



**DOĞRUSAL TAHMİN FİLTRELERİNİN YENİLENEBİLİR ENERJİ
SİSTEMLERİNDE KULLANIMI İÇİN WEB ARAYÜZ TASARIMI**

YÜKSEK LİSANS TEZİ

Emrah ÜSTÜNDAĞ

Danışman

Doç. Dr. Fatih Onur HOCAOĞLU

İkinci Danışman

Dr. Öğr. Üyesi Emre AKARSLAN

YENİLENEBİLİR ENERJİ SİTEMLERİ ANABİLİM DALI

ARALIK 2018

Bu tez çalışması 17.Fen.Bil.72 numaralı proje ile BAPK tarafından desteklenmiştir.

AFYON KOCATEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

**DOĞRUSAL TAHMİN FİLTRELERİNİN YENİLENEBİLİR ENERJİ
SİSTEMLERİNDE KULLANIMI İÇİN WEB ARAYÜZ TASARIMI**

Emrah ÜSTÜNDAĞ

Danışman

Doç. Dr. Fatih Onur HOCAOĞLU

İkinci Danışman

Dr. Öğr. Üyesi Emre AKARSLAN

YENİLENEBİLİR ENERJİ SİSTEMLERİ ANABİLİM DALI

ARALIK 2018

TEZ ONAY SAYFASI

Emrah ÜSTÜNDAĞ tarafından hazırlanan “Doğrusal Tahmin Filtrelerinin Yenilenebilir Enerji Sistemlerinde Kullanımı için Web Arayüzü Tasarımı” adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 20/12/2018 tarihinde aşağıdaki jüri tarafından **oy birliği / oy çokluğu** ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **Yenilenebilir Enerji Sistemleri Anabilim Dalı’nda YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman : Doç. Dr. Fatih Onur HOCAOĞLU

İkinci Danışman : Dr. Öğr. Üyesi Emre AKARSLAN

Başkan : Prof. Dr. Ömer Nezih GEREK
Eskişehir T. Üniversitesi, Mühendislik Fakültesi

Üye : Doç. Dr. Fatih Onur HOCAOĞLU
Afyon K. Üniversitesi, Mühendislik Fakültesi

Üye : Dr. Öğr. Üyesi Emre AKARSLAN
Afyon K. Üniversitesi, Mühendislik Fakültesi

İmza







Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü Yönetim Kurulu’nun

...../...../..... tarih ve

..... sayılı kararıyla onaylanmıştır.

.....

Prof. Dr. İbrahim EROL

Enstitü Müdürü

BİLİMSEL ETİK BİLDİRİM SAYFASI
Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

20 / 12 / 2018


Emrah ÜSTÜNDAĞ

ÖZET
Yüksek Lisans Tezi

**DOĞRUSAL TAHMİN FİLTRELERİNİN YENİLENEBİLİR ENERJİ
SİSTEMLERİNDE KULLANIMI İÇİN ARAYÜZ TASARIMI**

Emrah ÜSTÜNDAĞ
Afyon Kocatepe Üniversitesi
Fen Bilimleri Enstitüsü

Yenilenebilir Enerji Sistemleri Anabilim Dalı

Danışman: Doç. Dr. Fatih Onur HOCAOĞLU

İkinci Danışman: Dr. Öğr. Üyesi Emre AKARSLAN

Günümüzde enerji talebinin giderek artması, mevcut enerji üretim tesislerinin işletme maliyetleri ve çevreye verdikleri zarardan dolayı, enerji ihtiyacının karşılanmasını yenilenebilir enerji sistemlerine yöneltmektedir. Planlamanın doğru yapılabilmesi üretimin doğru tahmin edilmesine bağlıdır. Herhangi bir verinin tahmininde birçok yöntem kullanılmaktadır. Doğrusal filtreler özellikle görüntü işleme alanında kullanılan tahmin araçları olup filtre şablonunun görüntü üzerinde gezdirilmesi ile gelecek verinin tahmin edilmesine olanak sağlar. Bu çalışmada doğrusal tahmin filtrelerinin farklı tahmin uygulamalarında kullanılabilmesi için bir web arayüzü tasarlanmıştır. Uygulama ASP.NET ortamında oluşturulmuştur. Tasarlanan arayüzde kullanıcı kendi filtre şablonunu iki ya da üç boyutlu oluşturabilmekte, tahminler gerçekleştirebilmekte, hata analizi yapabilmekte, tahmine ait grafikler elde edebilmektedir. Ayrıca arayüz, doğrusal tahmin filtreleri ile güneş ışıının tahmin edildiği çalışmalar refereans alınarak tasarlanmış olmasına rağmen, istenilen periyotta kaydedilmiş herhangi bir veri için tahmin uygulamaları yapılmasına olanak sağlayacak esnekliğe sahiptir.

2018, ix + 75 sayfa

Anahtar Kelimeler: Web Arayüz, Güneş Işınımı, Filtreleme, Programlama, Tahmin

ABSTRACT
M.Sc. Thesis

**AN INTERFACE DESIGN TO USE LINEAR PREDICTION FILTERS FOR
RENEWABLE ENERGY SYSTEMS**

Emrah ÜSTÜNDAĞ

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Renewable Energy Systems

Supervisor: Assoc. Prof. Fatih Onur HOCAOĞLU

Co-Supervisor: Asst. Prof. Emre AKARSLAN

Nowadays by increasing demand of energy, suppling of energy need trends to renewable energy systems due to the operating costs of existing power generation systems and their damage to the environment. Planning accuracy depends on accurate prediction of production. There are many methods for forecasting. Linear prediction filters are predominantly used in the image processing field, allowing the prediction of the future data by shifting the filter template over the image. In this research, an user web interface is designed which the linear prediction tools can be used and works in different locations. Application is made on ASP.NET environment with .NET Framework tools. The user can create his own filter template as two or three dimensional, can make forecasts, error analysis and can see graphs about the prediction. Even though user interface is designed with referance about solar radiation forecasting with linear prediction filter, also has flexibility about forecasting any intended data with different intended period.

2018, ix + 75 pages

Keywords: Web Interface, Solar Radiation, Filter, Programming, Prediction

TEŞEKKÜR

Bu araştırmanın konusu, deneysel çalışmaların yönlendirilmesi, sonuçların değerlendirilmesi ve yazımı aşamasında yapmış olduğu büyük katkılarından dolayı tez danışmanım Sayın Doç. Dr. Fatih Onur HOCAOĞLU, araştırma ve yazım süresince yardımlarını esirgemeyen Sayın Dr. Öğr. Üyesi Emre AKARSLAN'a, 17.Fen.Bil.72 proje numarası ile destek sunan BAPK'a, her konuda öneri ve eleştirileriyle yardımlarını gördüğüm hocalarıma ve arkadaşlarıma teşekkür ederim.

Bu araştırma boyunca maddi ve manevi desteklerinden dolayı aileme teşekkür ederim.

Emrah ÜSTÜNDAĞ
AFYONKARAHİSAR, 2018

İÇİNDEKİLER DİZİNİ

Sayfa

ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER DİZİNİ.....	iv
KISALTMALAR DİZİNİ	vi
ŞEKİLLER DİZİNİ	vii
ÇİZELGELER DİZİNİ.....	viii
RESİMLER DİZİNİ	ix
1. GİRİŞ	1
2. GÜNEŞ IŞINIMI TAHMİNİNDE DOĞRUSAL TAHMİN FİLTRELERİ.....	3
2.1 Güneş Işınımı Tahmininde Kullanılan Yöntemler.....	3
2.2 İki Boyutlu Doğrusal Tahmin Modeli	4
2.3 Çok Boyutlu Doğrusal Tahmin Modeli	8
2.3.1 Extraterrestrial Işınım Hesabı.....	9
2.3.2 İstatistiksel Performans Değerlendirmesi	11
3. WEB PROGRAMLAMA.....	13
3.1 HTML (Hyper Text Markup Language).....	13
3.1.1 HTML Tarihçesi	15
3.2 ASP.NET (Active Server Pages)	15
3.3 JavaScript.....	17
3.4 CSS (Cascading Style Sheets)	18
3.4.1 CSS Kullanımına Örnek	19
3.5 Web Standartları	21
4. TASARIM.....	24
4.1 Bootstrap Kullanımı.....	26
4.2 Postback Olayı ve Update Panel Kullanımı.....	26
4.3 Model ve Şablon Seçimi	27
4.4 Dosya Yükleme.....	29
4.5 Tahmin İşlemi ve Hata Sonuçlarını İzleme	31
5. UYGULAMALAR.....	34
5.1 Arayüz ile Güneş Işınım Tahmini.....	34
5.2 Arayüz ile Kısa Dönem Rüzgar Hızı Tahmini.....	38

6. DEĞERLENDİRME VE SONUÇ	40
7. KAYNAKLAR.....	41
ÖZGEÇMİŞ.....	44
EKLER	45
EK-1 Kullanım Kılavuzu.....	45
EK-2 Program Kodları	48-75



KISALTMALAR DİZİNİ

Kısaltmalar

ASP	Active Server Pages
CERN	Conseil European pour la Recherche Nucleaire
CSS	Cascade Style Sheet
HTML	Hyper Text Markup Language
IETF	Internet Engineering Task Force
JS	JavaScript
MABE	Mean Absolute Bias Error
MAD	Mean Absolute Deviation
MAPE	Mean Absolute Percentage Error
MBE	Mean Bias Error
MPE	Mean Percentage Error
PHP	Hypertext Preprocessor
RMSE	Root Mean Square Error
VB	Visual Basic
WWW	World Wide Web
W3C	World Wide Web Consortium
YES	Yenilenebilir Enerji Sistemleri
YSA	Yapay Sinir Ağları

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 2.1 Yapay sinir ağı modeli	3
Şekil 2.2 Zaman serisi datasının iki boyutlu data-resim formuna dönüşmesi.....	5
Şekil 2.3 Örnek filtre şablonu (iki boyut)	6
Şekil 2.4 Örnek filtre şablonu (çok boyut)	8
Şekil 2.5 Çok boyutlu filtreleme işlemi.....	8
Şekil 3.1 Web teknolojileri tarihçesi	13
Şekil 3.2 HTML ve HTML5 tasarım şablonları karşılaştırması	14
Şekil 3.3 ASP.NET mimarisi	16
Şekil 4.1 Program akış diyagramı	25

ÇİZELGELER DİZİNİ

Sayfa

Çizelge 3.1 HTML tarihçesi.....	15
Çizelge 5.1 Uygulamada hesaplanan optimal katsayılar.....	35
Çizelge 5.2 Güneş ışıınım tahmin değerleri ile gerçek değerler arasındaki hata	38
Çizelge 5.3 Rüzgar hızı tahmin değerleri ile gerçek değerler arasındaki hata	39



RESİMLER DİZİNİ

Sayfa

Resim 3.1 ASP.NET ortamında oluşturulan FileUpload aracı	19
Resim 3.2 CSS ve JS ile görselleştirilmiş FileUpload aracı	19
Resim 4.1 Uygulama açılış ekranı	24
Resim 4.2 Hata mesajı(Mobil görünüm).....	26
Resim 4.3 Model seçimi.....	27
Resim 4.4 Şablon seçimi	28
Resim 4.5 Extraterrestrial Işınım hesabının uygulamada liste olarak gösterimi.....	29
Resim 4.6 Örnek güneş ışıınım verisi	30
Resim 4.7 Veri dosyasının yüklenmesi.....	30
Resim 4.8 Hesaplanan optimal katsayılar ve güneş ışıınım verisi	31
Resim 4.9 Tahmin işlemi sonrası ekran görüntüsü	32
Resim 4.10 Hata sonuçları	33
Resim 5.1 Güneş ışıınım tahmininde kullanılan filtre şablonu	35
Resim 5.2 Afyonkarahisar ili 2012 yılı güneş ışıınım verileriyle MOD-2 ile modellenen 2012 yılı verileri tahmini	36
Resim 5.3 Afyonkarahisar ili 2012 yılı güneş ışıınım verileriyle MOD-2 ile modellenen 2013 yılı verileri tahmini	37
Resim 5.4 Rüzgar hızı tahmininde kullanılan filtre şablonları	38

1. GİRİŞ

Tahmin yöntemleri işletmelerde veya yeni tesis kurulumlarında tahmin başarısına göre farklı metotlarla kullanılmaktadır. Karar verme yetkisinde olan her kişi geleceğin tahmin edilmesine ihtiyaç duymaktadır. Talep tahmini işletmelerde, piyasa taleplerinin ölçülerek gelecek zamanda oluşacak talebin tahmini ile üretim, depolama, istihdam ihtiyaçlarına yapılacak yatırımın planlaması tahmin başarısı ile orantılı olarak başarılı bir şekilde yapılabilmektedir. Literatürde tahmin yöntemleri olarak Zaman Serileri, Yapay Sinir Ağları, Regresyon, Bulanık Mantık, Dalgacık Ağları gibi metotlar geliştirilmiştir.

Artan enerji ihtiyacı ile günümüzde yenilenebilir enerji sistemlerinin temiz, sürekli ve hammadde maliyeti nedeni ile talebin karşılanmasını bu alana itmektedir. YES'lerde Kurulacak tesisin üretim planlamasının yapılması büyük önem arz etmektedir. Farklı sistemlerde hammadde veya ana maddenin sağlanma koşullarını tahmin etmek sistem kurulumlarının planlanmasında kullanılmaktadır. Güneş enerjisinden elektrik üreten sistemlerde panelin üreteceği elektrik enerjisi, panelin üzerine düşen güneş ışınımı ile doğrudan orantılıdır (Akarslan *et al.* 2014).

Güneş ışınım verisinden birçok tahmin modeli geliştirilmiştir. Çalışmalar, daha çok lineer olmayan yapılarıyla ön plana çıkan Yapay Sinir Ağları, Bulanık Mantık, Wavelets gibi modeller üzerinde yapılmıştır. Hoccoğlu vd. (2008) tarafından geliştirilen 2 boyutlu doğrusal tahmin filtreleri ile Güneş Işınım verisi iki boyutlu diziye dönüştürülmüş, bu dizinin elemanları yerine denk gelen pixellere dizi elemanının değeri ile renklendirilerek her data resme dönüştürülmüştür. Resim üzerinde önceden belirlenen filtre şablonları ile tarama yapılarak tahmin işlemi yapılmıştır.

Daha sonra Akarslan vd. (2014), çok boyutlu yaklaşım ile ışıma etkisi olan sıcaklık ve extraterrestrial ışıma dataları kullanılarak tasarlanan şablonlar sıcaklık ve extraterrestrial datalarına uygulanarak ışıma tahmini yapılmıştır. Geliştirilen metotlarda her bir şablonun verdiği sonuçlarda hata analizi yöntemleriyle karşılaştırılmıştır. Daha çok data ve daha çok geçmiş zaman katsayısının daha başarılı sonuçlar verdiği

görülmüştür.

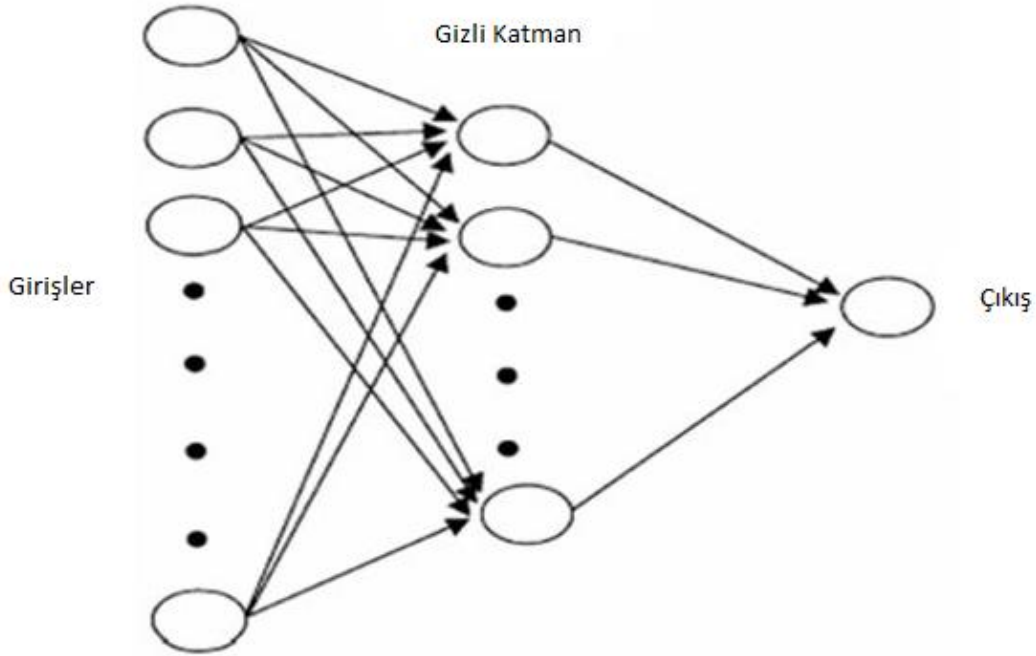
ASP Microsoft tarafından geliştirilen sunucu taraflı çalışan script dili olan bir teknolojidir. Kang vd. (2007) ASP. NET ile Voice User İnterface tasarımı yapmış ve diğer programlama dillerine göre iletişim hızının daha iyi olduğu sonucuna varmıştır. Bounnady vd. (2016) ASP.NET ile PHP programlama dillerini sayfa yükleme, algoritma işleme, veritabanı yönetimi, dosya yükleme ve dışsal dosya okuma-yazma işlemleri bakımından karşılaştırmış, ASP.NET'in sayfa yükleme, dışsal dosya işlemleri ve algoritma işlemlerinde daha hızlı olduğunu tespit etmiştir.

Bu çalışma kapsamında bahsedilen çalışmalara paralel olarak iki boyutlu doğrusal filtre yaklaşımının ister sadece kendisini, ister ilişkisel daha çok data ile tahmin edilmesine olanak sağlayan, kullanıcının kendi filtre şablonunu tasarlayabileceği, sonuçlarını farklı hata analiz yöntemleriyle gözlemleyip karşılaştırabileceği, web tabanlı kullanıcı arayüzü tasarımı yapılmıştır. Web üzerinden çalışması sayesinde mobil olarak her mekânda kullanılabilmesi, kendi datasına en uygun şablonu bulabilmesi amaçlanmıştır. Kullanıcı dostu bu yazılımda geliştirilen algoritma ile hesaplanan katsayılarla farklı bir datanın tahmini yapılmasına da olanak sağlanmıştır.

