	0,005361	0,010102	0,000724	0,009883	0,007820	0,053147	0,005017
Eğitim RMSE	0,118389	0,113079	0,128038	0,002625	0,121520	0,112901	0,143833	0,004612
Test RMSE	0,137459	0,129802	0,150014	0,003829	0,141786	0,130066	0,183791	0,007877

5: SONUÇ ve ÖNERİLER

Bu tez çalışmasında, Meta-AÖM ağ mimarisi ve öğrenme algoritmasının dinamik sistemlerin modellenmesinde başarımı incelenmiştir. Bölüm 1'de verilen literatür özetinden de anlaşılacağı üzere, Meta-AÖMnin bu tez çalışması kapsamında yapılan çalışmaya benzer bir kullanımı olduğu söylenemez. Bu bakımdan bu çalışmanın ilgili alanda katkı sunduğu değerlendirilmektedir.

Bölüm 3'de verilen AÖM ve Meta AÖM ağ yapısına göre dinamik sistemler için gerekli yazılım araçları oluşturulmuş, sistemlerin hem eğitilmesinde hem de test edilmesinde farklı nöron ve TKİBSA sayısı parametre olarak alınarak mimari ve algoritmanın başarımı her bir dinamik sistem modellemesi için ayrı ayrı elde edilmiş ve detaylı bir şekilde incelenerek kıyaslama yapılmıştır.

Bu çalışmada seçilen DS tanımlama problemleri üzerinde yapılan çalışmalar sonucunda, Çizelge 4.2, Çizelge 4.3, Çizelge 4.4, Çizelge 4.5, Çizelge 4.6, Çizelge 4.7 ve Çizelge 4.8'ye dayanarak, Meta-AÖM ağ mimarisinin her zaman eğitim süresi açısından üstün başarımlar sağladığını söylemek mümkündür.

Çizelgelerde eğitim sürelerini dikkate alırsak hücre sayımız, yani nöron sayımız arttıkça eğitim süremizde de artma meydana gelmiştir. Hücre sayısı arttıkça genelleştirme başarımı artmakta ancak eğitim sürelerinde artış meydana gelmekteydi. Bu da daha fazla işlemci daha fazla bellek gibi çeşitli donanımlara ihtiyacını meydana getirmekte, böylece yüksek maliyet oluşmaktaydı. Bu AÖMnin temel problemlerinden birisiydi. Meta-AÖM algoritmasını kullanarak, hücre sayısını arttırmadan verilerimizi gruplara bölerek, hem genelleştirme başarımı yükselmekte, hem de eğitim süreleri yani maliyet artmamaktadır. Ayrıca çizelgelerde verilen Meta-AÖM ağ mimarisinin modelleme başarım sonuçlarına dayanarak, genel olarak ağ parametrelerinin (N_h ve M) sayısı arttıkça, hem eğitim hem de test veri seti için başarımının iyileştiği söylenebilir.

