

Time Series Forecasting

Ömer Faruk Ercan ¹

1Elektronik ve Haberleşme Mühendisliği, Kocaeli Üniversitesi, Kocaeli, 41001, Türkiye, ORCID: XXXX-XXXX-XXXX-XXXX

2Electronics and Communication Engineering, Kocaeli University, Kocaeli, 41001, Turkey, ORCID: XXXX-XXXX-XXXX-XXXX

Article Info	Özet
Research paper Review paper	Bu projede, zaman serisi tahmini için Prophet, ARIMA, Holt-Winters, ETS ve XGBoost modelleri kullanılarak farklı veri setleri üzerinde analizler gerçekleştirilmiştir. Bitcoin fiyatları, avokado fiyatları ve AEP şirketinin elektrik tüketim verileri olmak üzere üç farklı veri seti kullanılmıştır. Her bir modelin tahmin performansı karşılaştırılarak, veri türüne ve özelliklerine göre hangi modelin daha etkili olduğu değerlendirilmiştir. Prophet ve ARIMA modelleri trend ve mevsimsellik bileşenlerini yakalama konusunda güçlü performans sergilerken, XGBoost, kompleks ve doğrusal olmayan ilişkileri yakalamada öne çıkmıştır. Holt-Winters ve ETS modelleri özellikle mevsimsel veriler için etkili olmuştur. Sonuçlar, model seçiminin verinin yapısına bağlı olarak değiştiğini göstermiştir.
Anahtar Kelimeler	
<i>Zaman serisi kestrimi, Ets, Prophet, Arima, Holt, XGBoost</i>	
Article Info	Abstract
Research paper Review paper	In this project, time series forecasting was performed using Prophet, ARIMA, Holt-Winters, ETS, and XGBoost models on different datasets. The datasets included Bitcoin prices, avocado prices, and electricity consumption data from the AEP company. The forecasting performance of each model was compared to evaluate which model performed better based on the type and characteristics of the data. Prophet and ARIMA models demonstrated strong performance in capturing trends and seasonality, while XGBoost excelled at identifying complex and non-linear relationships. Holt-Winters and ETS models were particularly effective for seasonal data. The results highlighted that model selection depends on the structure of the dataset.
Keywords	
<i>Time series forecasting, Ets, Prophet, Arima, Holt, XgBoost</i>	

- 1 **1. Giriş**
- 2
- 3 Zaman serisi analizi, geçmiş verilere dayanarak
- 4 gelecekteki değerleri tahmin etmek için güçlü araçlar
- 5 sunar. Bu bağlamda, farklı tahmin modellerinin
- 6 karşılaştırılması, doğru ve güvenilir öngörülerde
- 7 bulunmak adına büyük önem taşır. Bu raporda, zaman
- 8 serisi tahminleri için Prophet, Holt, Ets ve Arima gibi
- 9 istatistiksel modeller seçildiği gibi makine öğrenmesi
- 10 modeli olan XGBoost modeli de kullanılmıştır. Her bir
- 11 modelin teorik temelleri ve kullanım alanları açıklanacak,
- 12 ardından Python ile bu modellerin nasıl uygulanacağına
- 13 dair örnek kodlar sunulacaktır. Böylece, zaman serisi
- 14 verilerinin analizi ve tahmini için hangi modelin hangi
- 15 koşullarda daha etkili olabileceğine dair bir bakış açısı
- 16 sağlanması hedeflenmektedir
- 17
- 18 **2. Malzeme ve Yöntem**
- 19
- 20 Bu projede zaman serisi verilerinin analizi ve tahmin
- 21 edilmesi amacıyla Python programlama dili kullanılmıştır.
- 22 Kodlama ortamı olarak PyCharm IDE tercih edilmiştir.