2. GÜNEŞ IŞINIMI TAHMİNİNDE DOĞRUSAL TAHMİN FİLTRELERİ

2.1 Güneş Işınımı Tahmininde Kullanılan Yöntemler

Güneş ışıını tahmininde lineer ve lineer olmayan yapılarla birçok model üzerinde çalışmalar yapılmıştır. Bunlardan lineer olmayan veriler için Yapay Sinir Ağları sistemlerin çözömlenmesinde kullanılan iyi bir yöntemdir. Öğrenme özelliğine sahip yöntem model kaynak insan beyni ve sinirleridir. 1943 yılında Pitts ve McCulloch tarafından “Sinir Aktivitesindeki Düşöncelere Ait Mantıksal Hesap” adlı çalışma ile YSA konusunda ilk adımlar atılmıştır. Günümüzde mühendislikte, tıpta, uzay çalışmalarında tahmin, modelleme, filtreleme gibi amaçlarla çalışmalar yapılmaktadır. Yapay Sinir Ağları temel anlamda giriş değerlerine bağılı çıkış değerleri oluşturur. Birden fazla parametrenin de kullanılıp giriş ve çıkış değerleri arasında lineer olmayan bir ilişki kurarlar. Bu ilişki ile kazanılan öğrenme kabiliyeti problemi en uygun şekilde çözer.



Şekil 2.1 Yapay sinir ağı modeli.

Güneş ışıınım tahmininde Hocaoglu vd. (2008) tarafından ileri-beslemeli Yapay Sinir Ağları üzerinde yapılan çalışmada tahmin sonuçları tek boyutlu ve iki boyutlu doğrusal filtreler ile karşılaştırılmış, lineer olmayan davranış gösteren datalarda YSA ile yapılan tahminin daha başarılı sonuç verdiği belirtilmiştir (Hocaoglu *et al.* 2008).

Zaman serileri, bir değişkenin zaman içindeki değişimini gözlemleyerek, çıkarımlarının zamana göre dağılım gösterdiği serilerdir. Zaman serileri frekanslıdır. Serilerin frekansları periyodik olarak günlük, aylık, yıllık değişen değerler alabilmektedirler. Zaman serileri zamansal sıralı karakteristik bir özelliğe sahiptir. Bu özellik zaman serilerinin yatay kesit serilerinden farklı olduğu durumdur. Kendini tekrar eden durumlar için “t” indisi serilerin zaman serileri olduğunu göstermektedir. Gözlemlerin ardışık bir biçimde olması gerekli değildir. Ancak belirli frekanslarda dizinin gidişatını takip etmesi, analizin hatasının minimize olması adına gerekli bir koşuldur. Böylece, serilerin verileri zamansal olarak elde edilebilmektedir. Güneş ışıınımı tahmininde saatlik ışıınım verilerini bir boyutlu zaman serilerine dönüştürerek tahmin modelleri geliştirilmiştir (Chena *et al.* 2007-Cucumo *et al.* 2007-Kaplanis *et al.* 2007). Bu modellerde yıllık verilerde gün değişkeni modellemede göz önünde bulundurulmamıştır.

Linear olmayan modelleriyle Bulanık Mantık, Wavelets gibi yöntemler ile güneş ışıınımı tahmin algoritmaları geliştirilmiştir.

2.2 İki Boyutlu Doğrusal Tahmin Modeli

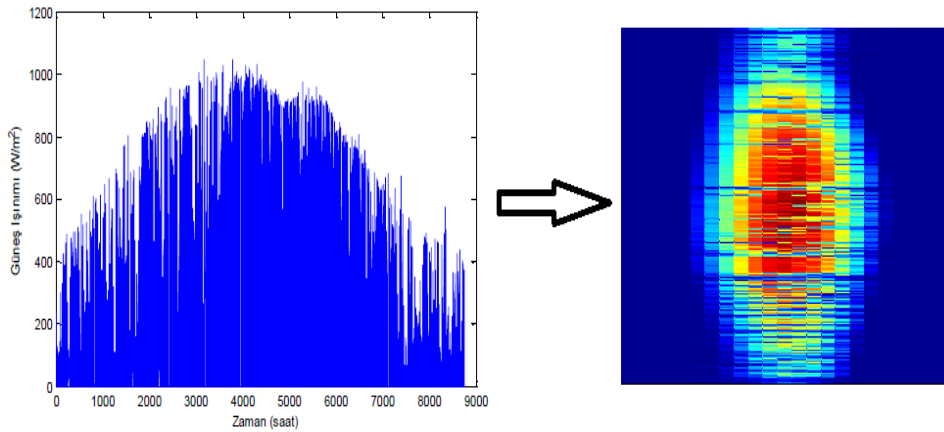
Doğrusal tahmin modeli, mevcut değeri geçmiş değerleri kullanarak tahmin eder. Tahmin edilen değer, geçmiş değerlerin doğrusal bir kombinasyonudur. Doğrusal tahmin katsayıları mevcut ve tahmin sinyali arasındaki farkı minimize edecek şekilde belirlenir. Doğrusal tahminin temel eşitliği aşağıdaki gibidir.

$$\hat{s}(n) = \sum_{k=1}^p (a_k s(n-k)) \quad (2.1)$$

$\hat{s}(n)$, mevcut $s(n)$ değeri için α_k katsayıları ile p adet örneğin doğrusal kombinasyonu ile elde edilen tahmin değerini ifade etmektedir. Doğrusal filtreleme işlemi bu mantık ile yapılır. Burada amaç, tahmin hatasını minimize etmektir ve bu aşağıdaki eşitliği minimize ederek elde edilir.

$$E = \sum_n e^2(n) \quad (2.2)$$

Burada E kısa zamanlı tahmin hatasıdır ve $e(n)$ mevcut değer ile tahmin değerinin farkına eşittir. Hocaoglu vd. (2008), güneş ışıının tahmini için doğrusal tahmin filtrelerini kullanmayı önermişlerdir. Bu filtrelerin kullanılması için güneş ışıının zaman serisi öncelikle 2 boyutlu matrise dönüştürülür. Bu matrisin satır ve sütunları, sırasıyla, yılın günleri ve günün saatlerine karşılık gelir. Bu yeni temsili en önemli avantajı güneş ışıının zamanla değişimi ile alakalı daha kolay anlaşılır bir fikir sağlamasıdır.



Şekil 2.2 Zaman serisi datasının iki boyutlu data-resim formuna dönüştürülmesi.

Şekil 2.2’de zaman eksenin gösterilen data iki boyutlu resim datasına dönüştürülmüştür. Resimde yer alan pixellerin yatay eksenini yılın günlerini dikey eksenini yılın o gününün saatini ifade eder. Tam o noktada yer alan pixel zaman serisinde yer alan değerine göre sinyal işleme tekniğine göre renklendirilir. Görüntü işleme tekniğinde tahmin edilecek pixeli etkileyen komşu pixelleri kullanarak modellenen birçok filtreleme yöntemi vardır.

Doğrusal tahmin filtreleri, verinin geçmiş değerlerini kullanarak mevcut değerini modeller. Modellemede hangi verilerin kullanılacağı, filtre şablonu ile belirlenmektedir. Farklı filtre şablonlarının kullanımı, tahmin başarısı üzerinde farklı etkilere neden olmaktadır. Bu nedenle daha yüksek bir tahmin başarısı için en uygun filtre yapısının belirlenmesi gerekir. Aşağıdaki şekilde verilen gibi bir filtre şablonu olduğu varsayılırsa;

Bir Gün Bir Saat Öncesi $Z_{i-1,j-1}$	Bir Gün Öncesi $Z_{i-1,j}$
Bir Saat Öncesi $Z_{i,j-1}$	$Z_{i,j}=?$

Şekil 2.3 Örnek filtre şablonu (iki boyut).

Tahmin edilecek pixel'in ($Z_{i,j}$) daha önceki pixeller ile ilişkisini açıklamak için her pixel bir katsayı ile çarpılır. Filtre katsayılarını hesaplamak için (a_1, a_2, a_3);

$$Z_{i,j} = a_1.Z_{i-1,j-1} + a_2.Z_{i-1,j} + a_3.Z_{i,j-1} \quad (2.3)$$

formülü kullanılır.

Filtre şablonu iki boyutta kaydırılır, her satır boyunca j değeri değişirken, her sütun boyunca i değeri değiştirilir. Burada i günü, j saati ve $Z_{i,j}$, i . gün ve j . saatteki güneş ışınlam değerini ifade eder. Kaydırma işleminin her adımında (her farklı i, j noktasında), (i, j) koordinatı için tahmin, filtre şablonunun belirlediği geçmiş değerler kullanılarak yapılır. İki boyutlu doğrusal tahminde, kaydırılan şablon içindeki her bir geçmiş örnek, karesel tahmin hatasını en küçükleyecek şekilde, belirlenmiş uygun katsayılar ile boyutlandırılır. Optimizasyon algoritması ile doğrusal filtre katsayıları a_1, a_2 ve a_3 optimize edilir ve bu durumda yapılan tahmindeki hata aşağıdaki eşitlik ile hesaplanır.

$$e_{i,j} = \check{Z}_{i,j} - Z_{i,j} \quad (2.4)$$

Tüm resim tahminine karşılık gelen toplam hatanın enerjisi aşağıdaki formül ile hesaplanabilir.

$$\varepsilon = \sum_{i=1}^m \sum_{j=1}^n (e_{i,j})^2 \quad (2.5)$$

Burada m ve n güneş ışınlam resminin boyutlarına karşılık gelmektedir. Bir yıllık güneş ışınlam verisi için m değeri 365'e ve n değeri 24'e karşılık gelir. Fonksiyonu minimize edecek filtre katsayıları;

$$\frac{\partial \varepsilon}{\partial a_1} + \frac{\partial \varepsilon}{\partial a_2} + \frac{\partial \varepsilon}{\partial a_3} = 0 \quad (2.6)$$

Eşitliği kullanılarak,

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix} \quad (2.7)$$

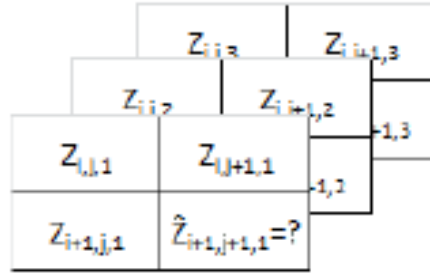
Burada r_{ij} , tahmin filtre şablonu içindeki z_i ve z_j pikselleri arasındaki korelasyonu, a_i , doğrusal filtre katsayısını ve r_k , tahmin edilen piksel ile filtre şablonu içindeki her bir piksel arasındaki korelasyonu temsil eder. Kompakt bir şekilde ifade edilecek olursa;

$$R \cdot a = r \quad (2.8)$$

Burada, a , filtre katsayılarını içeren vektör, r , tahmin edilecek piksel ile filtre şablonu içindeki pikseller arasındaki korelasyon vektörü ve R , tahmin şablonu içerisindeki korelasyonları içeren matristir.

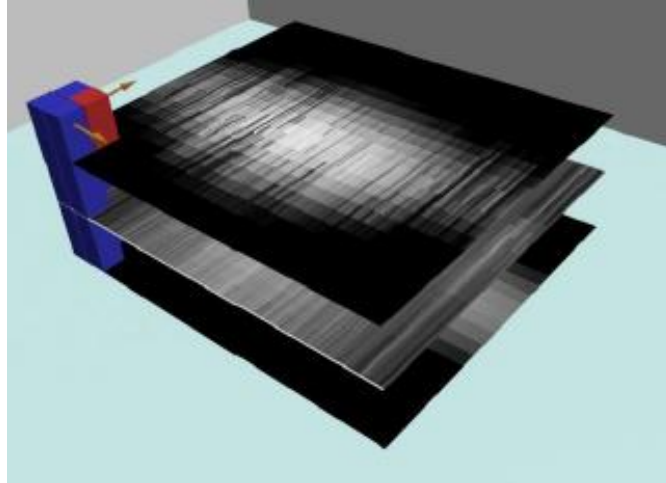
2.3 Çok Boyutlu Doğrusal Tahmin Modeli

Bir önceki bölümde anlatılan mantık kullanılarak Akarslan vd. (2014) ise bu tahmin filtrelerini çok boyuta taşımıştır. Bu yeni yaklaşım güneş ışıınım tahmininde güneş ışıınının kullanılmasının yanında, bu veri ile korelatif başka meteorolojik verilerin kullanımına da olanak sağlamaktadır. Buna örnek bir filtre şablonu aşağıda görölmektedir.



Şekil 2.4 Örnek filtre şablonu (çok boyut) (Akarslan *et al.* 2014).

Bu yöntemde veriler 2 boyutlu resimlere dönüştüröldükten sonra üst üste bindirilerek, belirlenen filtre şablonu resim üzerinde kaydırılır. Bu çok boyutlu filtreye ait üç boyutlu bir gösterim Şekil 2.5’de görölmektedir.



Şekil 2.5 Çok boyutlu filtreleme işlemi (Akarslan *et al.* 2014).

Şekilden de göröleceği üzere filtre uygun tahmin katsayılarını belirlemek için kaydırılırken, aynı zamanda üst üste bindirilmiş resimler arasında irtibatı da

sağlamaktadır. Sistemin matematiksel izahı iki boyutlu filtreler ile benzerdir ve doğrusal tahmin filtre katsayıları aşağıdaki formül ile hesaplanır.

$$a = R^{-1}.r \quad (2.9)$$

Filtre katsayıları belirlendikten sonra bu katsayılar kullanılarak tahminler gerçekleştirilir. Farklı filtre şablonları için farklı sayıda katsayı üretilir. Dolayısıyla hesaba ve tahmine giren katsayı sayısı da şablona göre değişir.

Tahmin edilecek verinin geçmiş zaman değerleri ile tahmin edilecek veriye bağlantılı datalar bu yöntem kullanılarak modellenenabilir. Akarslan vd. (2014) çalışmasında birçok model ve birçok şablon üreterek, bu modellerde extraterrestrial ışınlam ve sıcaklık gibi datalar veya bunların türevlerini kullanarak, modeller arasındaki performansı karşılaştırmıştır.

2.3.1 Extraterrestrial Işınlam Hesabı

Extraterrestrial ışınlam atmosfer üzerindeki güneş yoğunluğu olarak bölgenin solar geometrisi kullanılarak hesaplanır. Dünyanın eliptik yörüngesinden dolayı yıl boyunca dünya ile güneş arasındaki mesafenin değişmesiyle orantılıdır. Bunun yanında dünya yörüngesinin eksantriğinin de hesaba katılması ile eksantrik (eccentricity) düzeltme faktörü;

$$E_n = I_{sc}(1 + 0,033\cos(\frac{360n}{365})) \quad (2.10)$$

n yılın günlerini belirtir. Örneğin 1 Ocak için değeri '1' 31 Aralık için değeri '365' 'e tekabül etmektedir.

I_{sc} Güneş sabitidir ve metrekareye düşen güneş ışınlam yoğunluğu olarak;

$$I_{sc} = \Omega T^4 \left(\frac{4\pi R}{4\pi D} \right)^2 = 1367 \frac{W}{m^2} \quad (2.11)$$

formülü ile açıklanır. Eşitlikte;

Ω = Stefan-Boltzman Sabiti'ni

R= Güneşin yarıçapı'nı

D= Güneş ile Dünya arasındaki mesafe'yi

ifade eder.

Her hangi bir gün (n), herhangi bir saate (h) karşılık gelen Extraterrestrial ışıınım hesabı;

$$E_{n,h} = E_n \cdot \cos(\theta_z) \quad (2.12)$$

formülü ile ifade edilir. θ_z Zenith açısıdır ve değeri aşağıdaki formülle hesaplanır.

$$\cos(\theta_z) = \cos(\Phi) \cdot \cos(\delta) \cdot \cos(W) + \sin(\Phi) \cdot \sin(\delta) \quad (2.13)$$

Burada Φ mekanın enlemini ifade eder. W saat açısıdır ve değeri aşağıdaki formülle hesaplanır.

$$W = \left(2\pi \frac{h-12}{24} \right) \quad (2.14)$$

h günün saatidir. Sapma açısı olarak δ ise ;

$$\delta = 23,45 \cdot \sin \left(2\pi \frac{284+n}{365} \right) \quad (2.15)$$

olarak hesaplanır (Kharseh M.). Yağmur, bulutlanma gibi atmosferik olaylar nedeni ile güneş ışıının şiddeti extraterrestrial ışıının değerinden daha küçük çıkmaktadır (Akarslan *et al.* 2014).

2.3.2 İstatistiksel Performans Değerlendirmesi

Literatürde tahmin modellerinin hata oranlarını hesaplamaya ve istatistiksel olarak değerlendirmeye yarayan hata test yöntemleri vardır. Bunların başlıcalarından MPE (Ortalama hata yüzdesi) gerçek değer ile tahmin edilen değer arasındaki sapmanın yüzdesel değeridir. Aşağıdaki formül ile hesaplanır. (Bütün formüllerde T_i modelden tahmin edilen değer, G_i ise gerçek değerdir.)

$$MPE = \frac{1}{n} \sum_{i=1}^n \frac{T_i - G_i}{G_i} \times 100 \quad (2.16)$$

MPE hesaplama sonucunda T_i 'nin G_i 'den küçük olduğu durumlarda değerler negatif çıkacaktır. Dizide bulunan bir noktada negatif bir noktada pozitif çıkması toplam hata hesabını düşük çıkarmaktadır. Bu durum MAPE (Ortalama mutlak hata yüzdesi) kriteriyle giderilmiştir. MAPE hesabı;

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \left(\frac{T_i - G_i}{G_i} \times 100 \right) \right| \quad (2.17)$$

formülü ile ifade edilir. Hem negatif hem pozitif bileşenler içeren modellerde daha doğru sonuçlar vermektedir. MBE (Ortalama sapma hatası) modellerin uzun dönem performansları hakkında başarılı sonuçlar vermektedir ve aşağıdaki formül ile hesaplanır.

$$MBE = \frac{1}{n} \sum_{i=1}^n (T_i - G_i) \quad (2.18)$$

Aynı şekilde MPE-MAPE arasındaki ilişkiye benzer hata değerlerinin mutlak değerinin alınmasıyla MABE (Ortalama mutlak sapma hatası) hesaplanır.

$$MABE = \frac{1}{n} \sum_{i=1}^n |T_i - G_i| \quad (2.19)$$

RMSE (Ortalama karekök hatanın karekökü) ise kısa dönem tahmin modelleri hakkında

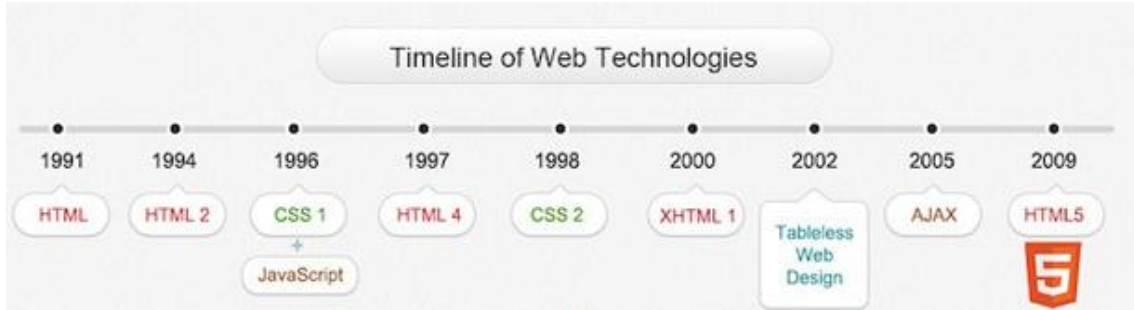
değerlendirme sonuçları verir. RMSE formülü aşağıdaki gibidir.

$$RMSE = \left[\frac{1}{n} \sum_{i=1}^n (T_i - G_i)^2 \right]^{\frac{1}{2}} \quad (2.20)$$



3. WEB PROGRAMLAMA

Web teknolojileri zamanın ihtiyaçlarına göre hızla farklı platformlara, farklı dillere, ve farklı geliştiricilere gereksinim duymaktadır. Yapılacak projenin işlevine göre Web tasarım projelerinde işaretleme dillerinden (HTML), stil dillerinden (CSS) ve scripting dilinden (Javascript) faydalınır. Bu diller kullanılarak hazırlanan tasarıma fonksiyonellik kazandırmak için ruby, asp, asp.net ve php gibi programlama dillerinden faydalanılır. Şekil 3.1’de web programlamaya yön veren teknolojilerin tarihçesi verilmiştir.



