

# Makine Öğrenmesi Modelleri Kullanarak Hava Durumu Analizi ve Sınıflandırılması

Ogün ATALAY  
Bilgisayar Mühendisliği 3.Sınıf  
Kahramanmaraş Sütçü İmam Üniversitesi  
Kahramanmaraş, Türkiye  
ogun.atalay33@gmail.com

June 11, 2024

## Özet:

Hava durumu analizi ve sınıflandırması, tarımın verimliliğinden ulaşımın güvenliğine, afet yönetiminden enerji planlamasına kadar geniş bir yelpazede hayati öneme sahip olan bir alanı temsil eder. Bu bağlamda, hava durumu verileri üzerinde yapılan bu proje, veri analizi ve makine öğrenimi modellemesi kullanarak hava durumu koşullarını sınıflandırmayı amaçlamaktadır. Proje, veri setinin keşfedilmesi, görselleştirilmesi ve ön işlenmesi aşamalarını içermektedir. Ardından, farklı makine öğrenimi algoritmaları, sıcaklık, yağış miktarı ve rüzgar hızı gibi özelliklere dayanarak hava durumu koşullarını sınıflandırmak için eğitilmiş ve test edilmiştir.

**Anahtar Kelimeler:** Hava durumu, veri analizi, makine öğrenmesi, sınıflandırma, sıcaklık, yağış miktarı, rüzgar.

## Abstract:

The analysis and classification of weather conditions represent a vital area that encompasses a wide range of importance, from agricultural productivity to transportation safety, disaster management, and energy planning. In this context, this project conducted on weather data aims to classify weather conditions using data analysis and machine learning modeling. The project includes stages of data exploration, visualization, and preprocessing. Subsequently, different machine learning algorithms have been trained and tested to classify weather conditions based on features such as temperature, precipitation, and wind speed.

**Keywords:** Weather, data analysis, machine learning, classification, temperature, precipitation, wind

## 1 GİRİŞ

Bu proje, gerçek dünya hava durumu verileri üzerinde kapsamlı bir veri analizi ve modelleme yaklaşımını benimseyerek, çeşitli makine öğrenimi tekniklerini kullanarak hava koşullarını tahmin etmeyi amaçlayan bir araştırma çalışmasıdır. Toplanan veri setinde yer alan minimum ve maksimum sıcaklık, yağış miktarı ve rüzgar hızı gibi hava durumu özellikleri, farklı makine öğrenimi algoritmalarıyla değerlendirilerek en

etkili tahminleme modelinin belirlenmesi hedeflenmektedir. Bu çalışma, makine öğrenimi yöntemlerinin pratik kullanımını göstermenin yanı sıra, hava durumu tahminlemesi üzerine genel bir çerçeve sunmayı amaçlamaktadır, böylelikle benzer uygulamalarda da bu metodolojinin kullanılmasına ışık tutmaktadır.

## 2 LİTERATÜR ÖZETİ

Hava durumu analizi ve sınıflandırması, çeşitli algoritma ve yapay zekâ metotları ile birden fazla giriş parametresi kullanılarak gerçekleştirilebilir. Yapay zekâ, karmaşık verilerin daha iyi temsil edilmesini ve yorumlanmasını sağlayan bir öğrenme yöntemidir [1].

Bu projede çeşitli makine öğrenmesi modelleri kullanılmıştır. Makine öğrenmesi; denetimli, denetimsiz ve takviyeli olmak üzere üç ayrı kategoride sınıflandırılmaktadır [2]. Denetimli öğrenme genellikle verinin bir veya daha fazla kategoriye ayrılması için kullanılırken, denetimsiz öğrenme ise veri setindeki değişkenler ve arasındaki ilişkiyi keşfetmeye odaklanmaktadır [3]. Takviyeli öğrenme ise, bir sistemin davranışsal kararlar verme yeteneğini geliştirmek için dünya ile etkileşime girerek kazanılan deneyimi ve değerlendirici geri bildirimi kullanmakla ilgili bir makine öğrenim alanıdır [4]. Projede denetimli makine öğrenmesi modellerinden faydalanılmıştır: Logistic Regression, Random Forest Classifier, Gradient Boosting Classifier, K-Nearest Neighbors Classifier, Support Vector Classifier (SVC), Decision Tree Classifier.

## 3 VERİ SETİ VE YÖNTEM

### 3.1 Veri Seti

Bu projenin veri seti, bir şehirdeki 2012-2015 yılı boyunca günlük hava durumu verilerini içermektedir. Veri setindeki değişkenler şu şekildedir [5]:

- **date:** Ölçülen tarih (yıl-ay-gün formatında).
- **precipitation:** Günlük yağış miktarı (inç cinsinden).

- **temp\_max:** Günlük maksimum sıcaklık (°C cinsinden).
- **temp\_min:** Günlük minimum sıcaklık (°C cinsinden).
- **wind:** Günlük rüzgar hızı (mil/saat cinsinden).
- **weather:** Günlük hava durumu (örneğin, yağmur, kar, güneş, vb.).

Bu veri seti, hava durumu koşullarının farklı özelliklerini içerir ve her bir günün belirli bir hava durumu kategorisine atanmış bir etiketi bulunmaktadır. Bu etiketler, hava durumu tahmin modelinin eğitilmesi ve test edilmesi için kullanılacaktır.