1 Projede veri manipülasyonu, modelleme ve görselleştirme
2 işlemleri için çeşitli Python kütüphaneleri kullanılmıştır.
3 Verilerin işlenmesi için Pandas kütüphanesi tercih edilmiştir.
4 Pandas, zaman serisi verilerinin düzenlenmesi, eksik
5 değerlerin doldurulması ve analiz için uygun hale
6 getirilmesi işlemlerinde kullanılmıştır. Tahmin modelleme
7 aşamasında, Prophet kütüphanesi, özellikle sezonluk ve
8 trend bileşenlerini modellemek için uygulanmıştır. Ayrıca,
9 zaman serisi analizinde klasik yöntemlerin kullanılması
10 amacıyla ARIMA ve Holt-Winters Exponential Smoothing
11 modelleri Statsmodels kütüphanesi aracılığıyla
12 gerçekleştirilmiştir. XGBoost, daha ileri düzey bir makine
13 öğrenmesi yöntemi olarak, tahminlerin doğruluğunu

14 artırmak için regresyon problemleri çözmekte kullanılmıştır.
15 Modelleme ve tahmin işlemleri tamamlandıktan sonra,
16 sonuçlar Matplotlib kütüphanesi ile görselleştirilmiş ve
17 farklı tahmin modelleri arasındaki performans farkları
18 görsel olarak karşılaştırılmıştır. Son olarak, proje
19 kapsamında kullanıcıların verileri ve tahmin sonuçlarını
20 görsel olarak görüntülemesi için Tkinter kütüphanesiyle
21 basit bir grafiksel kullanıcı arayüzü geliştirilmiştir. Bu
22 arayüz, kullanıcı dostu bir platform sağlayarak, tahmin
23 sonuçlarının etkili bir şekilde sunulmasına olanak tanımıştır.
24 Proje boyunca, zaman serisi verilerinin analizi, tahmini ve
25 sonuçların görselleştirilmesi gibi temel adımlar başarıyla
26 gerçekleştirilmiştir.

27

28 **Tablo 1.** Modeller,

Modeller	Tanım	Hangi Veri Türünde Daha İyi	Avantajları	Dezavantajları
ARIMA	Otoregresif Entegre Hareketli Ortalama Modeli, zaman serisi verilerinin geleceğini tahmin etmek için kullanılır.	Durağan zaman serileri (trend ve mevsimsellik olmayan) için uygundur.	- Matematiksel olarak basit ve anlaşılır. - Yüksek doğrulukla tahmin yapabilir.	- Durağan olmayan serilerde düşük performans. - Mevsimsel etkileri modellemek zor olabilir.
HOLT	Holt'un iki katmanlı eksponensiyel düzey modelidir ve zaman serilerindeki trendi modellemek için kullanılır.	Yalnızca trend gösteren verilerde etkilidir, mevsimsellik olmayan verilerde daha başarılıdır.	- Trendi iyi modelleyebilir. - Basit ve hızlıdır. - Kısa vadeli tahminlerde başarılıdır.	- Mevsimsel bileşenleri modelleyemez. - Uzun vadeli tahminlerde zayıf kalabilir.
ETS	Eksponansiyel Düzeyleme Yöntemi, zaman serilerinde trend ve mevsimsellik gibi bileşenleri dikkate alarak tahmin yapar.	Mevsimsel ve trend içeren zaman serileri için uygundur.	- Trend ve mevsimsel bileşenleri aynı anda modelleyebilir. - Kısa vadeli tahminlerde yüksek doğruluk.	- Karmaşık olmayan, sadece düz ve düz trendli verilerde başarısız olabilir. - Uzun vadeli tahminler için sınırlı başarı.
PROPHET	Facebook tarafından geliştirilen, sezonlukluk, tatiller ve trendleri otomatik olarak tespit eden bir model.	Mevsimsel ve tatil etkileri gösteren zaman serileri için uygundur.	-Kolayca ayarlanabilir. - Mevsimsel ve tatil etkilerini doğru bir şekilde modelleyebilir. - Eksik veri ve aykırı değerlere dayanıklıdır.	- Karmaşık yapılar ve çok fazla veri gerektiren projelerde daha yavaş çalışabilir. - Diğer modeller kadar esnek olmayabilir.
XGBOOST	İleri düzeyde makine öğrenmesi tekniği olan Gradient Boosting algoritmasını kullanır.	Büyük veri setleri ve karmaşık ilişkiler için uygundur.	- Yüksek doğruluk oranı. - Aykırı verilere karşı dayanıklıdır. - Esnek ve güçlüdür.	-Parametre ayarlaması zor olabilir. - Yüksek hesaplama gücü gerektirir.

29

3. Modeller

3.1 Arima

Geliştirici: George Box ve Gwilym Jenkins (1970'ler)

Kullanım Alanı: Trend ve mevsimsellik içermeyen zaman serilerini tahmin etmek için kullanılır.