KAYNAKLAR

- Alcin, Omer & Sengur, Abdulkadir & Ince, M. (2015). İLERİ-GERİ TAKİP ALGORİTMASI TABANLI SEYREK AŞIRI ÖĞRENME MAKİNESİ. Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi. 30.
- Babuska R., "Fuzzy System, Modeling and Identification", <http://www.dcsc.tudelft.nl/~babuska/transp/fuzzmod.pdf> (son erişim 19 Şubat 2012).
- Chia-Feng Juang. 2002. "A TSK-Type Recurrent Fuzzy Network for Dynamic Systems Processing by Neural Network and Genetic Algorithms." IEEE Transactions on Fuzzy Systems 10(2):155–70.
- Ertugrul, Ö. (2016). Aşırı Öğrenme Makineleri ile Biyolojik Sinyallerin Gizli Kaynaklarına Ayırıştırılması (Blind Signal Separation in Biological by Extreme Learning Machines). DUMF Mühendislik Dergisi. 7. 41-50.
- Fernández, César & Salinas, Luis & Torres, Claudio. (2018). A meta extreme learning machine method for forecasting financial time series. Applied Intelligence. 49.
- G. Zhao, Z. Shen, C. Miao, R. Gay, (2008). Enhanced extreme learning machine with stacked generalization, in: Proceedings of the IEEE international joint conference on neural networks, pp. 1191–1198.
- Huang, Guang-Bin & Zhu, Qin-Yu & Siew, Chee. (2004). Extreme learning machine: A new learning scheme of feedforward neural networks. IEEE International Conference on Neural Networks - Conference Proceedings. 2. 985 - 990 vol.2.
- Huang, G., Siew, C., (2015). Extreme learning machine: RBF network case. Conference: Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th, Volume: 2
- Jang, Jyh-shing Roger. 1993. "ANFIS : Adap Tive-Ne Twork-Based Fuzzy Inference System." 23(3).
- Karakuzu, C., Bakırcı, A., (2019). Dynamic System Identification Based on an Ensemble of AÖMs. 2. *International Congress on Engineering and Architecture*. 22-24 Nisan, Marmaris.
- Kaynar, Oguz & Arslan, Halil & Görmez, Yasin & IŞIK, Yunus Emre. (2018). Makine Öğrenmesi ve Öznitelik Seçim Yöntemleriyle Saldırı Tespiti. Bilişim Teknolojileri Dergisi, 11, (2).
- Liao, S., & Feng, C., (2014). Meta-AÖM: AÖM with AÖM hidden nodes. *Neurocomputing*, 128, 81-87.
- M.V. Heeswijk, Y. Miche, T. Lindh-Knuutila, P. Hilbers, T. Honkela, E. Oja, A. Lendasse,(2009). Adaptive ensemble models of extreme learning machines for time series prediction, in: Proceedings of the 19th International Conference on Artificial Neural Networks: Part II, vol. 5769 of ICANN'09, Springer-Verlag, Berlin, Heidelberg, pp. 305–314.

- M.V. Heeswijk, Y. Miche, E. Oja, A. Lendasse,(2011).GPU-accelerated and parallelized ELM ensembles for large-scale regression, *Neurocomputing* 74 (16) 2430–2437.
- Narendra K. S., Parthasarathy K., “Identification and control of dynamical systems using neural networks”, *IEEE Transactions on Neural Networks*, vol. 1, no 1, pp. 4-27, 1990.
- Oussar Y., Rivals I., Dreyfus L., "Training wavelet networks for nonlinear dynamic input output modeling", *Neurocomputing*, cilt 20, syf 173-188, 1998.
- Özçalıcı, M. (2017). AŞIRI ÖĞRENME MAKİNELERİ İLE HİSSE SENEDİ FİYAT TAHMİNİ. *Hacettepe Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, Cilt 35, Sayı 1, 2017, s. 67-88*
- Ramasamy, Savitha & Suresh, Sundaram & J. Kim, H. (2013). A Meta-Cognitive Learning Algorithm for an Extreme Learning Machine Classifier. *Cognitive Computation*.
- R. Collobert, S. Bengio, Y. Bengio, (2002). A parallel mixture of SVMs for very large scale problems, *Neural Comput* 14 (5) 1105–1114.
- R.E. Schapire, (1990). The strength of weak learnability, *Mach. Learn.* 5 (2) 197–227.
- Sastry P. S., Santharam G., Unnikrishnan K. P., "Memory neuron networks for identification and control of dynamical systems", *IEEE Transaction on Neural Networks*, cilt 5, no 2, 1994.
- Sezgin, N. (2016). Epileptik EEG işaretlerin aşırı öğrenme makineleri ile sınıflandırılması. *Mühendislik Dergisi, Cilt 7, Sayı 3, 2016, s. 481-490*.
- Y. Lan, Y.C. Soh, G.-B. Huang, (2009) . Ensemble of online sequential extreme learning machine, *Neurocomputing* 72 (135) 3391–3395
- Z.-L. Sun, T.-M. Choi, K.-F. Au, Y. Yu, (2008) . Sales forecasting using extreme learning machine with applications in fashion retailing, *Decis. Support Syst.* 46 (1) 411–419.

ÖZ GEÇMİŞ

Kişisel Bilgiler

Adı Soyadı : Ahmet BAKIRCI
Doğum Yeri ve Tarihi : Şanlıurfa 17/03/1991



Eğitim Durumu

Lisans Öğrenimi : Çukurova Üniversitesi, Mühendislik ve Mimarlık Fakültesi,
Bilgisayar Mühendisliği.(2013)
Bildiği Yabancı Diller : İngilizce

İletişim

Adres : Ankara

Tarih:21/08/2019