Şekil 3.1 Web teknolojileri tarihçesi.

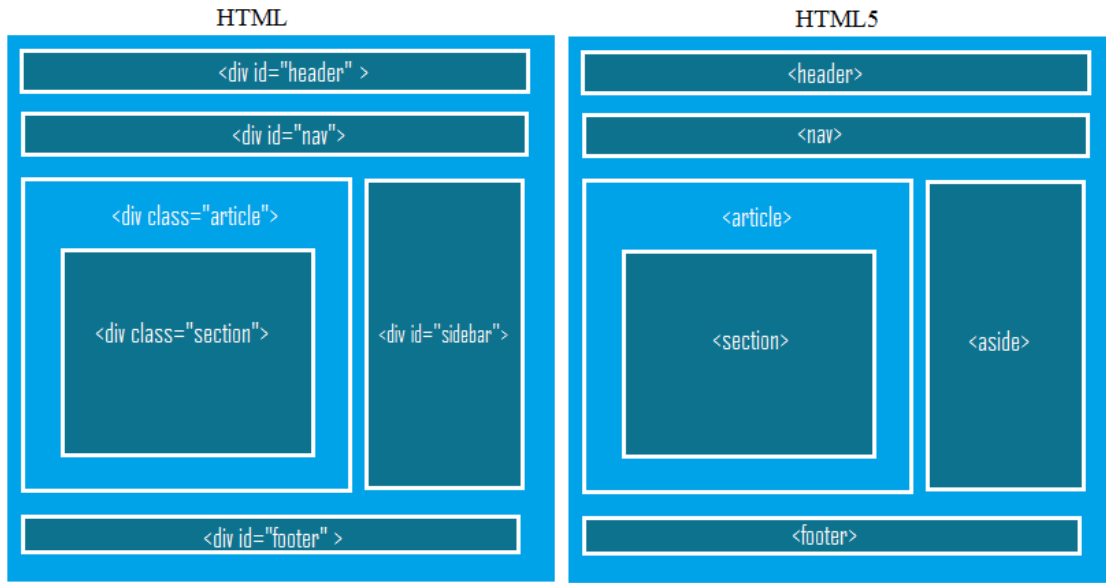
3.1 HTML (Hyper Text Markup Language)

HTML (Hyper Text Markup Language) internet tabanlı çalışan web sayfalarının oluşturulmasına imkan sağlayan bir betik dilidir. Öğrenmesi basit yaratıcılığa fazla açık olmayan bir dil olarak web programlamanın temel unsurudur. Günümüzde kullanılan son standardı HTML5 olarak karşımıza çıkmaktadır. HTML5 geliştirme çalışmaları devam etmektedir. Bu çalışmalarda görsellik ve pratik uygulamalar üzerinde yoğunlaşmıştır [int.kyn 3].

Sayfalarda tasarım ve şablon oluşturmaya olanak sağlayan HTML programlama dili değildir. Kontrol veya işlem gibi işlevleri yerine getiremez. HTML ile kod yazarken herhangi bir derleyiciye ihtiyaç duyulmaz. Bilgisayarda oluşturulacak metin belgesi ile bile HTML dili kullanılarak kodlar yazılıp çalıştırılabilir. Bir başka deyişle diğer

programlama dillerinde olduğu gibi (C, VB gibi) yazılan kodların derlenmesine gerek yoktur.

HTML'nin daha önceki sürümleri ile günümüzde kullanılan HTML5'in şablonları arasında fark Şekil 3.2'de karşılaştırmalı verilmiştir. HTML5 artık görülen syntax yapısı ile kullanılmaktadır.



Şekil 3.2 HTML ve HTML5 tasarım şablonları karşılaştırması.

HTML dosyaları sunucu tarafında “.html” uzantısı ile saklanır. Bir web sitesi temelde, HTML yazıları ve araçları içerir. HTML içermeyen web siteleri veri girişi gibi işlemlerden uzaktır. Örneğin kullanıcı adı ve şifre bilgisi girilen bir araç kutusunun oluşturulması HTML ile sağlanır. Tarayıcıda bir siteye giriş için tıklandığında adres çubuğunda `http://www.*****.com/index.html`, `http://www.*****.com/index.asp`, `http://www.*****.com/index.php` olarak karşımıza çıkar. Eğer “.html” olarak karşımıza çıkıyorsa sunucu o adresin sunucu tarafına atılan `index.html` dosyasında yazılı olanlar görüntülenir. Dosya aktarımları için HTTP (Hyper Text Transfer Protocol) kullanılır.

3.1.1 HTML Tarihçesi

HTML ilk önce Tim Berners-Lee tarafından 1991 yılı biterken oluşturulmuştur. 1991-1995 yılları arasında HTML üzerine birçok çalışma yapılmıştır. Yapılan çalışmalar ile birçok geliştirme yapıldı. Bu değişiklikler ilk olarak CERN’de, daha sonra da IETF’da sunulmuştur. W3C’nin kurulduktan sonra HTML herkes tarafından geliştirilebilir hale gelmiştir. İlk HTML geliştirmesi 1995 yılında HTML 3.0 ile başarısız olduğu söylenmektedir. Daha çok HTML 3.2 adıyla bilinen geliştirme 1997’de tamamlanmıştır. Daha sonra HTML4 ise 1998’de bitirilmiştir. 2009 yılında ise HTML5 olarak karşımıza çıkmaktadır. Hiper Metin İşaretleme Dili www’in yayın dilidir [int.kyn 4].

Çizelge 3.1 HTML Tarihçesi.

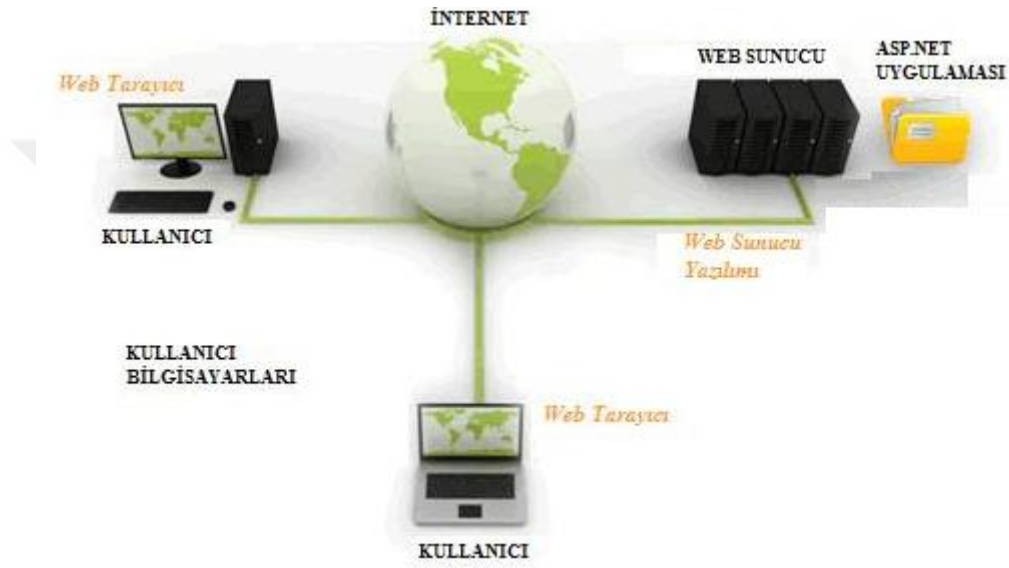
Sürüm	Yayınlanma Yılı
HTML+	1993
HTML2.0	1995
HTML3.2	1997
HTML4.01	1999

3.2 ASP.NET (Active Server Pages)

ASP Microsoft tarafından geliştirilen sunucu taraflı çalışan script dili olan bir teknolojidir. Ne kadar script dili de olsa ASP sunucu tarafında uygulama çalıştırılmasına da olanak sağlar. 1997 yılında geliştirilen platform kullanım kolaylığı sayesinde sektörde hızla popülerlik kazanmıştır. Daha sonra 2002 yılında ASP.NET, .NET Framework platformu altında, .NET kütüphanesini kullanarak daha hızlı, verimli ve dinamik uygulamalar geliştirilmesine olanak sağlayan bir yapı olarak karşımıza çıkmıştır.

Yapısı itibarıyla kullanıcıya daha yakın olan ASP.NET klasik ASP (Active Server Page)’den farklı olarak .NET Framework’ün altında IIS (Internet Information Server) ile bütünleşik çalışan bir mimari olup .NET Framework denilen çekirdek sistem üzerinde çalışmaktadır (Aytekin 2009).

.NET Framework Microsoft tarafından açık internet protokolleri üzerine kurulmuş uygulama geliştirme platformu olarak ASP.NET teknolojisini içerisinde barındırarak web tarayıcı uygulamalarında yazılım geliştirme olanağı sağlar. Sadece ASP.NET ile uygulama geliştirmek pek mümkün değildir. Bunun için web dili HTML veya XHTML ile script dillerinden (VBScript veya JScript) bilmek bu problemi çözer. ASP.NET teknolojisi HTML5, CSS ve JavaScript tabanlı olarak milyonlarca kullanıcıya hitaben kolay ve hızlı web sitesi tasarımı yapılmasına olanak sağlar (Jinglin 2009).



Şekil 3.3 ASP.NET mimarisi.

ASP. NET ile uygulama geliştirmenin avantajlarını şu şekilde sıralanabilir;

- Nesneye dayalı programlama dillerinin kullanımı ile daha verimli uygulamalar geliştirilmesine olanak sağlar.
- Dinamik WEB uygulamaları geliştirilebilmektedir.
- İçerdiği toollar sayesinde hazır kontroller ile daha kısa zamanda yazılım yapılmasını sağlar.
- .NET Framework platformunun önemli bir bölümüdür. .NET kütüphanesi ile zengin metotlar içermektedir.
- İçerik ve kod farklı dosyalar içerisinde sunucuya yüklenebilir.
- Hızlı derleme ve hata görme olanağına sahiptir.

İçerik ve program kodlarının birbirinden tamamen ayrı olduğu ASP .NET ; klasik ASP’deki scriptler ve istemci tabanlı uygulamalar yerine, doğrudan kod yazma ve uygulamaların sunucuda çalışıp sonuçlarının istemci bilgisayarlara aktarıldığı sunucu tabanlı/trafı bir modeli desteklemektedir. Tamamen nesneye dayalı programlama tekniklerini kullanan ASP. NET teknolojisi ile her türden veritabanına rahatlıkla bağlanabilmekte ve VB. NET, C# .NET, Java. NET, PHP. NET vb. birçok dille programlar geliştirilebilmektedir (Aytekin 2009).

Açılan internet sitesinde adres çubuğunda dosyanın “.asp” uzantılı olması uygulamanın ASP ile “.aspx” uzantılı olması ise ASP.NET ile geliştirildiğinin göstergesidir. ASP uzantılı bir dosya temelde HTML dosyası ile aynıdır. Dosya HTML ve Script dosyası içerebilir. Ayrıca bu dosya içerisindeki ASP kodları sayesinde sunucu tarafında derlenerek sayfayı açacak kullanıcıya HTML kodları gönderilir.

3.3 JavaScript

C dilinin tarayıcılara uyarlanmış hali diyebileceğimiz JavaScript Brandan Eich tarafından 1995 yılının sonlarına doğru geliştirilmiştir. O zamanlar Eich Netscape’te çalışmaktaydı ve geliştirilen dil “Mocha” ismiyle çıkartılmıştır. 1996 yılında adı JavaScript olarak değiştirilmiştir. ECMA tarafından standartlaştırılmıştır.

JavaScript web sayfa tasarımında sayfaya dinamizm katılmasını sağlar. HTML ile yazılmış kodlar daha önce anlatıldığı gibi statik olarak sayfada yer alır. Javascript eklentisi ile sayfada yer alan nesneler istenirse değiştirilir veya hareket ettirilebilir. Yazılan kodlarda karşılaştırma veya fonksiyon oluşturarak bu işlemler yapılır. Web sayfalarına dinamik içerik kazandırmak ve kullanıcı ile iletişimin sağlanmasına olanak sağlar. JavaScript ile ister istemci tarafında ister server tarafında çalıştırılabilen kodlar yazılabilir. Sunucu tarafında dosya oluşturabilmekte veya alınabilmektedir. Nesneye dayalı bir dildir. JavaScript herşeyi nesne olarak görür ve yeni özellik ve metotlar dinamik olarak eklenir. Miras özelliği prototip mekanizmasıyla gerçekleştirilir.

3.4 CSS (Cascading Style Sheets)

CSS (Cascading Style Sheets), site görselliğini düzenlememize olanak sağlayan bir markup dilidir. CSS web tasarımcıları için web sitesi içeriği ile sayfa düzenini birbirinden ayırarak projede bulunan tüm sayfaların görselliğini içerikten bağımsız bir şekilde tasarlamaya olanak sağlar (Roman 1997).

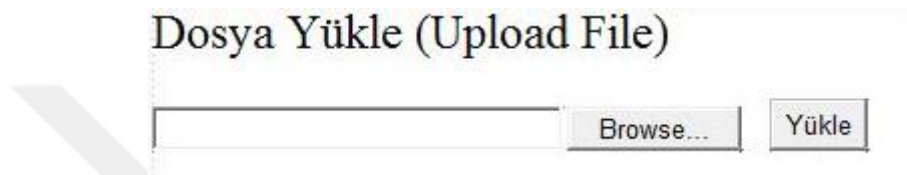
CSS bir programlama dili değildir. Sayfaya eklenmiş araçların (form, buton, dosya yükleme aracı gibi.) renk, şekil, görünüm gibi özelliklerini düzenlememize imkan sağlar. Projemizde CSS stil sayfaları ile gölgelendirme, renklendirme gibi butonun farklı bir görüntüye sahip olmasını sağlayarak sayfamızın görselliğini geliştirebiliriz. Bu görünüm özellikleri HTML kullanılarak da yapılabilmektedir. Ancak HTML ile yapılabilen düzenlemeler kısıtlıdır. İleri seviyede görsellik sağlayabilmek amacı ile(gölgelendirme, saydamlaştırma vb.) sayfa tasarımında CSS kullanmak kaçınılmaz hale gelmektedir. Ayrıca CSS bir programlama dili değil, sitenin makyajı ile alakalı düzen sağlayan bir web standardıdır.

Projede CSS kullanmanın faydalarını şu şekilde sıralanabilir;

- Bir araç için oluşturulmuş kodlardan oluşan görünüm özelliklerini bir başka sayfada veya araçta kullanılabilmesine olanak sağlar. Bu şekilde web sitesinin boyutu da gereksiz arttırılmamış olur.
- Farklı CSS parçalarını tek bir araç için kullanabilmeye olanak sağlar. (Gölgelendirme için hazırlanan CSS parçası ile renklendirme için hazırlanan CSS parçasını aynı araçta kullanılabilir.)
- Hazırlanan CSS parçası ile görsel özellikler, program parçasını değiştirmeden daha sonra değiştirilmesine olanak sağlar.
- Web sitesine ulaşımda sağlanan Windows veya Mobil platformlar için farklı CSS gösterimi sağlanarak farklı görünüm sunma imkanı sağlar.

3.4.1 CSS Kullanımına Örnek

Aşağıda CSS desteği ile hazırlanmış bir görsel araç ve bu görünümün ve işlevin sağlanması amacı ile yazılan örnek kodlar yer almaktadır. Resimde ASP.NET tarafından oluşturulan FileUpload aracı ve Resimde CSS ve JavaScript kullanılarak oluşturulan örnek bir araç verilmiştir.



Resim 3.1 ASP.NET ortamında oluşturulan FileUpload aracı.



Resim 3.2 CSS ve JS ile görselleştirilmiş FileUpload aracı.

HTML Kodları

```
<div class="container">
  <form class="form">
    <div class="file-upload-wrapper" data-text="Select your file!">
      <input name="file-upload-field" type="file" class="file-upload-field" value="">
    </div>
  </form>
</div>
```

CSS Kodları

```
@import "bourbon";
@import url(https://fonts.googleapis.com/css?family=Lato:400,700,300);

body {
  font-family: 'Lato', sans-serif;
}
```



```
.container {
  position: absolute;
  top: 0;
  right: 0;
  bottom: 0;
  left: 0;
  @include align-items(center);
  @include display(flex);
  @include justify-content(center);
  @include linear-gradient(to top right, #bf7a6b 0%, #e6d8a7 100%);
}
```

```
.form {
  width: 400px;
}
```

```
.file-upload-wrapper {
  $defaultColor: #4daf7c;
  $height: 60px;

  position: relative;
  width: 100%;
  height: $height;

  &:after {
    content: attr(data-text);
    font-size: 18px;
    position: absolute;
    top: 0;
    left: 0;
    background: #fff;
    padding: 10px 15px;
    display: block;
    width: calc(100% - 40px);
    pointer-events: none;
    z-index: 20;
    height: $height - 20px;
    line-height: $height - 20px;
    color: #999;
    border-radius: 5px 10px 10px 5px;
    font-weight: 300;
  }
}
```

```
&:before {
  content: 'Upload';
  position: absolute;
  top: 0;
  right: 0;
  display: inline-block;
  height: 60px;
  background: $defaultColor;
  color: #fff;
  font-weight: 700;
  z-index: 25;
  font-size: 16px;
  line-height: $height;
  padding: 0 15px;
}
```

```

text-transform: uppercase;
pointer-events: none;
border-radius: 0 5px 5px 0;
}

&:hover {
  &:before {
    background: darken($defaultColor, 10%);
  }
}

input {
  opacity: 0;
  position: absolute;
  top: 0;
  right: 0;
  bottom: 0;
  left: 0;
  z-index: 99;
  height: $height - 20px;
  margin: 0;
  padding: 0;
  display: block;
  cursor: pointer;
  width: 100%;
}
}

```

JAVASCRIPT Kodları

```

$("form").on("change", ".file-upload-field", function(){
  $(this).parent(".file-upload-wrapper").attr("data-text",
    $(this).val().replace(/.*(V|\\)/, " "));
});

```

3.5 Web Standartları

Web standardı dendiğinde değişik insanlar için farklı olaylar anlaşılabilir. Bazılarına göre ücretsiz olması bazıları için doğru kodlarla yazılması standarttır [int.kyn 1].