ARIMA modeli üç bileşenden oluşur:

- **AR (Auto-Regressive):** Geçmiş değerler kullanılır.
- **AR(p):** $Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \epsilon$
- **I (Integrated):** Veriyi durağan hale getirmek için fark alma işlemi uygulanır.
- **Yt'=Yt-Yt-1** (Birinci fark)
- **MA (Moving Average):** Geçmiş hata terimleri kullanılır.
- **MA(q):** $Y_t = \epsilon_t + \theta_1 \epsilon_{t-1} + \dots + \theta_q \epsilon_{t-q}$

ARIMA(p,d,q), d sayısı kadar fark alarak seriyi durağanlaştırır, ardından p gecikmeli AR ve q gecikmeli MA terimleri eklenir.

Python kodu :

```
# ARIMA Modeli
arima_model = ARIMA(train_data["y"],
order=(1, 1, 1))
arima_fit = arima_model.fit()
arima_forecast =
arima_fit.forecast(steps=forecast_steps
).values
```

3.2 ETS (Error, Trend, Seasonality)

Geliştirici: Hyndman ve diğerleri (2000'ler)

Kullanım Alanı: Trend ve mevsimsellik içeren zaman serilerinde kullanılır.

- **E (Error):** Hata bileşeni (additif veya çarpımsal)
- **T (Trend):** Lineer, exponansiyel vb. olabilir.
- **S (Seasonality):** Mevsimsellik etkisi.

$$Y_t = l_t - 1 + b_t - 1 + s_t - m + \epsilon_t$$

- l_t : Seviye (Level)
- b_t : Trend (Slope)
- s_t : Mevsimsellik (Seasonality)
- ϵ_t : Hata terimi.

Python kodu :

```
# ETS Modeli
ets_model =
ExponentialSmoothing(train_data["y"],
seasonal=None, trend="add",
damped_trend=True)
ets_fit = ets_model.fit()
ets_forecast =
ets_fit.forecast(steps=forecast_steps).
values
```

3.3 Holt (Holt-Winters)

Geliştirici: Charles Holt ve Peter Winters (1950'ler)

Kullanım Alanı: Trend ve mevsimselliği olan seriler için uygundur.

- **Seviye (Level):** $l_t = \alpha \frac{Y_t}{S_{t-m}} + (1 - \alpha)(l_{t-1} + b_{t-1})$
- **Trend:** $b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1}$
- **Mevsimsellik:** $S_t = \gamma \frac{Y_t}{l_{t-1}} + (1 - \gamma)S_{t-m}$
- **Tahmin:** $Y_{t+h} = (l_t + hb_t)S_{t+h-m}$

Python kodu:

```
# Holt Modeli
holt_model = Holt(train_data["y"],
damped_trend=True)
holt_fit = holt_model.fit()
holt_forecast =
holt_fit.forecast(steps=forecast_steps)
.values
```

3.4 Prophet

Geliştirici: Facebook (2017)

Kullanım Alanı: Günlük/haftalık veriler gibi düzensiz gözlemler, eksik veriler veya büyük mevsimsel etkiler içeren zaman serilerinde kullanılır.

Prophet modeli üç bileşene dayanır:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

- **g(t):** Trend fonksiyonu
- **s(t):** Mevsimsel bileşen
- **h(t):** Tatil etkisi (holiday)
- **εt:** Hata terimi

1 Python kodu:

```

2# Prophet Modeli
3prophet = Prophet()
4prophet.fit(train_data)
5future =
6prophet.make_future_dataframe( periods=f
7forecast_steps)
8prophet_forecast =
9prophet.predict(future)[-
10forecast_steps:] ["yhat"].values

```

12 3.5 XGBoost

14 Geliştirici : The XGBoost Contributors

16 **Kullanım alanı :** XGBoost, zaman serisi tahmininde
17 oldukça etkili bir algoritmadır ve finansal piyasalar, enerji
18 tüketimi, talep tahmini, sağlık, iklim ve trafik gibi pek çok
19 farklı alanda kullanılabilir. Özellikle geçmiş verilere dayalı
20 tahminler yapmak için güçlü bir yöntemdir ve geçmiş
21 gözlemlerle gelecekteki değerlerin tahmin edilmesine
22 olanak tanır. Zaman serisi problemlerinde, XGBoost'un
23 doğrusal olmayan ilişkileri modelleme yeteneği büyük
24 avantaj sağlar. Bununla birlikte, doğru sonuçlar almak için
25 özellik mühendisliği önemlidir; örneğin, geçmiş veriler,
26 mevsimsel faktörler, tatil günleri ve trendler gibi özellikler
27 modele dahil edilmelidir. XGBoost, karar ağaçları
28 kullanarak hataları düzelter ve her iterasyonda daha doğru
29 tahminler yapan bir modeldir. Ayrıca, aşırı uyumun
30 önlenmesi için regularizasyon teknikleri ve erken durdurma
31 yöntemleri sunar. Bu özellikler, zaman serisi verilerinde
32 modelin genelleme gücünü artırır. XGBoost'un paralel
33 işleme yeteneği, büyük veri setlerinde hızlı eğitim ve tahmin
34 yapmayı sağlar. Enerji, perakende, sağlık ve trafik gibi
35 alanlarda, geçmiş verilerin doğru şekilde modellenmesiyle
36 doğru tahminler elde edilebilir. Bunun yanı sıra, makine
37 arızası tespiti ve lojistik gibi endüstriyel uygulamalarda da
38 etkili sonuçlar alınabilir. Zaman serisi analizi için
39 XGBoost'un esnekliği, karmaşık ilişkilerin öğrenilmesine
40 yardımcı olur. Ancak, doğru özellik mühendisliği ve uygun
41 hiperparametre ayarlamaları ile en iyi sonuçlar elde
42 edilebilir. Sonuç olarak, XGBoost, zaman serisi tahmininde
43 güçlü ve esnek bir araçtır.

45 Modelin Genel Formülü :

$$\hat{y}_t = \sum_{k=1}^K f_k(x_t)$$

- 49 • \hat{y}_t = Tahmin edilen değer.
- 50 • K = Ağaç sayısı (yani XGBoost'un öğrenmeye karar
51 verdiği sayıda zayıf öğrenici).
- 52 • $f_k(x_t)$ = k-inci karar ağacının tahmini.

54 Her bir karar ağacı şu şekilde çalışır :

55

$$f_k(x) = leaf(x) = \theta_k$$

57

- 58 • θ_k = k-inci ağacın çıkış değerini ifade eder.

59

60 XGBoost, her bir yeni ağacı eklerken, önceki ağaçların

61 tahmin hatalarını minimize etmeye çalışır. Öğrenme süreci

62 şu adımlarla yapılır:

- 63 • Her bir ağaç, önceki tahminlerin hatalarını
64 minimize etmeye yönelik bir düzeltme yapar.
- 65 • Ağaçları toplarken, öğrenme oranı (τ) kullanılarak
66 ağırlıklandırılır:

$$\hat{y}_t = \hat{y}_t^{(k-1)} + \tau \cdot f_k(x_t)$$

- 68 • τ : Öğrenme oranı (0 ile 1 arasında bir değer).

69 XGBoost'un temel hedefi, kayıp fonksiyonunu minimize

70 etmektir. XGBoost'un kullandığı kayıp fonksiyonu

71 genellikle şu şekilde tanımlanır:

72

$$L(\theta) = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$$

74

- 75 • $L(y_i, \hat{y}_i)$ = Tahmin hatası
- 76 • $\Omega(f_k)$ = Regularizasyon terimi, modelin
77 karmaşıklığını cezalandıran bir terimdir.

78

79 Özetle, XGBoost'un matematiksel modellemesi, zayıf

80 öğreniciler olan karar ağaçlarının birleştirilmesiyle yapılan

81 bir boosting yöntemine dayanır ve hedef, her yeni ağacın

82 önceki tahmin hatalarını düzeltmesidir.