Aşağıdaki listelerde gerek kodun kalitesi gerekse içeriğin uyumluluğu gibi başlıklarla standart kaliteli bir web uygulaması sorgulanabilir.

Kodun kalitesine göre;

- Doğru belge tipi kullanılıyor mu?
- Karakter seti kullanılıyor mu?
- Geçerli (X)HTML kullanılıyor mu?

- Geçerli CSS kesmesi kullanılıyor mu?
- Gereksiz sınıflar veya kimlikler kullanılıyor mu?
- Kod yapısı iyi mi?
- Sitede kırık linki var mı?
- Hız ve boyut açısından performansı uygun mu?

İçerik-Sunum Ayırlığına göre;

- Bütün sunuşlarında CSS'i kullanıyor mu?
- Dekoratif imgeler, CSS içinde mi ya da (X)HTML içinde mi?

Kullanıcılar için erişilebilirliğe göre;

- Site metin boyutu için bağıl birim mi mutlak birim mi kullanıyor?
- Eğer font boyutu artırılırsa site yerleşimi etkileniyor mu?
- Site menü geçişleri görünür mü?
- Erişilebilir şekiller kullanıyor mu?
- Erişilebilir tablolar kullanıyor mu?
- Renk ve parlaklık yeterli mi?
- Linkler anlaşılır mı?

Cihazlar için erişilebilirliğe göre;

- Modern veya eski tarayıcılarda çalışıyor mu?
- CSS desteklenmediğinde içeriğe erişilebiliyor mu?
- Resimler desteklenmediğinde içeriğe erişilebiliyor mu?
- Lynx gibi metin tarayıcılarında da çalışıyor mu?
- Site yazdırma işlemi ile iyi çalışıyor mu?
- Mobil cihazlarla da iyi çalışıyor mu?
- Farklı çözünürlüklerde iyi çalışıyor mu?

Temel kullanılabilirliğe göre;

- Açık bir görsel hiyerarşi var mı?
- Başlık seviyeleri kolayca ayırt edilebilecek durumda mı?
- Sitenin akışı kolayca anlaşılabilir mi?

- Sitenin akışı tutarlı mı?
- Site tutarlı ve uygun dil kullanıyor mu?
- Site haritası sayfasına ve iletişim sayfasına sahip mi?
- Site içi arama aracına sahip mi? (büyük siteler için)
- Tüm sayfalarda ana sayfaya bir link var mı?
- Linklerin altı çizili mi?
- Ziyaret edilen linkler, diğer linklerden ayırt ediliyor mu?

Site yönetimine göre;

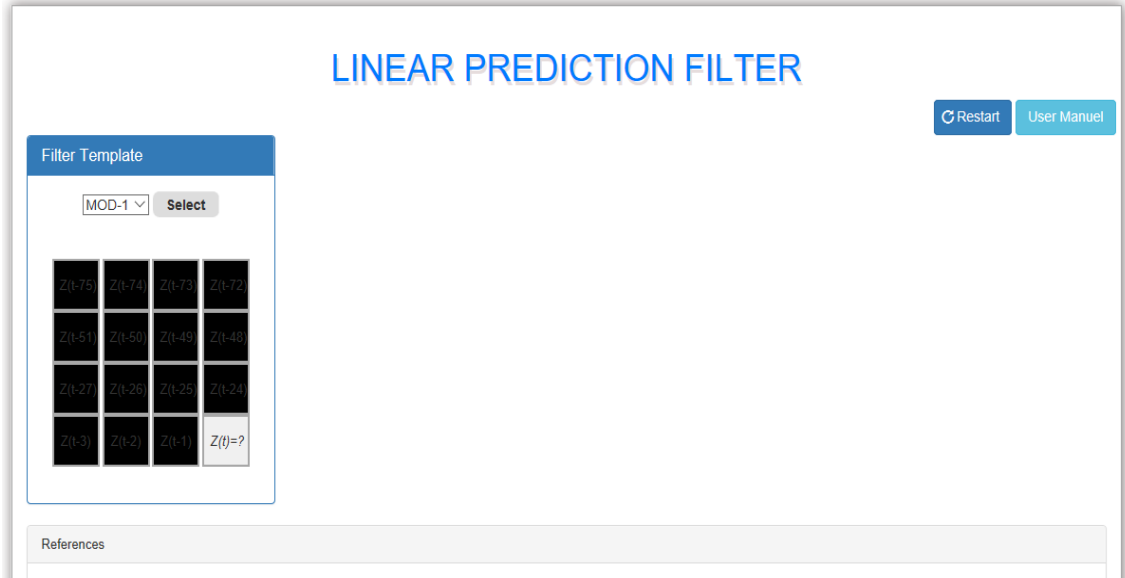
- Sitede çökme veya erişilebilirlik sıkıntısı olduğunda anlaşılabilir ve yardımcı bir 404 hata sayfası var mı?
- Site yardımcı adresler kullanıyor mu?
- URL'ler "www" yazmadan da çalışıyor mu?
- Sitede sık kullanılan simgeler var mı?

Bütün bu liste sitenin kullanılış amacına, biçimine ve içeriğine göre genişletilip daraltılabilir.

4. TASARIM

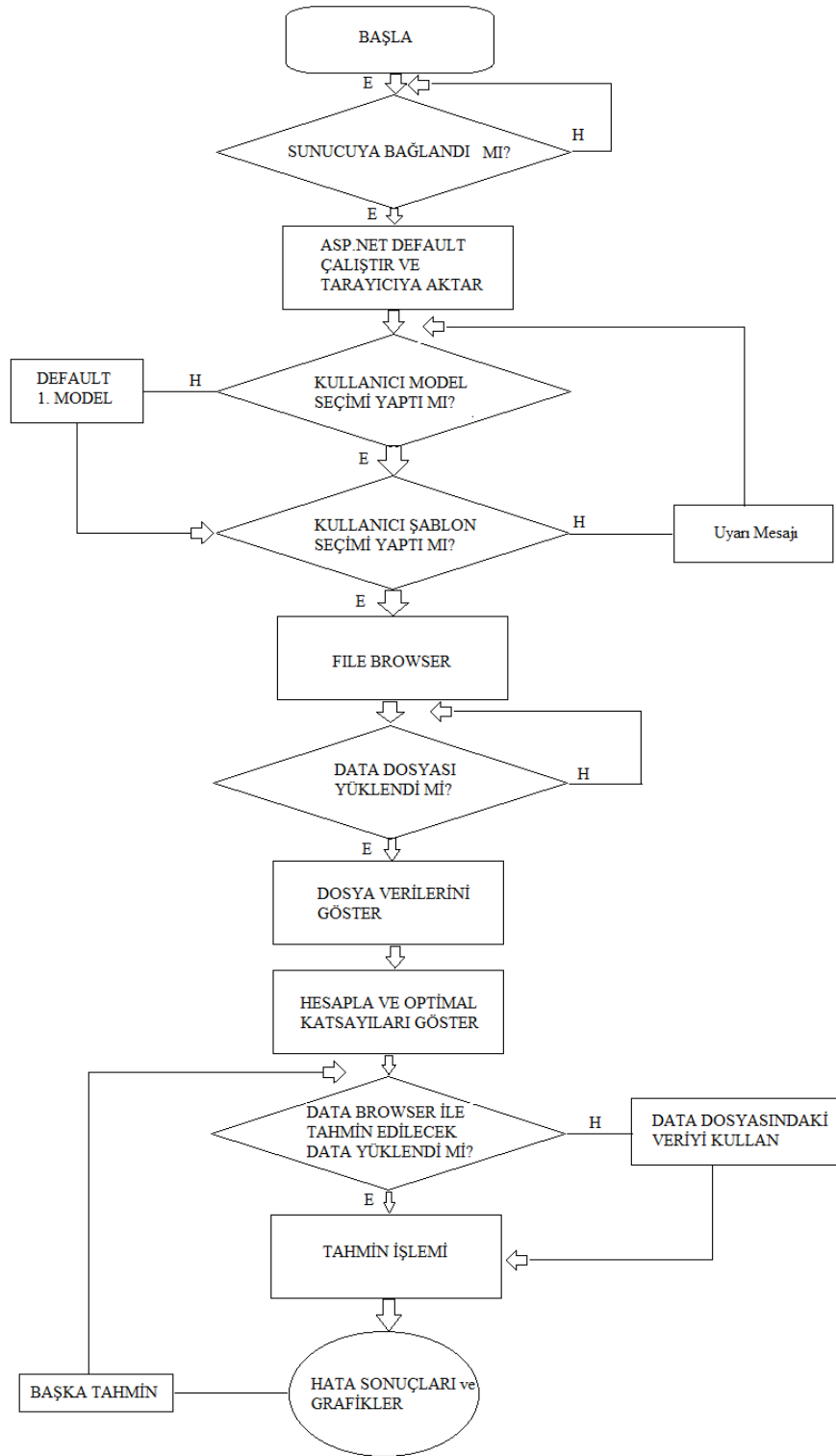
Çalışmada önceki bölümlerde anlatılan yöntemler kullanılarak doğrusal tahmin filtreleri ile tahmin yapan bir kullanıcı arayüzü tasarımı yapılmıştır. Arayüz ASP.NET ortamı kullanılarak web tabanlı olması sağlanmıştır. Kullanıcının kullanım hatası yapmasını engelleyecek algoritmalar geliştirilmiştir.

Şekil 4.1’de yapılan programlamaya ait akış diyagramı verilmiştir. Uygulama tamamı ile ASP.NET araçları ile tasarlanmıştır. Açılış sayfasında filtre şablonu seçimi ve filtre modeli seçimi için Resim 4.1’de görülen ekran gelmektedir. Filtre şablonu pixellerin üzerine tıklanarak seçilmesi sağlanmıştır.



Resim 4.1 Uygulama açılış ekranı.

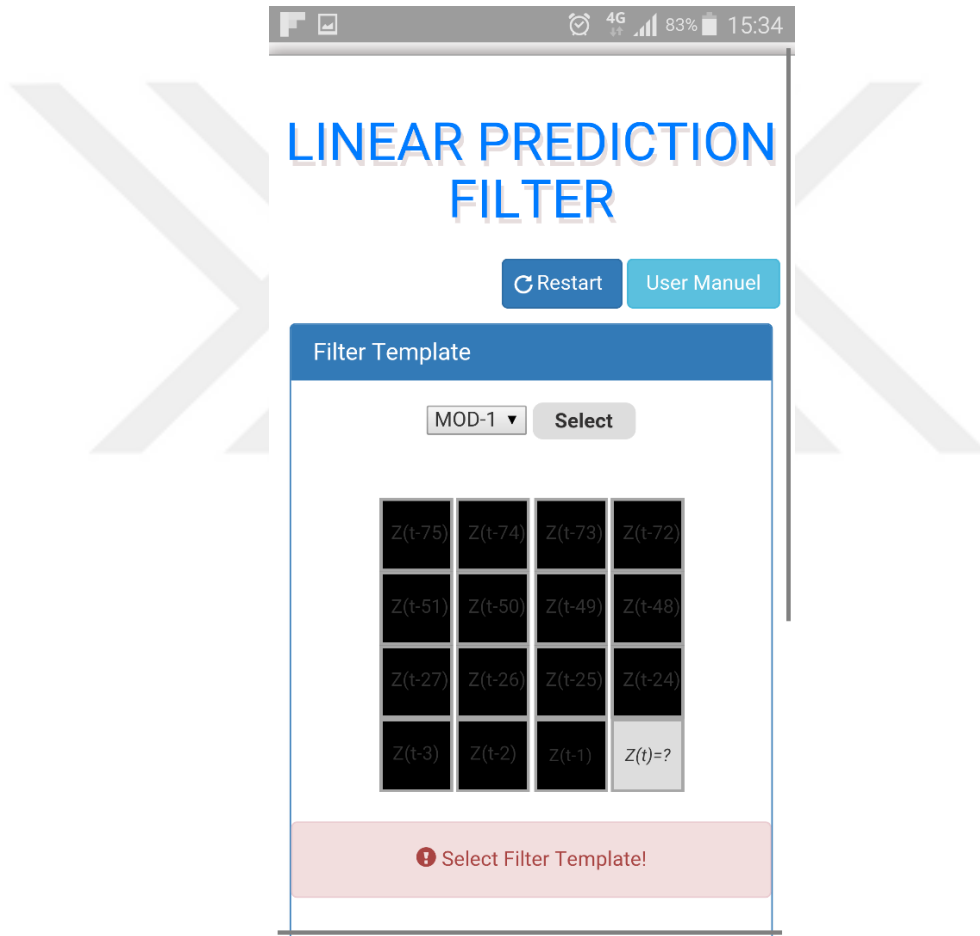
Tahmin edilecek datayla ilişkilendirilecek herhangi bir şablon seçilmezse Resim 4.2’deki hata mesajı yayınlanacaktır. Bu fonksiyonun yapımında CSS ve JavaScript birlikte kullanılmıştır. CSS ile mesajın görselliği oluşturulmuş JavaScript ile sorgulama fonksiyonu yazılmıştır. Resim 4.2’de görülen ekran mobil kullanımdan alınmıştır. Bunun için projede CSS Framework olan bootstrap kullanılmıştır.



Şekil 4.1 Program akış diyagramı.

4.1 Bootstrap Kullanımı

Bootstrap Front-End kullanıcılar için Twitter tarafından geliştirilmiş açık kaynak kodlu bir CSS Framework'tür. Bootstrap sayesinde tüm tarayıcılara uyumlu mobil veya masaüstü arayüzler geliştirilebilir. Web sitesi için gerekli bileşenleri bünyesinde barındırır (Form, tablo, buton, menü vb.). Resim 4.2'de görüldüğü gibi yapılan tasarım, mobil cihaz ekranlarına uyumlu hale getirilmiştir. Eklenen sütun bileşenleri ekranı kaplayacak şekilde alt alta hizalanmaktadır.



Resim 4.2 Hata mesajı (Mobil görünüm).

4.2 Postback Olayı ve Update Panel Kullanımı

Web sitelerinde bazı kontroller yardımıyla istemci tarafından bir talepte bulunulduğunda sayfa içeriği sunucuya gönderilir, istenilen talep sunucu tarafında

yapıldıktan sonra sayfa tekrar istemci tarafına gönderilir. ASP.NET platformunda uygulanan form işlemleriyle yapılan her bir istek sonrası sayfa yüklendiği duruma dönecektir. Bu durumun engellenmesi amacıyla projede ASP.NET ögesi Update Panel aracı kullanılmıştır. Update Panel aracı içerisinde yapılan işlemler sunucuya gönderilmez, istemci tarafında çalışır.

4.3 Model ve Şablon Seçimi

Model seçimi için 3 çeşit model tasarlanmıştır. Model seçimi yapılmadığında default olarak model-1'e göre işlem yapılacaktır. Eğer ikinci model seçilirse extraterrestrial datası hesaplanacak ve bu data tahmin katsayıları elde etmede kullanılacaktır. Bunun için modellenecek datanın alındığı mekânın enlem bilgisinin girilmesi gereklidir. MOD-2 seçildiğinde bir textbox aracı görünecektir.

The image shows a 'Filter Template' dialog box. At the top, there is a blue header bar with the text 'Filter Template'. Below the header, there are two radio buttons: 'MOD-1' (which is selected) and 'MOD-2'. To the right of these buttons is a 'Select' button. Below the radio buttons is a 4x4 grid of boxes. The boxes contain the following text: Row 1: Z(t-75), Z(t-74), Z(t-73), Z(t-72); Row 2: Z(t-51), Z(t-50), Z(t-49), Z(t-48); Row 3: Z(t-27), Z(t-26), Z(t-25), Z(t-24); Row 4: Z(t-3), Z(t-2), Z(t-1), Z(t)=?.

Resim 4.3 Model seçimi.

Model-3'te ise üç boyutlu doğrusal tahmin yapma imkanı sağlanmıştır. İstenirse iki boyutlu iki harici veri dizisi ile istenirse üç boyutlu üç harici veri dizisi ile modelleme yapmaktadır. Bunun için istenilen formatta dosyanın yüklenmesi gerekmektedir. Ayrıntılı bilgi bölüm 4.4'te verilmiştir. Model-2 ve Model-3 ile yapılan modellemede şablonda seçilen pixel sayısının iki veya üç katı kadar katsayı üretmelidir. Çünkü aynı

şablon diğer resimlerin de üzerinde gezdirilecek buradan da aynı indislerden katsayı üretecektir. Filtre seçimini tamamlamak için pixel şablonundaki üstünde yer alan “select” butonuna tıklanmalıdır. Tıklandığı takdirde modele göre extraterrestrial hesaplandı ise saatlik olarak görünecektir.

Resim 4.4’de MOD-2’de seçilmiş bir filtre şablonu, Resim 4.5’de ise Afyonkarahisar ili için ($\Phi=38.45$) hesaplanmış ve ekranda liste şeklinde gösterilmiş extraterrestrial ışınım verisi saatlik olarak görünmektedir. Örnek olarak değerlerin başındaki 1.2.3’ değeri birinci ayın ikinci gününün üçüncü saatindeki hesaplanan extraterrestrial ışınımıdır. Bu model ile sadece güneş ışınım tahmini yapılabilir. Diğer modellerden farklı olarak güneş ışınımının olmadığı zamanlara göre (extraterrestrial referans alınarak) tahminin sıfırlanması adına algoritma geliştirilmiştir.