83

84 Python kodu:

```

85 lags = 10
86 xgb_df = pd.DataFrame({f"lag_{i}":
87 train_data["y"].shift(i) for i in
88 range(1, lags + 1)})
89 xgb_df["target"] = train_data["y"]
90 xgb_df.dropna(inplace=True)
91
92 X_train = xgb_df.iloc[:-forecast_steps,
93 :-1]
94 y_train = xgb_df.iloc[:-forecast_steps,
95 :-1]
96 X_test = xgb_df.iloc[-forecast_steps:,
97 :-1]
98
99 xgb_model = XGBRegressor()
100 xgb_model.fit(X_train, y_train)

```

```
1xgb_forecast =
2xgb_model.predict(X_test)
3
```

4. Sonuçlar

Bu çalışmada, zaman serisi tahmin modellerinin performansını değerlendirmek ve gerçek sonuçlarla karşılaştırmak amacıyla verilerin son 10 periyodu modellerin eğitimi dışında bırakılmıştır. Bu yaklaşım, modellerin yalnızca eğitim sırasında gördüğü verilerden etkilenmeyerek, tamamen bağımsız bir veri seti üzerinde öngörü performanslarının test edilmesine olanak tanımıştır. Bu yöntem, tahminlerin doğruluğunu ve güvenilirliğini artırmak açısından kritik bir adım olarak değerlendirilmiştir.

Çalışmada analizler üç farklı veri seti kullanılarak gerçekleştirilmiştir. Bu veri setleri şunlardır:

- Avokado satış verileri:** Amerika'nın farklı bölgelerindeki avokado satışlarına ilişkin tarihsel verileri içermektedir. Veriler, mevsimsel satış dalgalanmalarını, fiyat değişimlerini ve talep trendlerini analiz etmek için uygun bir kaynak sunmuştur. Bu veri setine, Kaggle platformu üzerinden ulaşılmış ve çalışmada kullanılmıştır.
- Enerji tüketim verileri:** American Electric Power (AEP) şirketinin saatlik verileri kullanılmıştır. Enerji tüketimini içeren bu veri seti, enerji sektöründeki talep öngörülerini modellemek için kullanılmıştır. Enerji tüketimindeki mevsimsel ve gün içi değişiklikleri analiz etmek açısından bu veri seti oldukça faydalı olmuştur ve yine Kaggle platformundan temin edilmiştir.
- Apple hisse senedi kapanış fiyat verileri:** Apple Inc.'in hisse senedi fiyatlarının kapanış değerlerini içeren bu veri seti, finansal piyasalardaki zaman serisi analizlerini test etmek için kullanılmıştır. Bu veriler, Python kütüphanesi olan *yfinance* aracılığıyla kolaylıkla elde edilmiştir ve hisse fiyatlarının tarihsel hareketlerini analiz etme imkanı sağlamıştır.

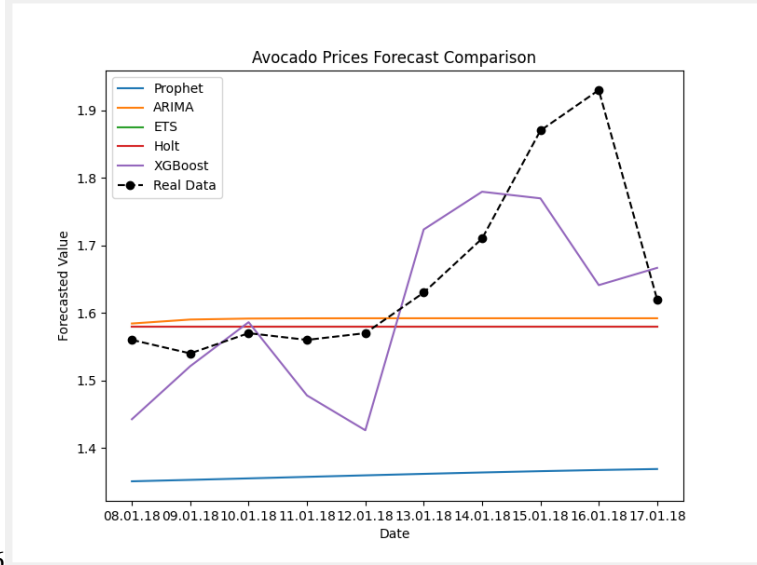
Çalışmada kullanılan tüm veri setleri, farklı alanlardan (gıda, enerji ve finans) seçilerek geniş bir analiz kapsamı oluşturulmuştur. Veriler, farklı kaynaklardan elde edilmiştir ve bu süreçte açık veri kaynaklarının etkin kullanımı vurgulanmıştır. Çalışmanın daha iyi anlaşılabilmesi için kullanılan kodlar ve analiz süreçleri, GitHub üzerinden erişime sunulmuş ve proje bağlantısı raporun ilgili bölümünde paylaşılmıştır.

5. Çıktılar

Resim 5.1 Avokado verilerinin tablosu

Date	Prophet	ARIMA	ETS	Holt	XGBoost	Real Data
08.01.18	1.35057527980269	1.58428754594437	1.57959611311010	1.57959026506112	1.44252538681030	1.56
09.01.18	1.35269312901334	1.59017489513033	1.57959611311010	1.57959026506112	1.52109706401824	1.54
10.01.18	1.35490317323893	1.59160200022019	1.57959611311010	1.57959026506112	1.58624517917633	1.57
11.01.18	1.35715658964298	1.59194793331727	1.57959611311010	1.57959026506112	1.47778034210205	1.56
12.01.18	1.35940265495803	1.59203178818438	1.57959611311010	1.57959026506112	1.42616975307464	1.57
13.01.18	1.36159004183860	1.59205211477006	1.57959611311010	1.57959026506112	1.72352659702301	1.63
14.01.18	1.36366810546167	1.59205704197464	1.57959611311010	1.57959026506112	1.77944958209991	1.71
15.01.18	1.36558812739235	1.59205823633878	1.57959611311010	1.57959026506112	1.76967275142669	1.87
16.01.18	1.36730448571087	1.59205852585501	1.57959611311010	1.57959026506112	1.64111983776092	1.93
17.01.18	1.36877572317111	1.59205859603432	1.57959611311010	1.57959026506112	1.66668486595153	1.62

Resim 5.1 Avokado verilerinin grafiği



Model Performansları:

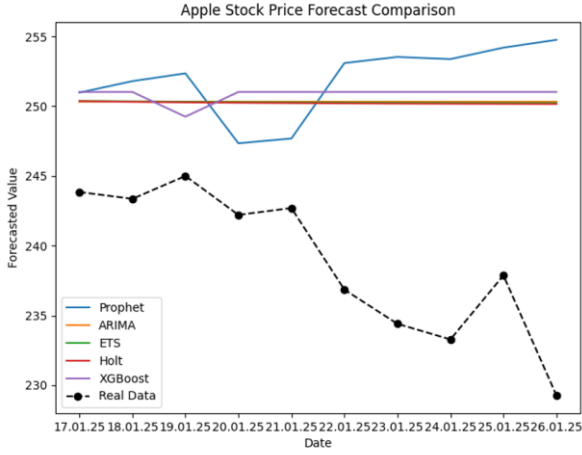
- Prophet:** Grafikteki en düşük ve en düz çizgiye sahip. Gerçek verideki dalgalanmaları yakalayamamış ve genel olarak düşük tahminler üretmiş.
- ARIMA:** Gerçek veriye daha yakın tahminler üretmiş gibi görünse de, özellikle son dönemdeki ani yükselişi yakalayamamış.
- ETS ve Holt:** Bu iki modelin tahminleri neredeyse aynı ve sabit bir çizgi halinde. Gerçek verideki değişimleri hiç yakalayamamışlar.
- XGBoost:** Gerçek veriye en yakın tahminleri üreten model gibi görünüyor. Dalgalanmaları diğer modellere göre daha iyi yakalamış. Özellikle son dönemdeki yükselişi en iyi tahmin eden model.
- En İyi Model:** Grafiklere bakıldığında, XGBoost modelinin avokado fiyatlarını tahmin etmede diğer modellere göre daha başarılı olduğu söylenebilir. Gerçek veriye en yakın tahminleri üretmiş ve dalgalanmaları en iyi şekilde yakalamış.