Filter Template

MOD-2 Select

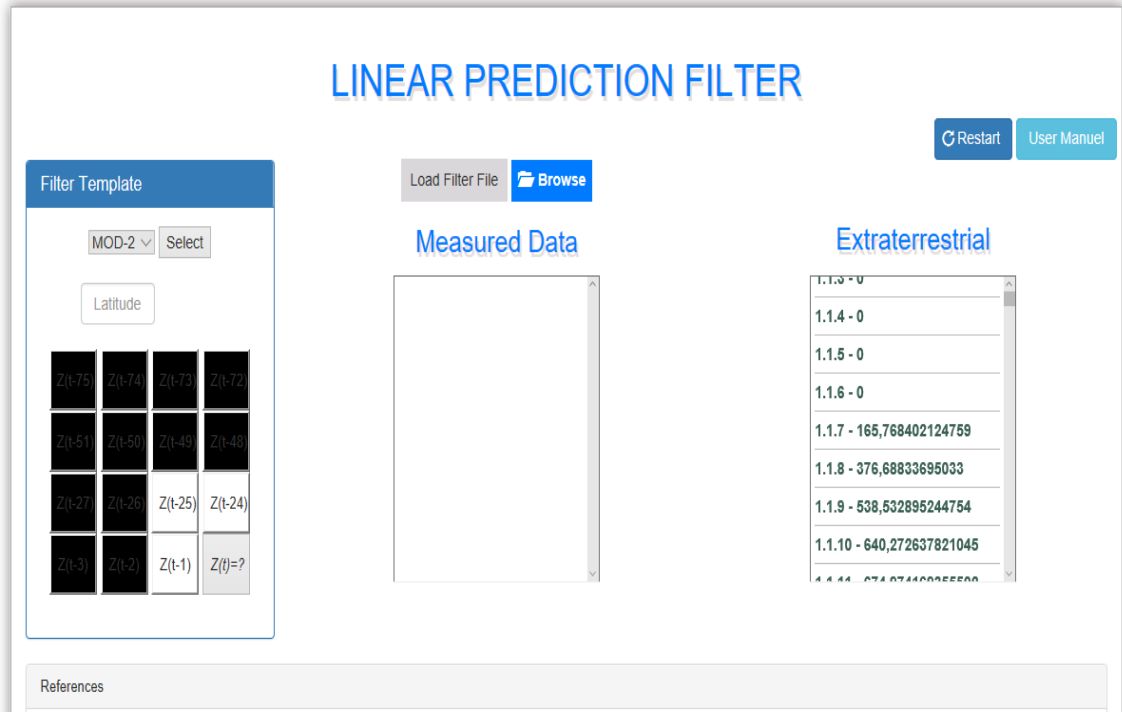
Latitude

$Z(t-75)$	$Z(t-74)$	$Z(t-73)$	$Z(t-72)$
$Z(t-51)$	$Z(t-50)$	$Z(t-49)$	$Z(t-48)$
$Z(t-27)$	$Z(t-26)$	$Z(t-25)$	$Z(t-24)$
$Z(t-3)$	$Z(t-2)$	$Z(t-1)$	$Z(t)=?$

Resim 4.4 Şablon seçimi.

Resim 4.4’de görülen ekranda “select” butonuna tıklandığında FileUpload aracı ve data listeleri görünecektir. MOD-2 seçildiğinden dolayı extraterrestrial hesabı Resim 4.5’de sağda gösterilmiştir. Hesaplanan extraterrestrial ışınım verisi yıllık olarak text formatında vektörel formda indirilebilmektedir. Bu ekrandan “browse” tıklanarak

modellenecek veri dosyasının yüklenmesi gerekmektedir.



Resim 4.5 Extraterrestrial Işınım hesabının uygulamada liste olarak gösterimi.

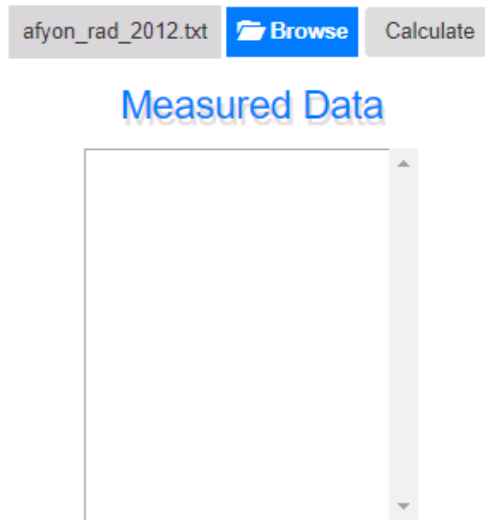
4.4 Dosya Yükleme

İkinci adımda FileUpload aracı ve data listesi gelecektir. Kullanıcı, bilgisayarında kayıtlı modellenecek olan güneş ışıınım data dosyasını yüklemesi gerekecektir. Yüklenmesi gereken dosya her bilgisayarda çalışabilmesi açısından metin belgesi formatında Resim 4.6’de görüldüğü şekilde olmalıdır. Eğer bu noktada hata yapıldıysa program ilerlemeyecektir. Dosyanın text formatında vektörel olarak, şekildeki gibi yazılması gerekmektedir. Vektörün sonuna “*ENDOF ALL*” imi yazılmalıdır. Eğer MOD-3 ile işlem yapılacaksa vektörel formda kaydedilen veri dizileri arasında “*ENDOF 1*” ve “*ENDOF 2*” imleri konularak, en sona “*ENDOF ALL*” imi eklenmelidir. İki boyutlu yani iki veri dizisi kullanılacaksa, diziler arasında “*ENDOF 1*” ikinci dizinin arkasına “*ENDOF ALL*” konularak dosya kaydedilmelidir. Üç boyutlu tahmin yapılacaksa birinci ile ikinci veri dizileri arasında “*ENDOF 1*”, ikinci ve üçüncü veri

dizleri arasına “*ENDOF 2*”, son veri dizisinin sonuna da “*ENDOF ALL*” imleri yazılmalıdır. Ayrıca veri dizilerinin boyutları da (eleman sayısı) eşit sayıda olmalıdır. Bu şekilde yapılmadığı takdirde program ilerlemeyecektir. Dizilerden ilki her zaman tahmin edilecek veri olmalıdır.

```
4.2783333e+000
5.2333333e+000
5.9766667e+000
5.6816667e+000
7.3600000e+000
7.4383333e+000
1.8916667e+001
4.0288333e+001
7.7715000e+001
1.5295500e+002
2.3162500e+002
2.1909500e+002
1.4164333e+002
6.4183333e+001
1.5390000e+001
7.5333333e+000
7.0200000e+000
7.6900000e+000
8.8166667e+000
9.6500000e+000
9.6233333e+000
1.2598333e+001
1.4508333e+001
0.0000000e+000
0.0000000e+000
0.0000000e+000
0.0000000e+000
0.0000000e+000
0.0000000e+000
```

Resim 4.6 Örnek güneş ışınlam verisi.



Resim 4.7 Veri dosyasının yüklenmesi.

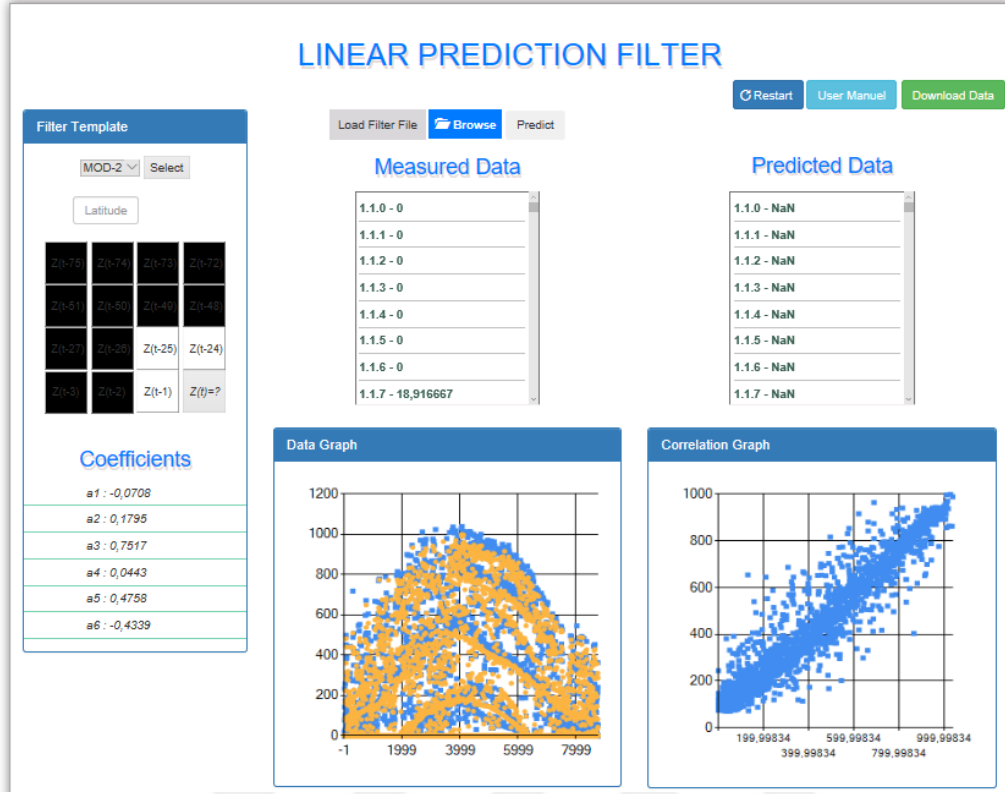
Eğer formata uygun dosya yüklendiyse dosyadan alınan veriler liste halinde

gösterilecek ve tahmin modeli üzerinde hesaplamaların yapılması amacı ile Resim 4.7’deki “calculate” isimli buton görünecektir. Butona tıklandığında seçilen şablona göre tahmin modeli oluşturulacak, şablon veri üzerinde gezdirilecek ve optimal katsayılar hesaplanıp Resim 4.8’deki şekilde görünecektir.

Resim 4.8 Hesaplanmış optimal katsayılar ve güneş ışınlam verisi.

4.5 Tahmin İşlemi ve Hata Sonuçlarını İzleme

Tahmin modeli eğitim verisi kullanılarak oluşturulduktan sonra, bu model ile istenirse eğitim datası ile yapılan tahminin hatası hesaplanabilmekte, istenirse farklı bir test verisinin tahmini yapılabilmektedir. Bunun için farklı bir test verisinin yüklenmesi gerekecek, yüklenmediği takdirde eğitimde kullanılan veri test verisi olarak kullanılacaktır. Bu aşamada ekranda “calculate” butonu “predict” butonuna dönüşmektedir. “Predict” butonuna tıklandığında yukarıda anlatıldığı şekilde modelden çıkarılan hata değerleri görünecek, asıl veri ile tahmin edilen veri arasındaki korelasyon, ekranın altında görünecek olan grafikte görüntülenecektir.



Resim 4.9 Tahmin işlemi sonrası ekran görüntüsü.

Resim 4.9’da görüldüğü üzere gezdirilen filtre şablonu, yüklenen - tahmin edilen dataların grafiksel gösterimi ve aralarındaki ilişkinin korelasyon grafiği tek bir ekranda gösterilmesi sağlanmıştır. Tahmin edilen datanın “NaN” ifadesiyle gösterilmesinin sebebi, belirtilen zamana ait geçmiş zaman verisi bulunmadığı için, bu zamanlar tahmin edilmemiştir. Aynı zamanda hata hesabına da katılmamıştır.

Tahmin işlemi sonrasında ekranın sağ üst köşesinde “Download Data” butonu görünecektir. Buradan tahmin edilen veri vektörel formda metin belgesi olarak indirilebilmektedir.

Error Analysis
<i>MSD :4244,94911352672</i>
<i>RMSE :65,1532740046632</i>
<i>MBE :-3,62759155089066</i>
<i>MAPE :24,8558499050183</i>

Resim 4.10 Hata sonuçları.



5. UYGULAMALAR

Arayüz muhtelif güneş ışınlam verilerinin farklı filtre şablon ve modellerle tahmin işlemleri ile test edilmiştir. Yapılan testlerde önceki yıllarda yapılan çalışmalara benzer sonuçlar alınmıştır. Akarslan vd. (2014) yaptığı çalışmada, farklı şablonlarda ve farklı modellerde yaptığı testlerde, RMSE kriterine göre en uygun sonucun ilişkisel daha çok datanın yer aldığı modelde sağlandığını tespit etmiştir.

Programda MOD-1 seçildiğinde lineer özellik gösteren her çeşit data, veri sayısına bakılmaksızın tahmin edilebilir (Rüzgar, basınç vb.). Önceki çalışmalardan farklı olarak MOD-2 ile extraterrestrial ışınlam verisi hesabı ile sadece yıllık saatlik bazda güneş ışınlam tahmini yapılmasına uygun algoritma geliştirilmiştir. Burada güneşin doğuş ve batış saatleri baz alınarak, bu saatler dışında ölçülen veya tahmin edilen veri “0” olarak kabul edilmiştir. Bu düzeltme ile yapılan tahminin toplam hatası daha düşük çıkmaktadır. Örneğin güneşin olmadığı zamanlarda herhangi bir ışık kaynağından (ay, sokak lambası, far vb.) yayılan ışık, ölçüm cihazlarında değer vermektedir. Bu değerler tahmin edilen veri güneş ışınlamı ise hesaba katılmamalıdır. Ayrıca güneşin olmadığı zamanlarda (güneşin ilk battığı saatlerde) geçmiş değerler kullanılarak tahmin yapılmamalıdır. Bunun için programda MOD-2 seçilmelidir. MOD-2’de kullanılacak eğitim verisinin 8760 elemanlı olmasına dikkat edilmelidir.

MOD-3 seçildiğinde ise herhangi periyotta üretilmiş istenilen sayıda eleman içeren verileri iki veya üç boyutlu olarak tahmin modeli geliştirilebilmektedir. MOD-3 veri dizilerinin birbirileri ile ilişkilerini sorgulamada olumlu sonuçlar vermektedir.

5.1 Arayüz ile Güneş Işınım Tahmini

Yapılan tasarımda geçmişe ait en çok katsayı kullanan şablon Resim 5.1’ de görünmektedir. Bu şablonda üç güne kadar önceki veriler kullanılacaktır. O yüzden tahmin dördüncü günün dördüncü saatinden başlayarak yapılacaktır. Ekranda görülen 38.45 değeri Afyonkarahisar ilinin enlem bilgisidir. Enlem bilgisi girilmediği takdirde, sistem hesabın Afyonkarahisar iline göre yapılacağını varsayar ve bu şehire ait

Extraterrestrial ışıınım verilerini hesaplayıp gösterecektir.

The image shows a 'Filter Template' window. At the top, there is a dropdown menu currently showing 'MOD-2' and a 'Select' button. Below this is a text input field containing the number '38.45'. At the bottom, there is a 4x4 grid of boxes, each containing a time-lagged variable: $Z(t-75)$, $Z(t-74)$, $Z(t-73)$, $Z(t-72)$ in the first row; $Z(t-51)$, $Z(t-50)$, $Z(t-49)$, $Z(t-48)$ in the second row; $Z(t-27)$, $Z(t-26)$, $Z(t-25)$, $Z(t-24)$ in the third row; and $Z(t-3)$, $Z(t-2)$, $Z(t-1)$, $Z(t)=?$ in the fourth row.

Resim 5.1 Güneş ışıınım tahmininde kullanılan filtre şablonu.

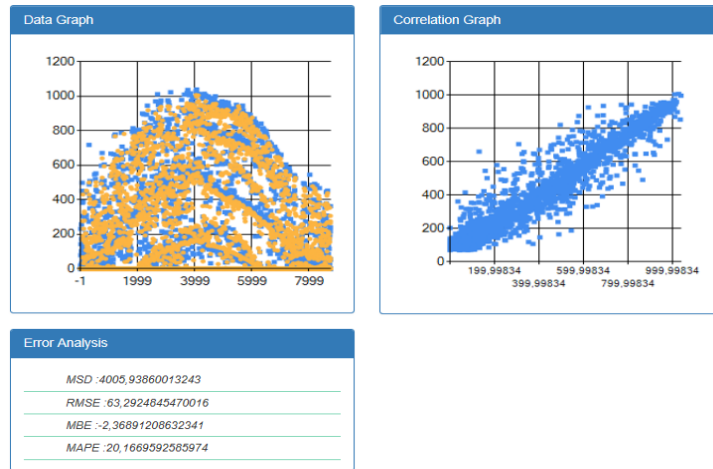
Bu şablon ile ve MOD-2 modeline göre extraterrestrial verisinin de yönteme katılarak Afyonkarahisar iline ait 2012 yılı güneş ışıınım verisi modellenmiştir. Oluşturulan tahmin modeli ile 2012 yılına ait saatlik güneş ışıınım değerleri tahminine ait sonuçlar Resim 5.2’de verilmiştir. Oluşturulan model ile 2013 yılına yılına ait saatlik güneş ışıınım değerleri tahmin edilmiş ve tahmin sonuçları Resim 5.3’de verilmiştir. Buna göre extraterrestrial ışıınım verisinin modelde etkileşimi ile güneş ışıınım verisinin yıllara göre çok az farklılık içermesinden dolayı RMSE değerlerinde çok az fark oluşmuştur. Çıkarılan model ile 2012 yılına ait verilerle yapılan işlemde RMSE’ye göre tahmin hatası daha iyi sonuçlar verirken 2013 yılına ait verilerle yapılan işlemde MAPE’ye göre tahmin hatası daha iyi sonuçlar vermiştir.

Çizelge 5.1 Uygulamada hesaplanan optimal katsayılar.

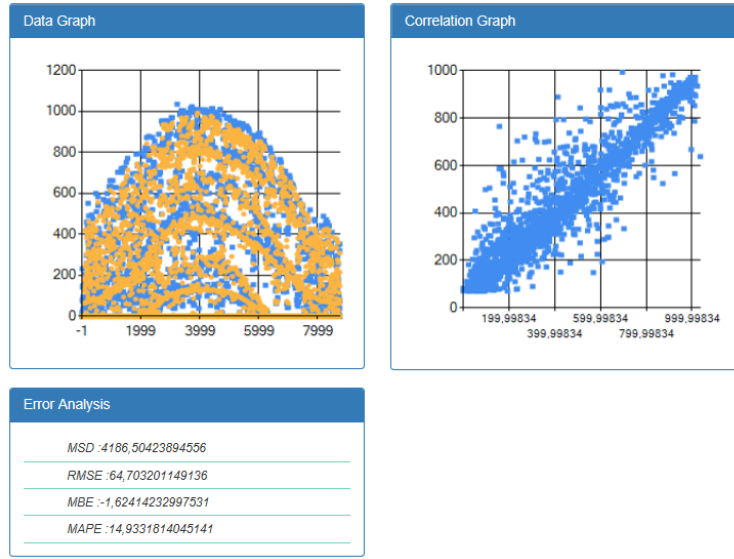
Katsayı	Değer	Açıklama
A1	0,0291	Tahmin edilenin üç gün üç saat öncesi
A2	-0,0461	Tahmin edilenin üç gün iki saat öncesi
A3	-0,0519	Tahmin edilenin üç gün bir saat öncesi
A4	0,1021	Tahmin edilenin üç gün öncesi
A5	0,0492	Tahmin edilenin iki gün üç saat öncesi
A6	0,0262	Tahmin edilenin iki gün iki saat öncesi

Çizelge 5.1 (Devamı) Uygulamada hesaplanan optimal katsayılar.

Katsayı	Değer	Açıklama
A7	-0,016	Tahmin edilenin iki gün bir saat öncesi
A8	0,0687	Tahmin edilenin iki gün öncesi
A9	-0,0529	Tahmin edilenin bir gün üç saat öncesi
A10	-0,0184	Tahmin edilenin bir gün iki saat öncesi
A11	-0,0038	Tahmin edilenin bir gün bir saat öncesi
A12	0,1271	Tahmin edilenin bir gün öncesi
A13	0,0176	Tahmin edilenin üç saat öncesi
A14	-0,0756	Tahmin edilenin iki saat öncesi
A15	0,7987	Tahmin edilenin bir saat öncesi
A16	-2,2599	Extraterrestrial verisi üç gün üç saat öncesi
A17	-5,4609	Extraterrestrial verisi üç gün iki saat öncesi
A18	-7,7876	Extraterrestrial verisi üç gün bir saat öncesi
A19	-6,9753	Extraterrestrial verisi üç gün öncesi
A20	4,3167	Extraterrestrial verisi iki gün üç saat öncesi
A21	5,7759	Extraterrestrial verisi iki gün iki saat öncesi
A22	8,3217	Extraterrestrial verisi iki gün bir saat öncesi
A23	13,9873	Extraterrestrial verisi iki gün öncesi
A24	-1,9735	Extraterrestrial verisi bir gün üç saat öncesi
A25	5,5057	Extraterrestrial verisi bir gün iki saat öncesi
A26	6,3415	Extraterrestrial verisi bir gün bir saat öncesi
A27	-6,6801	Extraterrestrial verisi bir gün öncesi
A28	-0,0565	Extraterrestrial verisi üç saat öncesi
A29	-5,8258	Extraterrestrial verisi iki saat öncesi
A30	-7,1471	Extraterrestrial verisi bir saat öncesi



Resim 5.2 Afyonkarahisar ili 2012 yılı güneş ışınlam verileriyle MOD-2 ile modellenen 2012 yılı verileri tahmini.



Resim 5.3 Afyonkarahisar ili 2012 yılı güneş ışınlam verileriyle MOD-2 ile modellenen 2013 yılı verileri tahmini.

Arayüz böylelikle herhangi bir eğitim verisi kullanılarak belirlenen katsayılar ile farklı bir verinin tahmin edilmesine olanak sağlarken, aynı zamanda hata analiz yöntemleriyle oluşturulan modelin sorgulanmasına da olanak sağlamaktadır.

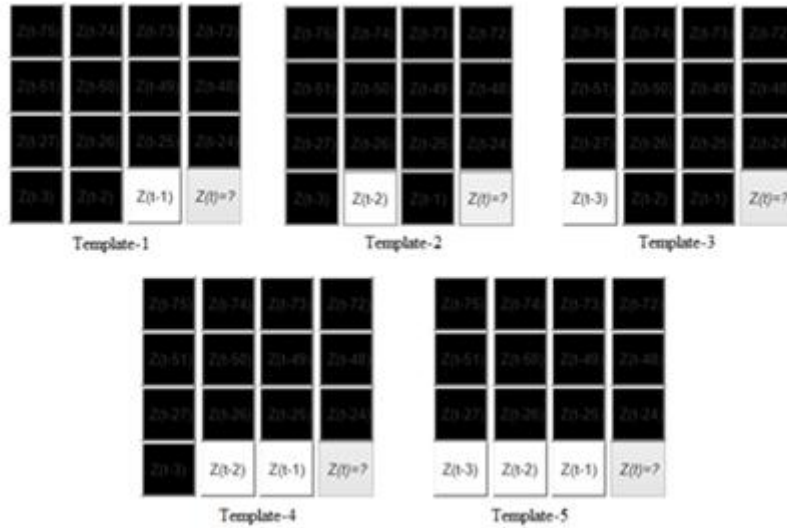
Yapılan uygulamaya ek olarak test verisi ile test verisinden tahmin edilen değerleri karşılaştırmak dışında, güneş ışınlamının bir önceki yılın verisi ile tahmin edilerek bulunan yılın gerçek verileriyle karşılaştırılması adına bir çalışma yapılmıştır. Böyle bir modelde gerçekte nasıl bir hata ile karşılaşacağı fikrinden yola çıkılarak, Çizelge 5.1’de sunulan model katsayıları ile 2012 yılı verileri tahmin edilerek tahmin değerleri arayüzden indirilmiştir. Arayüzden alınan saatlik tahmin edilen veriler ile 2013 yılına ait saatlik gerçek güneş ışınlam verileri uygulama dışında oluşturulan hata hesaplama algoritması ile hesaplanmıştır. Aynı filtre şablonu kullanılarak MOD-1’de de aynı işlem yapılmıştır. Buna göre 2013 yılında olması beklenen değerler, gerçek değerler ile karşılaştırıldığında Çizelge 5.2’de verilen hata değerleri hesaplanmıştır. MOD-2’de veri bir bütün olarak düşünüldüğünde yani bir yıllık toplamda $8760 \times 1,71 \text{ W/m}^2$ kadarlık hata ile tahmin edilmiştir. 8760 değeri yıllık tahmin edilen veri sayısıdır. Yapılan testte MAPE değeri %30,83 olarak hesaplanmıştır. MOD-1’de yapılan tahminde ise bütün kriterler MOD-2’ye göre daha başarısız sonuçlar vermiştir. Burada extraterrestrial ışınlam verisinin yıllık toplam güneş ışınlam tahminine katkısı görünmektedir.

Çizelge 5.2 Güneş ışınlam tahmin değerleri ile gerçek değerler arasındaki hata.

	RMSE [W/m ²]	MBE [W/m ²]	MAPE [%]
MOD-1	130,26	6,79	31,12
MOD-2	124,05	1,71	30,83

5.2 Arayüz ile Kısa Dönem Rüzgar Hızı Tahmini

Afyon Kocatepe Üniversitesi kampüsüne kurulan meteorolojik istasyondan alınan 23 Mart-4 Mayıs 2018 tarihleri arasına ait 10’ar dakikalık rüzgar hızı verisi Resim 5.4’ de görülen filtre şablonları ile tahmin edilmiştir.



Resim 5.4 Rüzgar hızı tahmininde kullanılan filtre şablonları.

Birinci şablonda tahmin edilecek rüzgar hız verisinin 10 dakika öncesi, ikinci şablonda 20 dakika öncesi, üçüncü şablonda 30 dakika öncesi, dördüncü şablonda 10 dakika ve 20 dakika öncesi birlikte, beşinci şablonda ise 10 dakika, 20 dakika ve 30 dakika öncesine ait veriler birlikte kullanılmıştır. Şablonlar sadece rüzgar verisi üzerine uygulanmış yani MOD-1 seçilmiştir. Bu şablonlar ile yapılan tahminin hata sonuçları Çizelge 5.3’de verilmiştir.

Çizelge 5.3 Rüzgar hızı tahmin değerleri ile gerçek değerler arasındaki hata.

Filtre Şablonu	RMSE [m/s]	R²
Template-1	0,6041	0,8474
Template-2	0,8015	0,7339
Template-3	0,9246	0,6485
Template-4	0,6026	0,8479
Template-5	0,6011	0,8485

Hata sonuçlarından da anlaşıldığı gibi filtre şablonları arasında en başarılı tahmin 5 numaralı şablon ile yapılmıştır. Yapılan bu uygulamada, Güneş ışınlamı ile yapılan uygulamadan farklı olarak 10 dakika periyotlarla kaydedilmiş rüzgâr verileri kullanılmıştır. Böylece arayüzün veri sayısına, türüne (rüzgar, güneş, herhangi bir zaman serisi vs.), periyot vb.'ine bakılmaksızın, gerçekleştirilecek tahmin uygulamalarında kullanılabileceği gösterilmiştir.

Ekranın sağ üst köşesinde yer alan “user manuel” butonuna tıklandığında uygulamanın kullanım kılavuzu açılmaktadır. Programın kullandığı metotların açıklaması ve uygulamanın kullanımı ile kılavuz incelenebilir. Yine yanında yer alan “restart” butonu ile yeni bir simülasyon ile tahmin işlemi başlatılabilir.

6. DEĞERLENDİRME ve SONUÇ

Doğrusal tahmin filtreleri kullanılarak kolayca tahmin uygulamaları yapabilen kullanıcı dostu bir web arayüz tasarlanmıştır. ASP.NET ortamında .NET platformunun sunduğu kolaylıklardan faydalanılmıştır. Uygulamada farklı filtre şablonları tasarlanabilmekte, farklı verilerin veri sayısına bakılmaksızın modellenerek tahmin işlemi yapılabilmekte ve eğitim verisi kullanılarak çıkarılan model ile farklı test verileri de tahmin edilebilmektedir. Görüntü işleme modellerinde de programda kullanılan algoritmaya benzer komşu pixellerin esas pixelle ilişkilendirme mantığına dayanan farklı filtreleme yöntemleri kullanılmaktadır. Arayüz aynı zamanda ilişkisel veri dizilerinin eğitim verisine dahil edilmesi ile çok boyutlu filtreleme işlemi yapılarak daha başarılı tahminlerin gerçekleştirilmesine de olanak sağlamaktadır. Bu özellikle birlikte herhangi bir veri dizisinin başka veri dizileri ile ilişkisi de sorgulanabilmektedir.

Program web tabanlı bir uygulama olarak kullanıcıların her mekândan işlem yapmalarına olanak sağlamakta, mobil uyumluluğu sayesinde mobil aygıtlarda da sorunsuz çalışabilmektedir. Kullanıcı tasarlanan web arayüz sayesinde kolayca bu işlemleri yapabilecek, kendi verilerini ve yöntemi sorunsuzca test edebilecek, sonuçlarını alabilecektir. Yapılan hata hesabı ile gelecek zamanın tahminini daha net bir şekilde yapabilecektir. Ayrıca programda kullanılan yöntemi kendi yöntemleriyle test ederek karşılaştırma yapabilecek. Bu çalışma ile bu alanda yapılan çalışmaların karşılaştırılması, yorumlanması birbiriyle etkileşimini hızlandırılmasına katkı sağlanması amaçlanmıştır.

Yazılımın geliştirilmesi ile gerek farklı modellerin oluşturulması, gerekse anlık verinin, iterasyon benzeri yöntemler ile dinamik olarak canlı tahmin algoritması üzerinde çalışmalar yapılabilir. Bu şekilde anlık olarak bir zaman sonrasının tahmin edilerek görüntülenmesi sağlanabilir. Tahmin işleminde örneğin, güneş ışınımına etkisi olan bulutluluk, nem, sıcaklık vb. verilerin filtreleme yöntemine modellenerek dahil edilmesi ile daha başarılı sonuçlar alınması mümkün olabilecektir.

Programa [lpf.aku.edu.tr](http://pf.aku.edu.tr) web adresinden ulaşım sağlanmaktadır.

7. KAYNAKLAR

- Akarslan, E., (2014). Güneş ışınlam tahmininde sinyal işleme tabanlı yeni yaklaşımlar. Doktora Tezi, Osmangazi Üniversitesi
- Akarslan, E., Hocaoglu, F.O. (2016). A novel adaptive approach for hourly solar radiation forecasting. *Renewable Energy*. **87**: 628–633.
- Akarslan, E., Hocaoglu, F.O. (2016). A novel method based on similarity for hourly solar irradiance forecasting. *Renewable Energy*. **112**: 337–346.
- Akarslan, E., Hocaoglu, F.O., Edizkan, R. (2014). A novel M-D (multi-dimensional) linear prediction filter approach for hourly solar radiation forecasting. *Energy*. **73**: 978–986.
- Aytekin, Ö.Ç, (2009). Web 2.0'ın tasarıma getirdiği yeni yaklaşımlar doğrultusunda web standartlarına uygun uzaktan eğitim portalı tasarımı, Yüksek Lisans Tezi
- Bounnady, K., Phanthavong, K., Pathoumvanh, S., Sihlath, K. (2016). Comparison the processing speed between PHP and ASP.NET. 13th International Conference on Electrical Engineering, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Mai, Thailand, 28 June-01 July, 1-5
- Bacher, P., Madsen, H., Nielsen, H.,A., (2009). Online short-term solar power forecasting. *Solar Energy*. **83**: 1772–1783.
- Chena, R., Kanga, E., Jia, X., Yanga, J., Wang, J. (2007). An hourly solar radiation model under actual weather and terrain conditions: a case study in Heihe river basin. *Energy*, **32**: 1148–1157.
- Cucumo, M., De Rosa, A., Ferraro, V., Kaliakatsos, D., Marinelli, V. (2007). Experimental testing of models for the estimation of hourly solar radiation on vertical surfaces at Arcavacata di Rende. *Solar Energy*, **81**: 692–695.
- Hocaoglu, F.O, Gerek, O.N., Kurban, M. (2008). Hourly solar radiation forecasting using optimal coefficient 2-D linear filters and feed-forward neural networks. *Solar Energy*, **82**: 714-726
- Hocaoglu, F.O, Gerek, O.N., Kurban, M. (2009). Novel models for hourly solar radiation using a 2-D approach. *International Journal of Innovative Computing, Information and Control*. **5**: 855-864
- Jinglin, M., (2011), Design and implementation of distance teaching platform based on ASP.Net. *Energy Procedia*, **13**: 7281-7287.

- Kang, B., Yoo, G. (2007). Efficient Voice User Interface System Using VoiceXML and ASP.NET 2.0. *Advanced Parallel Processing Technologies* **4847**: 608–616.
- Kaplanis, S., Kaplani, E. (2007). A model to predict expected mean and stochastic hourly global solar radiation I values. *Renewable Energy* **32**: 1414–1425.
- Kharseh, M., Solar Radiation Calculations. Course Notes.
- Reikard, G., (2009). Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy*. **83**: 342–349.
- Roman, T.V., (1997). Cascaded speech style sheets. *Computer Networks and ISDN Systems*, **29**: 1377-1383.
- Mansour, N., Hourri, M. (2006). Testing web applications, *Information and Software Technology*. **48**: 31–42.
- Mijailovic, Z., Milicev, D. (2014). Emprical analysis of GUI programming concerns. *International Journal of Human-Computer Studies*. **72**: 757–771.
- Ustundag E., Hocaoglu F.O., Akarslan E., Serttas F.(2018). Short Term Wind Speed Forecasting using Linear Prediction Filters. IIER International Conference, Crete, Greece, 8-9 July, 9-12

İnternet Kaynakları

- 1) <http://www.maxdesign.com>, 01.01.2017
- 2) <http://solardat.uoregon.edu/SolarRadiationBasics.html>, 01.01.2017
- 3) <https://www.w3schools.com/>, 27.11.2017
- 4) <https://www.w3.org>, 27.11.2017



ÖZGEÇMİŞ

Adı Soyadı : Emrah ÜSTÜNDAĞ
Doğum Yeri ve Tarihi : Akkonak-29.04.1986
Yabancı Dili : İngilizce
İletişim (Telefon/e-posta) : 0 506 810 09 72

Eğitim Durumu (Kurum ve Yıl)

Lise : Afyon Süleyman Demirel Fen Lisesi, (2001-2004)
Lisans : Ege Üniversitesi, Elektrik-Elektronik Mühendisliği
Bölümü, (2004-2009)
Yüksek Lisans : Afyon Kocatepe Üniversitesi, Fen Bilimleri
Enstitüsü, Yenilenebilir Enerji Sistemleri, (2016-
2018)

Yayınlar/Çalışmalar : -Manyetik Geçirgenlik Ölçer-(Histerezis) (Ege
Üniversitesi-2009)
-Enerji Nakil Hatları Proje Hazırlama Yazılımı
(2017)
-Short Term Wind Speed Forecasting using Linear
Prediction Filters, IJACEN V:6,I:9 (2018)
-Very Short Term Wind Speed Forecasting with
ANN, Innovation and Global Issues in
Multidisciplinary Sciences IV, (2018)
-Çok Boyutlu Doğrusal Tahmin Filtreleri ile Kısa
Dönem Rüzgar Hızı Tahmini, Uluslararası
Bilimsel Araştırmalar Kongresi (2018)

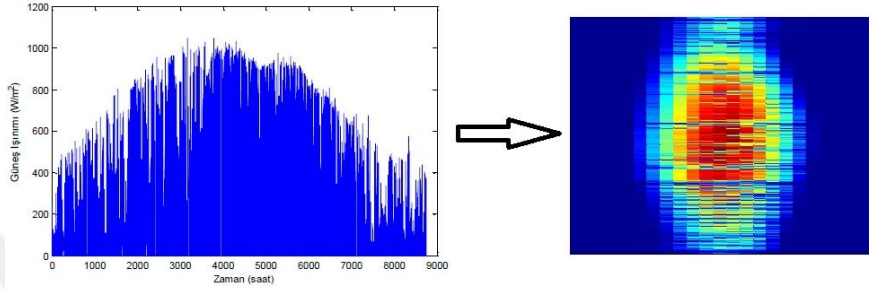
Çalıştığı Kurum/Kurumlar ve Yıl : OEDAŞ 2010-2018

EKLER

EK 1. Kullanım Kılavuzu

USER MANUEL

The application is an interface that predicts with the linear filter method. Any kind of annual hourly data can be predicted by image processing linear filter method with desired filter templates. You can analyze the prediction with using error methods and you can compare the success of the method with predictions made with other methods.



Firstly, the data is converted to a two-dimensional image format. The selected template is filtered on the image. For example a filter template is shown below,

Z_{ij}	$Z_{i,j+1}$
$Z_{i+1,j}$	$\check{Z}_{i+1,j+1}=?$

To calculate filter coefficients(a_1, a_2, a_3) with formula;

$$\check{Z}_{i+1,j+1} = a_1 Z_{ij} + a_2 Z_{i,j+1} + a_3 Z_{i+1,j}$$

Here i day, j time and $Z_{i,j}$ expresses the value of solar radiation at the i . day and j . hour time.

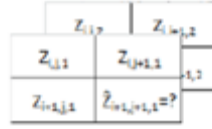
In the linear forecasting, each past sample in the shifted template is sized with the appropriate coefficients determined to minimize the quadratic prediction error.

$$\frac{\partial \epsilon}{\partial a_1} + \frac{\partial \epsilon}{\partial a_2} + \frac{\partial \epsilon}{\partial a_3} = 0$$

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

Where r_{ij} represents the correlation between the z_i and z_j pixels, and r_k represents the correlation between the predicted pixel and each pixel in the filter template. If another data is used for prediction,

EK 1. (Devamı) Kullanım Kılavuzu



$$\hat{Z}_{i+1,j+1,1} = a_1 Z_{i,j,1} + a_2 Z_{i,j+1,1} + a_3 Z_{i+1,j,1} + a_4 Z_{i,j,2} + a_5 Z_{i,j+1,2} + a_6 Z_{i+1,j,2}$$

How to Use?