1Resim5.3 Apple Hisse Kapanış Fiyatı Tablosu

Date	Prophet	ARIMA	ETS	Holt	XGBoost	Real Data
17.01.25	250.976722779758	250.324245964360	250.379445625313	250.362143489792	251.024642944335	243.850006103515
18.01.25	251.799093084937	250.332170080225	250.347003569475	250.315859725603	251.024642944335	243.360000610351
19.01.25	252.357899161325	250.331514308251	250.321049924804	250.278832714196	249.2509765625	245.0
20.01.25	247.344189119927	250.331568577634	250.300287009068	250.249211105026	251.024642944335	242.210006713867
21.01.25	247.687532182132	250.331564086491	250.283676676479	250.225513817654	251.024642944335	242.699996948242
22.01.25	253.095714436402	250.331564458162	250.270388410407	250.20655987729	251.024642944335	236.850006103515
23.01.25	253.534438513777	250.331564427404	250.259757797550	250.191389723766	251.024642944335	234.399993896484
24.01.25	253.380128669711	250.331564429948	250.251253307264	250.179256712577	251.024642944335	233.279998779296
25.01.25	254.202498974884	250.331564429739	250.244449715036	250.169550303611	251.024642944335	237.869995117187
26.01.25	254.761305051281	250.331564429756	250.239006841253	250.161785176427	251.024642944335	229.264999389648

2
3

4Resim5.4 Apple Hisse Kapanış Fiyatı Grafiği



5
6

7Model Performansları:

- **Prophet:** Nispeten düz bir çizgi izlemiş ve genel olarak gerçek değerlerin üzerinde tahminler üretmiş. Düşüş trendini hiç yakalayamamış.
- **ARIMA:** Prophet'e benzer şekilde, nispeten sabit bir çizgi izlemiş ve gerçek verideki düşüşü modelleyememiş. Tahminleri genellikle gerçek değerlerin üzerinde seyretmiş.
- **ETS ve Holt:** Bu iki model neredeyse aynı çizgiyi izlemiş ve tahminleri oldukça sabit kalmış. Gerçek verideki düşüş trendini ve dalgalanmaları hiç yakalayamamışlar. Bu modellerin bu veri seti için uygun olmadığı açıkça görülüyor.
- **XGBoost:** Diğer modellere kıyasla gerçek verilere en yakın tahminleri üretmiş. Düşüş trendini kısmen yakalamış ve dalgalanmalara daha duyarlı bir performans sergilemiş. Ancak, düşüşün keskinliğini tam olarak modelleyememiş ve bazı noktalarda sapmalar göstermiş.

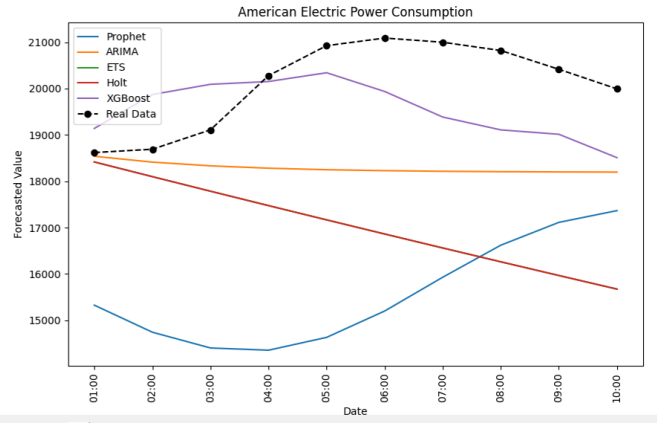
26**En İyi Model:** Grafiklere bakıldığında, **XGBoost**
27modelinin diğer modellere göre **daha iyi** performans
28gösterdiği söylenebilir. Düşüş trendini kısmen yakalaması
29ve dalgalanmalara diğer modellere göre daha duyarlı
30olması, bu modelin bu veri seti için daha uygun olduğunu
31gösteriyor.

32Resim5.5 AEP Tüketim verisi (MWatt cinsinden) Tablosu

Date	Prophet	ARIMA	ETS	Holt	XGBoost	Real Data
01:00	15327.0084687781	18538.9273505170	18416.4357915214	18416.4357915214	19139.8359375	18619.0
02:00	14743.2363148054	18412.8342646080	18099.3146786963	18099.3146786963	19871.060546875	18691.0
03:00	14403.4630397366	18332.5633824061	17785.3647769994	17785.3647769994	20091.109375	19109.0
04:00	14355.5107757579	18281.4629237011	17474.5543743196	17474.5543743196	20151.501953125	20279.0
05:00	14631.9032614426	18248.9323620800	17166.8520756665	17166.8520756665	20341.21484375	20925.0
06:00	15200.8201584696	18228.2234004345	16862.2267999999	16862.2267999999	19936.541015625	21089.0
07:00	15929.685330177	18215.0400715566	16560.6477770900	16560.6477770900	19385.095703125	20999.0
08:00	16621.1058574356	18206.6475619137	16262.0845444092	16262.0845444092	19108.015625	20820.0
09:00	17113.1661769269	18201.3048886969	15966.5069440552	15966.5069440552	19013.279296875	20415.0
10:00	17366.5331845803	18197.9037419223	15673.8851197048	15673.8851197048	18508.93359375	19993.0