Firstly, the filter template on the opening screen is selected by clicking on the black boxes. If you click on any black box what is past value of present value that will be predicted, black box turn into white. For example, when you select the box $Z(t-25)$, the forecasting model will be developed by using the data of the one day and one hour before the current time. You can choose model from 3 developed models. First model which is called MOD-1 presents one dimensional linear prediction with any data which has any quantity.

If the template is selected, the file upload tool and data list will be available. The file you need to load must be .txt extension. If you use MOD-2 file must contain 8760 (365x24) piece of data as shown below. Data file must finish with "ENDOF ALL" sign. If you use MOD-3, between the data series you must "ENDOF 1" or "ENDOF 2" signs. If you will predict with two dimensional filter, there must be "ENDOF 1" between two data series. If three dimensionan filter will be used, there must be "ENDOF 2" between second and third data series. Even so data file must end with "ENDOF ALL" sign.

```
5.6816667e+000
7.3600000e+000
7.4383333e+000
1.8916667e+001
4.0288333e+001
7.7715000e+001
1.5295500e+002
```

You can see the data that you have imported on the data list. To do this, click the "calculate" button. When you click the button, data will be modelled and predicted. Then coefficients will be listed under the filter template. How many coefficients are selected from the template are sorted like $a_1, a_2 \dots$ left to right then top to bottom.

Once the optimal coefficients have been calculated, the "predict" button will appear next to the file uploader. If you click directly on the button, it will predict the data that you used in the education model. You can also predict with the same model by loading a different test data. When the "predict" button is clicked, the graphs and error calculation results will be displayed below. The data value shown as "NaN" in the predicted data list indicates that it is not predicted because there is no history in the template. In addition, the "save" button will appear on the upper right. You can download the predicted data with this button.

The forecast can be monitored and analyzed numerically and visually. Three different MODs have been developed. Any data can be filtered in the first and third MODs. In the second MOD, only solar radiation can be predicted and data which will be predicted must contain 8760 pieces. Here, extraterrestrial radiation data is calculated by entering the latitude information. In solar radiation forecasting two data are superimposed on

EK 1. (Devamı) Kullanım Kılavuzu

each other to obtain better results in the filtering process. When latitude information is not entered, the program defaults to 38.45 for Afyonkarahisar/Turkey.

In solar radiation forecasting with image processing lineer filter method, when there is no sun, method gives a prediction with coefficients and past values. This logic error is solved in the second MOD with the established algorithm. In this case, the total error is smaller.

You can send your offers and comments to ustundag_emrah@hotmail.com, fohocaoglu@gmail.com, ekarslan@aku.edu.tr

EK 2. Program Kodları

ASP.NET KODLARI(C#)

//Aşağıda verilen kodlara ilave “Filter” ve “Matrix” adında iki sınıf oluşturulmuştur.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Drawing;
using System.IO;

namespace LINEAR_PREDICTION_FILTER
{
    public partial class Default : System.Web.UI.Page
    {
        int[] b = new int[30];
        Dictionary<int, Label> kas = new Dictionary<int, Label>();
        Dictionary<int, int> aa = new Dictionary<int, int>();
        int fb = 75;
        double fider; // enlem

        double[] isinim = new double[24 * 365];
        double[] extra = new double[24 * 365];
        double[] preisinim = new double[24 * 365];

        protected void Page_Load(object sender, EventArgs e)
        {
            kas.Add(1, Label2);
            kas.Add(2, Label3);
            kas.Add(3, Label4);
            kas.Add(4, Label5);
            kas.Add(5, Label6);
            kas.Add(6, Label7);
            kas.Add(7, Label8);
            kas.Add(8, Label9);
            kas.Add(9, Label10);
            kas.Add(10, Label11);
            kas.Add(11, Label12);
            kas.Add(12, Label13);
            kas.Add(13, Label14);
            kas.Add(14, Label15);
            kas.Add(15, Label16);
            kas.Add(16, Label17);
            kas.Add(17, Label18);
            kas.Add(18, Label19);
            kas.Add(19, Label20);
```

```

kas.Add(20, Label21);
kas.Add(21, Label22);
kas.Add(22, Label23);
kas.Add(23, Label24);
kas.Add(24, Label25);
kas.Add(25, Label31);
kas.Add(26, Label32);
kas.Add(27, Label33);
kas.Add(28, Label34);
kas.Add(29, Label35);
kas.Add(30, Label36);
aa.Add(0, 75);
aa.Add(1, 74);
aa.Add(2, 73);
aa.Add(3, 72);
aa.Add(4, 51);
aa.Add(5, 50);
aa.Add(6, 49);
aa.Add(7, 48);
aa.Add(8, 27);
aa.Add(9, 26);
aa.Add(10, 25);
aa.Add(11, 24);
aa.Add(12, 3);
aa.Add(13, 2);
aa.Add(14, 1);

Panel3.Visible = false;
Panel1.Visible = false;
Panel6.Visible = false;
if (Page.IsPostBack == false)
{
    Panel2.Visible = false;
    Panel4.Visible = false;
    DropDownList1.Items.Add("MOD-1");
    DropDownList1.Items.Add("MOD-2");
    for (int i = 1; i < 31; i++)
        kas[i].Visible = false;
}
}
#region butonlar
protected void Button1_Click(object sender, EventArgs e)
{
    if (Button1.BackColor == Color.Black)
        Button1.BackColor = Color.White;
    else
        Button1.BackColor = Color.Black;
}

```

```

}
protected void Button2_Click(object sender, EventArgs e)
{
    if (Button2.BackColor == Color.Black)
        Button2.BackColor = Color.White;
    else
        Button2.BackColor = Color.Black;
}
protected void Button3_Click(object sender, EventArgs e)
{
    if (Button3.BackColor == Color.Black)
        Button3.BackColor = Color.White;
    else
        Button3.BackColor = Color.Black;
}
protected void Button4_Click(object sender, EventArgs e)
{
    if (Button4.BackColor == Color.Black)
        Button4.BackColor = Color.White;
    else
        Button4.BackColor = Color.Black;
}
protected void Button5_Click(object sender, EventArgs e)
{
    if (Button5.BackColor == Color.Black)
        Button5.BackColor = Color.White;
    else
        Button5.BackColor = Color.Black;
}
protected void Button6_Click(object sender, EventArgs e)
{
    if (Button6.BackColor == Color.Black)
        Button6.BackColor = Color.White;
    else
        Button6.BackColor = Color.Black;
}
protected void Button7_Click(object sender, EventArgs e)
{
    if (Button7.BackColor == Color.Black)
        Button7.BackColor = Color.White;
    else
        Button7.BackColor = Color.Black;
}
protected void Button8_Click(object sender, EventArgs e)
{
    if (Button8.BackColor == Color.Black)
        Button8.BackColor = Color.White;
    else

```

```

        Button8.BackColor = Color.Black;
    }
    protected void Button16_Click(object sender, EventArgs e)
    {
        if (Button16.BackColor == Color.Black)
            Button16.BackColor = Color.White;
        else
            Button16.BackColor = Color.Black;
    }
    protected void Button17_Click(object sender, EventArgs e)
    {
        if (Button17.BackColor == Color.Black)
            Button17.BackColor = Color.White;
        else
            Button17.BackColor = Color.Black;
    }
    protected void Button18_Click(object sender, EventArgs e)
    {
        if (Button18.BackColor == Color.Black)
            Button18.BackColor = Color.White;
        else
            Button18.BackColor = Color.Black;
    }
    protected void Button19_Click(object sender, EventArgs e)
    {
        if (Button19.BackColor == Color.Black)
            Button19.BackColor = Color.White;
        else
            Button19.BackColor = Color.Black;
    }
    protected void Button13_Click(object sender, EventArgs e)
    {
        if (Button13.BackColor == Color.Black)
            Button13.BackColor = Color.White;
        else
            Button13.BackColor = Color.Black;
    }
    protected void Button14_Click(object sender, EventArgs e)
    {
        if (Button14.BackColor == Color.Black)
            Button14.BackColor = Color.White;
        else
            Button14.BackColor = Color.Black;
    }
    protected void Button15_Click(object sender, EventArgs e)
    {
        if (Button15.BackColor == Color.Black)
            Button15.BackColor = Color.White;
    }

```



```

        else
            Button15.BackColor = Color.Black;
    }
    #endregion

    protected void Button11_Click(object sender, EventArgs e)
    {
        if (Button19.BackColor == Color.Black && Button18.BackColor ==
            Color.Black && Button17.BackColor == Color.Black && Button16.BackColor ==
            Color.Black && Button15.BackColor == Color.Black && Button14.BackColor ==
            Color.Black && Button13.BackColor == Color.Black && Button1.BackColor ==
            Color.Black && Button2.BackColor == Color.Black && Button3.BackColor ==
            Color.Black && Button5.BackColor == Color.Black && Button6.BackColor ==
            Color.Black && Button7.BackColor == Color.Black && Button8.BackColor ==
            Color.Black && Button4.BackColor == Color.Black)
            Panel1.Visible = true;
        else
        {
            Panel2.Visible = true;
            if (DropDownList1.SelectedIndex == 0)
                Panel5.Visible = false;
            else
            {
                extra_calculate();
                listfill(extra, ListBox2);
                Panel5.Visible = true;
            }
            Button1.Enabled = false;
            Button2.Enabled = false;
            Button3.Enabled = false;
            Button4.Enabled = false;
            Button5.Enabled = false;
            Button6.Enabled = false;
            Button7.Enabled = false;
            Button8.Enabled = false;
            Button9.Enabled = false;
            Button13.Enabled = false;
            Button14.Enabled = false;
            Button15.Enabled = false;
            Button16.Enabled = false;
            Button17.Enabled = false;
            Button18.Enabled = false;
            Button19.Enabled = false;
            DropDownList1.Enabled = false;
            Button11.Enabled = false;
        }
    }
}

```

```

protected void Button10_Click(object sender, EventArgs e)
{
    if (FileUpload1.HasFile)
    {
        if (Server.MapPath("~/Data/").Contains(FileUpload1.FileName))
            File.Delete(Server.MapPath("~/Data/" + FileUpload1.FileName));
        FileUpload1.SaveAs(Server.MapPath("~/Data/") + "veri.txt");
        FileStream fs = new FileStream(Server.MapPath("~/Data/") + "veri.txt",
            FileMode.Open, FileAccess.Read);
        StreamReader sr = new StreamReader(fs);

        string yazi;
        string terx;
        double sayi = 0;
        ListBox1.Items.Clear();

        for (int i = 0; i < 24 * 365; i++)
        {
            terx = sr.ReadLine();
            try
            {
                yazi = terx.Replace(".", ",");
                sayi = Convert.ToDouble(yazi);
            }
            catch { }
            isinim[i] = sayi;
        }

        sr.Close();
        fs.Close();

        if (Button10.Text == "Calculate")
        {
            if (DropDownList1.SelectedIndex == 1)
            {
                extra_calculate();
                for (int i = 0; i < 8760; i++)
                {
                    if (extra[i] <= 0)
                        isinim[i] = 0;
                }
            }
            filterin();

            for (int i = 1; i < 8760; i = i + 2)
                Chart2.Series["Series1"].Points.AddXY(i, isinim[i]);
        }
    }
}

```

```

        listfill(isinim, ListBox1);
        Panel3.Visible = true;
        Button10.Text = "Predict";
    }
    else
    {
        if (DropDownList1.SelectedIndex == 1)
        {
            extra_calculate();
            for (int i = 0; i < 8760; i++)
            {
                if (extra[i] <= 0)
                    isinim[i] = 0;
            }
        }
        listfill(isinim, ListBox1);
        predict();
    }
}
else if (Button10.Text == "Predict")
{
    predict();
}
}

void filterin()
{
    filter_selection();
    Filter fir = new Filter(isinim, extra, b, fb);
    fir.filterin();
    double gor;
    for (int i = 1; i <= fir.coef.Length; i++)
    {
        gor = Math.Round(fir.coef[i - 1, 0], 4);
        kas[i].Text = "a" + i + " : " + gor.ToString();
        kas[i].Visible = true;
    }
}

void filter_selection()
{
    if (Button1.BackColor == Color.White)
        b[5] = 1;
    else
        b[5] = 0;
    if (Button2.BackColor == Color.White)
        b[6] = 1;
    else
        b[6] = 0;
}

```

```

if (Button3.BackColor == Color.White)
    b[7] = 1;
else
    b[7] = 0;
if (Button4.BackColor == Color.White)
    b[9] = 1;
else
    b[9] = 0;
if (Button5.BackColor == Color.White)
    b[10] = 1;
else
    b[10] = 0;
if (Button6.BackColor == Color.White)
    b[11] = 1;
else
    b[11] = 0;
if (Button7.BackColor == Color.White)
    b[13] = 1;
else
    b[13] = 0;
if (Button8.BackColor == Color.White)
    b[14] = 1;
else
    b[14] = 0;
if (Button13.BackColor == Color.White)
    b[4] = 1;
else
    b[4] = 0;
if (Button14.BackColor == Color.White)
    b[8] = 1;
else
    b[8] = 0;
if (Button15.BackColor == Color.White)
    b[12] = 1;
else
    b[12] = 0;
if (Button16.BackColor == Color.White)
    b[0] = 1;
else
    b[0] = 0;
if (Button17.BackColor == Color.White)
    b[1] = 1;
else
    b[1] = 0;
if (Button18.BackColor == Color.White)
    b[2] = 1;
else
    b[2] = 0;

```

```

if (Button19.BackColor == Color.White)
    b[3] = 1;
else
    b[3] = 0;

if (DropDownList1.SelectedIndex == 1)
    for (int i = 0; i < 15; i++)
        b[i + 15] = b[i];
else
    for (int i = 0; i < 8; i++)
        b[i + 15] = 0;

bool s1, s2, s3, s4,s5,s6;

s1 = (b[0] == 0 && b[1] == 0 && b[2] == 0 && b[3] == 0);
s2 = (b[4] == 0 && b[5] == 0 && b[6] == 0 && b[7] == 0);
s3 = (b[8] == 0 && b[9] == 0 && b[10] == 0 && b[11] == 0);
s4 = (b[0] == 0 && b[4] == 0 && b[8] == 0 && b[12] == 0);
s5 = (b[1] == 0 && b[5] == 0 && b[9] == 0 && b[13] == 0);
s6 = (b[2] == 0 && b[6] == 0 && b[10] == 0 && b[14] == 0);

if (s1)
    fb = 51;
if (s1&& s2)
    fb = 27;
if (s1&& s2&& s3)
    fb=3;
if (s4)
    fb--;
if (s4&& s5)
    fb--;
if (s4&& s5&& s6)
    fb--;

}

void predict()
{
    filter_selection();
    if(ListBox1.Items.Count!=0)
    {
        string[] parca;
        for (int i = 0; i < ListBox1.Items.Count; i++)
        {
            parca=ListBox1.Items[i].Text.Split('-');
            string sss=null;
            foreach (string k in parca)
                sss = k;
        }
    }
}

```

```

        isinim[i] = Convert.ToDouble(sss);
    }
}
extra_calculate();
for (int i = 1; i < 8760; i = i + 2)
{
    Chart2.Series["Series1"].Points.AddXY(i, isinim[i]);
}

double[] katsayilar = new double[30];
int m = 1;

for (int i = 0; i < 30; i++)
{
    katsayilar[i] = 0;
    if (b[i] != 0)
    {
        string str = kas[m].Text;
        string s = str.Remove(0, 6);
        katsayilar[i] = Convert.ToDouble(s);
        m++;
    }
}

for (int i = fb; i < 8760; i++)
{
    for (int j = 0; j < 15; j++)
    {
        if (i >= aa[j])
            preisinim[i] = preisinim[i] + isinim[i - aa[j]] * katsayilar[j] + extra[i -
aa[j]] * katsayilar[j + 15] ;
    }
    if (preisinim[i] < 0) preisinim[i] = 0;
    if (extra[i] <= 0 && DropDownList1.SelectedIndex==1) preisinim[i] = 0;
    if (extra[i] > 0 && DropDownList1.SelectedIndex == 1 && preisinim[i] == 0)
preisinim[i] = isinim[i];

}

listfill(preisinim, ListBox2);
int month = 1, day = 1, hour;
for (int i = 0; i < fb; i++)
{
    hour = i;
    if (i > 23)
    { hour = i - 24;
    day = 2;
    }
}

```

```

        if (i > 47) {
            hour = i - 48;
            day = 3;
        }
        ListBox2.Items[i].Text = month.ToString() + "." + day.ToString() + "." +
hour.ToString() + " - " + "NaN
    }

    Label1.Text = "Predicted Data";

    FileStream file = new FileStream(Server.MapPath("~/Data/" + "save.txt"),
    FileMode.Truncate);
    StreamWriter stream = new StreamWriter(file);

    string[] parca2;
    if (ListBox2.Items.Count > 0)
    {
        for (int i = 0; i < ListBox2.Items.Count; i++)
        {
            parca2 = ListBox2.Items[i].Text.Split('-');
            string sss = null;
            foreach (string k in parca2)
                sss = k;
            stream.WriteLine(sss);
        }
    }
    stream.Flush();
    stream.Close();
    Panel6.Visible = true;
    Panel3.Visible = true;

    for (int i = 1; i < m; i++)
    {
        kas[i].Visible = true;
    }

    for (int i = 1; i < 8760; i = i + 2)
    {
        if (preisinim[i] > 70)
            Chart1.Series["Series1"].Points.AddXY(isinim[i], preisinim[i]);
    }
    double error = 0;
    for (int i = fb; i < 8760; i++)
        error += (isinim[i] - preisinim[i]) * (isinim[i] - preisinim[i]);

    Label26.Text = "MSD  :" + (error / (8760-fb)).ToString();
    error = Math.Sqrt(error / (8760 - fb));
    Label27.Text = "RMSE  :" + error.ToString();

```

```

error = 0;
for (int i = fb; i < 8760; i++)
    error += isinim[i] - preisinim[i];

error = error / (8760 - fb);
Label28.Text = "MBE  :" + error.ToString();
error = 0;
for (int i = fb; i < 8760; i++)
    if (isinim[i] != 0)
        error += (Math.Abs(isinim[i] - preisinim[i]) / isinim[i]);

error = (error / (8760 - fb)) * 100;
Label29.Text = "MAPE  :" + error.ToString();
Label26.Visible = true;
Label27.Visible = true;
Label28.Visible = true;
Label29.Visible = true;
Panel4.Visible = true;
Panel5.Visible = true;
for (int i = 1; i < 8760; i = i + 2)
{
    Chart2.Series["Series2"].Points.AddXY(i, preisinim[i]);
}
}

void listfill(double[] dat, ListBox liste)
{
    liste.Items.Clear();
    int month = 1, day = 1, hour, k = 0;
    for (int i = 0; i < 24 * 365; i++)
    {
        if ((i / 24) == 31)
        {
            k = 31;
            month = 2;
        }
        if ((i / 24) == 59)
        {
            k = 59;
            month = 3;
        }
        if ((i / 24) == 90)
        {
            k = 90;
            month = 4;
        }
        if ((i / 24) == 120)
        {

```



```

        k = 120;
        month = 5;
    }
    if ((i / 24) == 151)
    {
        k = 151;
        month = 6;
    }
    if ((i / 24) == 181)
    {
        k = 181;
        month = 7;
    }
    if ((i / 24) == 212)
    {
        k = 212;
        month = 8;
    }
    if ((i / 24) == 243)
    {
        k = 243;
        month = 9;
    }
    if ((i / 24) == 273)
    {
        k = 273;
        month = 10;
    }
    if ((i / 24) == 304)
    {
        k = 304;
        month = 11;
    }
    if ((i / 24) == 334)
    {
        k = 334;
        month = 12;
    }

    day = i / 24 + 1 - k;
    hour = i - (day + k - 1) * 24
    liste.Items.Add(ay + "." + day + "." + hour + " - " + dat[i]);
}
}

void extra_calculate()
{
    if (TextBox1.Text == "")

```

```

        fider = 38.45;
    else
    {
        string str;
        if (TextBox1.Text.Contains("."))
            str = TextBox1.Text.Replace(".", ",");
        else
            str = TextBox1.Text;
        fider = Convert.ToDouble(str);
    }

    int e = 0;
    for (int i = 1; i <= 365; i++)
        for (int j = 1; j <= 24; j++)
        {
            double delta = (23.45 * Math.Sin(2 * Math.PI * (284 + i) / 365)) * Math.PI
/ 180;
            double W = (2 * Math.PI * (j - 12)) / 24;
            double fi = fider * Math.PI / 180;
            extra[e] = 1367.7 * (1 + 0.033 * Math.Cos((2 * Math.PI * i / 365))) *
(Math.Cos(fi) * Math.Cos(delta) * Math.Cos(W) + Math.Sin(fi) * Math.Sin(delta));
            if (extra[e] < 0)
                extra[e] = 0;
            e++;
        }
    }

    protected void Button12_Click(object sender, EventArgs e)
    {
        Response.Redirect("Default.aspx");
    }
}
}

```

HTML KODLARI

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits=" LINEAR_PREDICTION_FILTER.Default" %>
```

```
<%@ Register Assembly="System.Web.DataVisualization, Version=4.0.0.0,
Culture=neutral, PublicKeyToken=31bf3856ad364e35"
Namespace="System.Web.UI.DataVisualization.Charting" TagPrefix="asp" %>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="bootstrap/css/bootstrap.min.css" rel="stylesheet" type="text/css" />
    <link href="Styles/Site.css" rel="stylesheet" type="text/css" />
    <script src="script.js" type="text/javascript"></script>
    <script type="text/javascript" src="bootstrap/js/bootstrap.min.js"></script>
```

```
<script type="text/javascript">
    $(document).ready(function () {
        $('#Button10').hide();
        $('#TextBox1').hide();
        $('#TextBox1').attr('placeholder', 'Latitude = ?');

        if ($('#DropDownList1').val() == 'MOD-2') {
            $('#TextBox1').show();
        } else {
            $('#TextBox1').hide();
        }
        $('#DropDownList1').change(function () {
            if ($(this).val() == 'MOD-2') {
                $('#TextBox1').show();
            } else {
                $('#TextBox1').hide();
            }
            $('#.alert').hide();
        })
        var listBox1 = $('#ListBox1 option').length;
        if (listBox1 > 0) {
            $('#Button10').show();
        }
        $('#FileUpload1').change(function () {
            var filepath = this.value;
            var m = filepath.match(/([^\v\\]+)$/);
```

```

        var filename = m[1];
        $('#filename1').html(filename);
        $('#Button10').show(500);

    });
})
</script>
</head>
<body>
    <form id="form1" runat="server">

        <div class="container">
            <h1 class="text-uppercase text-center prediction-title"> LINEAR PREDICTION
FILTER
        </h1>
        <div class="row">
            <div class="pull-right restart-button">
                <i class="glyphicon glyphicon-repeat"></i> <asp:Button ID="Button12"
                    runat="server" class="btn btn-primary" Text=" Restart"
                    onclick="Button12_Click" />
                <button type="button" class="btn btn-info" data-toggle="modal" data-
target="#myModal">
                    User Manuel
                </button>
                <asp:Panel ID="Panel6" runat="server">
                    <a href="data/save.txt" class="btn btn-success" download>Download
Data</a>
                </asp:Panel>
                <div class="modal fade" id="myModal" tabindex="-1" role="dialog"
aria-labelledby="myModalLabel">
                    <div class="modal-dialog" role="document">
                        <div class="modal-content">
                            <div class="modal-header">
                                <button type="button" class="close" data-dismiss="modal" aria-
label="Close"><span aria-hidden="true">&times;</span></button>
                                <h4 class="modal-title text-center" id="myModalLabel">User
Manuel</h4>
                            </div>
                            <div class="modal-body">
                                <p>The application is an interface that predicts with the linear
filter method. Any kind of annual hourly data can be predicted by image processing
linear filter method with desired filter templates. You can analyze the prediction with
using error methods and you can compare the success of the method with predictions
made with other methods.</p>
                                <p class="text-center"></p>
                                <p>Firstly, the data is converted to a two-dimensional image format. The selected
template is filtered on the image. For example a filter template is shown below,</p>

```

<p class="text-center"></p>

<p>

To calculate filter coefficients(a_1, a_2, a_3) with formula;</p>

<p class="text-center">

</p>

<p>

Here i day, j time and $Z_{i,j}$ expresses the value of solar radiation at the i . day and j . hour time.

In the linear forecasting, each past sample in the shifted template is sized with the appropriate coefficients determined to minimize the quadratic prediction error.

</p>

<p class="text-center">

</p>

<p class="text-center"></p>

<p>Where r_{ij} represents the correlation between the z_i and z_j pixels, and r_k represents the correlation between the predicted pixel and each pixel in the filter template. If another data is used for prediction,</p>

<p class="text-center"></p>

<p class="text-center">

</p>

<h4>How to Use?</h4>

<p>Firstly, the filter template on the opening screen is selected by clicking on the black boxes. If you click on any black box what is past value of present value that will be predicted, black box turn into white. For example, when you select the box Z (t-25), the forecasting model will be developed by using the data of the one day and one hour before the current time.

</p>

<p>

If the template is selected, the file upload tool and data list will be available. The file you need to load must be .txt extension and it must contain 8760 (365x24)piece of data as shown below.</p>

<p class="text-center"></p>

<p>You can see the data that you have imported on the data list. To do this, click the "calculate" button. When you click the button, coefficients will be listed under the filter template. How many coefficients are selected from the template are sorted like $a_1, a_2 \dots$ left to right then top to bottom.</p>

<p>Once the optimal coefficients have been calculated, the "predict" button will appear next to the file uploader. If you click directly on the button, it will predict the data that

you used in the previous model. You can also predict with the same model by loading a different dataset. When the "predict" button is clicked, the graphs and error calculation results will be displayed below. The data value shown as "NaN" in the predicted data list indicates that it is not predicted because there is no history in the template. In addition, the "save" button will appear on the upper right. You can download the predicted data with this button.

The forecast can be monitored and analyzed numerically and visually. Two different MODs have been developed. Any data can be filtered in the first MOD. In the second MOD, only solar radiation can be predicted. Here, extraterrestrial radiation data is calculated by entering the latitude information. In solar radiation forecasting two data are superimposed on each other to obtain better results in the filtering process. When latitude information is not entered, the program defaults to 38.45 for

Afyonkarahisar/Turkey.

In solar radiation forecasting with image processing linear filter method, when there is no sun, method gives a prediction with coefficients and past values. This logic error is solved in the second MOD with the established algorithm. In this case, the total error is smaller.

You can send your offers and comments to ustundag_emrah@hotmail.com, fohocaoglu@gmail.com, eakarslan@aku.edu.tr

```

</p>
</div>
<div class="modal-footer">
  <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
  <!--<button type="button" class="btn btn-primary">Save
changes</button>-->
</div>
</div>
</div>
</div>
</div>
<div class="col-md-3">
  <div class="panel panel-primary panel-setting">
    <div class="panel-heading">
      <h3 class="panel-title">Filter Template</h3>

    </div>
    <div class="panel-body">
      <div class="row filter-selection text-center">
        <asp:DropDownList ID="DropDownList1" runat="server">
</asp:DropDownList>
        <asp:Button ID="Button11" runat="server" class="sec"
OnClick="Button11_Click" Text="Select" />
      <div class="clearfix"></div>
      <br />

```

```

        <asp:TextBox ID="TextBox1" class="form-control"
runat="server"></asp:TextBox>
        <div class="clearfix"></div>
        <br />
        <asp:ScriptManager ID="ScriptManager1" runat="server">
</asp:ScriptManager>
        <asp:UpdatePanel ID="UpdatePanel1" runat="server">

        <ContentTemplate>
        <asp:Button ID="Button16" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button16_Click" Text="Z(t-75)" />
        <asp:Button ID="Button17" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button17_Click" Text="Z(t-74)" />
        <asp:Button ID="Button18" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button18_Click" Text="Z(t-73)" />
        <asp:Button ID="Button19" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button19_Click" Text="Z(t-72)" /><br />
        <asp:Button ID="Button13" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button13_Click" Text="Z(t-51)" />
        <asp:Button ID="Button1" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button1_Click" Text="Z(t-50)" />
        <asp:Button ID="Button2" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button2_Click" Text="Z(t-49)" />
        <asp:Button ID="Button3" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button3_Click" Text="Z(t-48)" /><br />
        <asp:Button ID="Button14" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button14_Click" Text="Z(t-27)" />
        <asp:Button ID="Button4" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button4_Click" Text="Z(t-26)" />
        <asp:Button ID="Button5" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button5_Click" Text="Z(t-25)" />
        <asp:Button ID="Button6" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button6_Click" Text="Z(t-24)" /><br />
        <asp:Button ID="Button15" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button15_Click" Text="Z(t-3)" />

```

```

        <asp:Button ID="Button7" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button7_Click" Text="Z(t-2)" />
        <asp:Button ID="Button8" runat="server" class="image-submit"
BackColor="Black"
        onclick="Button8_Click" Font-Italic="False" Font-Size="Smaller"
        Text="Z(t-1)" />
        <asp:Button ID="Button9" runat="server" class="image-submit" Font-
Italic="True"
        Font-Size="Smaller" Text="Z(t)=?" /><br /><br />

```

```

    </ContentTemplate>
</asp:UpdatePanel>
    <asp:Panel ID="Panel1" runat="server">
        <div class="alert alert-danger" role="alert">
            <span class="glyphicon glyphicon-exclamation-sign" aria-
hidden="true"></span>
            <span class="sr-only">Error:</span>
            Select Filter Template!
        </div>
    </asp:Panel>
    <asp:Panel ID="Panel3" runat="server">
        <h1 class="baslik text-center">Coefficients
        </h1>
        <asp:Label ID="Label2" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label3" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label4" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label5" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label6" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label7" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label8" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label9" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label10" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label11" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label12" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label13" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label14" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label15" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label16" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label17" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label18" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label19" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label20" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label21" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label22" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label23" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label24" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label25" runat="server" Text="Label"></asp:Label>
    </asp:Panel>

```



```

        <asp:Label ID="Label31" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label32" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label33" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label34" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label35" runat="server" Text="Label"></asp:Label>
        <asp:Label ID="Label36" runat="server" Text="Label"></asp:Label>
    </asp:Panel>
</div>
</div>
</div>
</div>
<div class="col-md-9">
    <asp:Panel ID="Panel2" runat="server">
        <div class="row">
            <div class="col-md-6">
                <div class="file-upload row">
                    <span class="filename" id="filename1">Load Filter File</span>
                    <label for="file-upload"><i class="glyphicon glyphicon-folder-
open"></i>Browse<asp:FileUpload ID="FileUpload1" class="file-upload1"
runat="server" Height="25" />
                    </label>
                    <asp:Button ID="Button10" runat="server" class="btn"
onclick="Button10_Click"
                        Text="Calculate" />
                </div>
                <div class="col-md-12 text-center">
                    <asp:Label ID="Label30" class="text-center baslik" runat="server"
Text="Measured Data"></asp:Label>
                </div>
                <div class="clearfix text-center">
                    <asp:ListBox ID="ListBox1" class="list-style"
runat="server"></asp:ListBox>
                </div>
            </div>
            <asp:Panel ID="Panel5" runat="server">
                <div class="col-md-6">
                    <div class="col-md-12 text-center ">
                        <asp:Label ID="Label1" class="text-center title2 baslik"
runat="server" Text="Extraterrestrial"></asp:Label>
                    </div>
                    <div class="clearfix text-center">
                        <asp:ListBox ID="ListBox2" runat="server"></asp:ListBox>
                    </div>
                </div>
            </asp:Panel>
        </div>
    </asp:Panel>
</div>

```

```

<asp:Panel ID="Panel4" runat="server">
  <div class="row">

    <div class="col-md-6">

      <div class="efect">
        <div class="panel panel-primary">
          <div class="panel-heading"> <h3 class="panel-title">Data
Graph</h3> </div>
          <div class="panel-body">
            <asp:Chart ID="Chart2" runat="server">
              <Series>
                <asp:Series Name="Series1" ChartType="Point">
                  </asp:Series>
                <asp:Series ChartArea="ChartArea2" ChartType="Point"
Name="Series2">
                  </asp:Series>
              </Series>
              <ChartAreas>
                <asp:ChartArea Name="ChartArea2">
                  </asp:ChartArea>
              </ChartAreas>
            </asp:Chart>
          </div>
        </div>
      </div>
    </div>
    <div class="col-md-6">

      <div class="efect">
        <div class="panel panel-primary">
          <div class="panel-heading"> <h3 class="panel-title">Correlation
Graph</h3> </div>
          <div class="panel-body">
            <asp:Chart ID="Chart1" style="height:350px"
runat="server"><Series>
              <asp:Series ChartType="Point"
Name="Series1"></asp:Series></Series><ChartAreas><asp:ChartArea
Name="ChartArea1"><AxisY><ScaleBreakStyle StartFromZero="Yes"
/></AxisY><AxisX InterlacedColor="Transparent"><ScaleBreakStyle
StartFromZero="Yes" /></AxisX></asp:ChartArea></ChartAreas></asp:Chart>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```


CSS KODLARI

```
.container
{
    border: 1px solid #a7a7a7;
    padding: 20px 0;
    box-shadow: 1px 0px 19px 1px #847d7d;
    overflow: hidden;
}
.image-submit
{
    border: 2px solid #a7a7a7;
    margin-right: -1px;
    width: 35px;
    height: 35px;
}
#form1
{
    padding: 10px 0;
}
.file-upload
{
    margin-bottom: 10px;
    text-align: center;
}
@import url("https://fonts.googleapis.com/css?family=Spinnaker");
.file-upload1 {
    position: absolute;
    opacity: 0;
    margin-left: -6px;
    height: 34px;
    margin-top: -26px;
    width: 87px;
}

label[for="file-upload"] {
    padding: 0.5em;
    display: inline-block;
    color: #fff;
    background-color: #007bff;
    border-color: #007bff;
    cursor: pointer;
    transition: all 0.5s ease;
}
label[for="file-upload"] i
{
    margin-right: 7px;
}
```

```

label[for="file-upload"]:hover {
    background: #6b3880;
}

.filename {
    padding: 0.7em;
    width: 150px;
    white-space: nowrap;
    overflow: hidden;
    margin-top : 1px;
    background : #dad7da;
}
#ListBox1
{
    width: 70%;
    height: 250px;
    margin-top: 10px;
}
#ListBox2
{
    width: 70%;
    height: 250px;
}
.title2
{
    padding: 47px 0 12px 0;
    display: inline-block;
}
#Panel4
{
    margin-top: 20px;
}

#Panel2 select
{
    border-bottom: 1px solid #f3eeee;
    padding: 4px;
    margin-bottom: 7px;
    display: inline-block;
    width: 53%;
    color: #3b6153;
    font-weight: bold;
}
#Panel2 select option
{
    border-bottom: 1px solid #c3c0c0;
    padding: 5px 0;
}

```

```

#Chart1
{
    width: 100% !important;
}
#Panel3
{
    text-align: left;
}
#Panel3 span
{
    width: 100%;
    display: inline-block;
    border-bottom: 1px solid #77d4b1;
    padding: 5px 0;
    padding-left: 73px;
    font-style: italic;
}
#Div1 span
{
    width: 100%;
    display: inline-block;
    border-bottom: 1px solid #77d4b1;
    padding: 5px 0;
    padding-left: 50px;
    font-style: italic;
}
.baslik
{
    font-size: 25px;
    text-shadow: 2px 4px #e6dddd;
    color: #007bff;
}
.sec
{
    width: 70px !important;
    height: 25px !important;
    border: none !important;
    color: #313030;
    font-weight: bold;
    border-radius: 7px;
    -webkit-border-radius: 7px;
    -moz-border-radius: 7px;
    -o-border-radius: 7px;
    background: #dddddd;
}
.prediction-title
{
    text-shadow: 2px 4px #e6dddd;
}

```

```

        color: #007bff;
    }
    .filter-selection .form-control
    {
        width: 30%;
        margin: 0 0 0 57px;
        color: #a4a4a4;
    }
    div#UpdatePanel1 input
    {
        width: 50px;
        height: 50px;
        font-size: 13px;
    }
    #Chart2
    {
        height: 349px !important;
        width: 100% !important;
    }
    .glyphicon-repeat
    {
        right: -25px;
        color: #fff;
        position: relative;
        top: 3px;
        font-size: 13px;
        text-shadow: none;
        -o-text-shadow: none;
        -moz-text-shadow: none;
        -webkit-text-shadow: none;
    }
    #Button12
    {
        padding-left: 23px;
    }
    .restart-button
    {
        margin-right: 20px;
    }
    #Panel6
    {
        float: right;
        margin-left: 5px;
    }
    .pre-article p
    {
        width: 100%;
        display: inline-block;
    }

```

```
    color: #9e9797;
    font-style: italic;
    font-size: 11px;
}
.panel-default .panel-title
{
    font-size: 13px;
}
.modal-body
{
    line-height: 24px;
    padding: 0 50px;
}
.articles
{
    width: 100%;
    margin: 0 auto;
}
@media(max-width: 480px)
{
    .pull-right
    {
        margin: 10px 25px;
    }
}
```