33
34

35Resim5.5 AEP Tüketim verisi (MWatt cinsinden) Grafiği



36
37

38Model Performansları:

- **Prophet:** Grafikteki en düşük seyreden modeldir. Gerçek verilerin genel trendini yakalamakta zorlanmış ve özellikle artış döneminde oldukça düşük tahminler üretmiştir.
- **ARIMA:** Prophet'e göre daha iyi bir performans sergilemiş olsa da, gerçek verilerin zirve noktasını ve ardından gelen düşüşü tam olarak yansıtamamıştır. Tahminleri, gerçek değerlerin altında kalmıştır.
- **ETS ve Holt:** Bu iki model neredeyse aynı çizgiyi izlemiş ve genel olarak gerçek verilerin üzerinde tahminler üretmiştir. Trend değişimlerini yakalamakta zorlanmış ve sabit bir çizgiye yakın bir seyir izlemişlerdir.
- **XGBoost:** Diğer modellere kıyasla gerçek verilere en yakın tahminleri üretmiş ve trend değişimlerini (artış ve düşüş) daha iyi yakalamıştır. Özellikle zirve noktasını ve ardından gelen düşüşü diğer modellere göre daha başarılı bir şekilde modellemiştir.

59**En İyi Model:** Grafiklere bakıldığında, **XGBoost**
60modelinin diğer modellere göre **en iyi** performansı
61gösterdiği söylenebilir. Gerçek verilerin genel trendini ve
62önemli değişim noktalarını en iyi şekilde yakalaması, bu
63modelin bu veri seti için daha uygun olduğunu
64göstermektedir.

1	Kaynakça	47
2		
3	Makaleler	48
4	• Bölük, E. (2012). Zaman Serisi Modellerinin	
5	Belirlenmesinde Artıklara Uygulanan Tanı	49
6	Teknikleri Üzerine Bir Çalışma. Dokuz Eylül	
7	Üniversitesi Sosyal Bilimler Enstitüsü.	50
8	ACIK BILIM	
9		51
10	• Yüksel, B. (2023). Zaman Serilerinde Talep Tahmini	
11	(Demand Prediction in Time Series). YBS	52
12	Ansiklopedi, 11(2), 1-16.	
13	YBS ANSIKLOPEDI	53
14		
15	• Utku, A., & Akcayol, M. A. (2021). Derin Öğrenme	54
16	Tabanlı Model ile Bir Olayın Sonraki Olma	55
17	Zamanının Tahmini. Politeknik Dergisi, 24(1), 1-	
18	15.	56
19	HOME	
20		57
21	• Yıldız, S. (2008). Zaman Serisi Analiz	
22	Yöntemlerinin Karşılaştırılması. Yüksek Lisans	58
23	Tezi, İstanbul Üniversitesi Fen Bilimleri Enstitüsü.	
24	ACIK BILIM	59
25		
26	Online Kaynaklar ve Bloglar	60
27	• Brownlee, J. (2020). <i>Time Series Forecasting</i>	61
28	<i>With Python</i> . Machine Learning Mastery.	
29	Retrieved from	62
30	https://machinelearningmastery.com/time-series-	
31	forecasting/	63
32	• TensorFlow. (n.d.). <i>Time series forecasting</i> .	
33	TensorFlow Tutorials. Retrieved from	64
34	https://www.tensorflow.org/tutorials/structured_da	
35	ta/time_series	65
36	Python Kütüphaneleri Belgeleri	66
37	• Prophet Documentation:	
38	https://facebook.github.io/prophet/	67
39	• Statsmodels Documentation:	
40	https://www.statsmodels.org/stable/index.html	68
41	• Scikit-learn Documentation: https://scikit-	
42	learn.org/stable/	69
43	Veri Kaynakları	70
44	• Kaggle Time Series Datasets:	71
45	https://www.kaggle.com/datasets	
46		72

- 1 <https://github.com/ercann06/muhtas> 3