### 3.2 Yöntem

Proje yöntemi olarak, veri seti üzerinde çeşitli makine öğrenimi sınıflandırma modelleri eğitilerek, günlük hava durumu koşullarının tahmin edilmesi amaçlanmaktadır. İşlem adımları şu şekildedir:

1. **Veri Keşfi ve Görselleştirme:** Veri seti görselleştirilerek (histogramlar, çizgi grafikleri, vs.) hava durumu özelliklerinin dağılımı ve değişimi incelenecek ve anlaşılabilecektir.
2. **Veri Ön İşleme:** Veri seti makine öğrenimi modeline uygun hale getirilecektir. Bu adımda eksik verilerin doldurulması, kategorik değişkenlerin kodlanması gibi işlemler gerçekleştirilecektir.
3. **Modelleme:** Farklı sınıflandırma modelleri (Logistic Regression, Random Forest Classifier, Gradient Boosting Classifier, K-Nearest Neighbors Classifier, Support Vector Classifier (SVC), Decision Tree Classifier) kullanılarak veri seti eğitilecek ve test edilecektir.
4. **Performans Değerlendirmesi:** Her modelin performansı değerlendirilecek ve en iyi performans sergileyen model seçilecektir. Bu süreçte, model performansını değerlendirmek için doğruluk, F1 skoru ve diğer sınıflandırma metrikleri kullanılacaktır.

$$F1 \text{ Skoru} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

5. **Sonuçların İncelenmesi ve Yorumlanması:** En iyi model seçildikten sonra, sonuçlar incelenecek ve elde edilen bulgular yorumlanarak bir rapor hazırlanacaktır.

Bu yöntem, günlük hava durumu tahmininde kullanılan makine öğrenimi tekniklerini uygulayarak, veri setindeki bilgilerin kullanılabilirliğini artırmayı ve hava durumu tahminlerinin doğruluğunu artırmayı amaçlamaktadır.

## 4 KULLANILAN MODELLER

### 4.1 Logistic Regression

- **Kullanım Alanı:** Lojistik regresyon, ikili sınıflandırma problemlerinde kullanılan bir modeldir. Bu projede, günlük hava durumu koşullarının belirli bir kategoriye (örneğin, yağmur, kar, güneş) atanması için lojistik regresyon modeli kullanılabilir.
- **Veri Hazırlığı:** Lojistik regresyon modeline veri seti, bağımsız değişkenler (temp\_min, temp\_max, precipitation, wind) ve bağımlı değişken (weather) olarak ayrılır. Bağımlı değişken, kategorik olarak kodlanır.
- **Model Eğitimi:** Veri seti, eğitim ve test setlerine ayrılır. Ardından, lojistik regresyon modeli eğitim veri seti üzerinde eğitilir. Model, giriş özellikleri ile bağımlı değişken arasındaki ilişkiyi öğrenir.
- **Model Tahmini:** Eğitilen model, test veri seti üzerinde kullanılarak tahminler yapar. Model, giriş özelliklerine dayanarak her bir gözlemin hangi hava durumu kategorisine ait olduğunu tahmin eder.
- **Performans Değerlendirmesi:** Modelin performansı, doğruluk (accuracy), hassasiyet (precision), özgüllük (recall) ve F1 skoru gibi metrikler kullanılarak değerlendirilir. Modelin doğruluğu, gerçek ve tahmin edilen hava durumu kategorileri arasındaki uyumu ölçer.
- **Sonuçlar ve Yorumlar:** Modelin performansı incelenir ve yorumlanır. Eğer model yeterince iyi performans gösterirse, günlük hava durumu koşullarının tahmininde kullanılabilir. Ayrıca, modelin güçlü ve zayıf yönleri hakkında bir değerlendirme yapılır ve iyileştirmeler veya değişiklikler önerilebilir.

**Logistic Regression Formülü:** [6]

$$P(Y = 1|X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n)}}$$

**Logistic Regression İçin F1 Skoru Hesaplama Formülü:**

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{Precision (Hassasiyet): } \frac{TP}{TP + FP}$$

$$\text{Recall (Özgüllük): } \frac{TP}{TP + FN}$$

Burada,  $TP$  doğru pozitif sayısını,  $FP$  yanlış pozitif sayısını ve  $FN$  yanlış negatif sayısını temsil eder.

## 4.2 Random Forest Classifier

**Çalışma Prensipleri:** Random Forest, birden fazla karar ağacını (decision tree) bir araya getirir ve her bir ağacın tahminini alarak çoğunluk oyunu bir sonuç verir. Her karar ağacı, rastgele seçilen özellikler alt kümesi üzerinde eğitilir ve veri setinin bir alt kümesi üzerinde tahmin yapar. Bu şekilde, birçok karar ağacının tahminleri birleştirilerek daha doğru ve genelleştirilebilir bir tahmin elde edilir.

**Veri Hazırlığı:** İlk adımda, hava verileri okunur ve gereksiz sütunlar düşürülerek, hava durumu etiketleri sayısal değerlere dönüştürülür. Bu adımlar veri setinin hazırlanmasını sağlar.

### Bağımsız ve Bağımlı Değişkenlerin

**Belirlenmesi:** Makine öğrenimi modelinin eğitilmesi için bağımsız değişkenler ( $X$ ) ve bağımlı değişken ( $y$ ) belirlenir. Bağımsız değişkenler, hava durumunu tahmin etmek için kullanılan özellikleri içerirken, bağımlı değişken hava durumu etiketlerini içerir.

**Eğitim ve Test Setlerinin Oluşturulması:** Veri seti, eğitim ve test setlerine ayrılır. Bu, modelin eğitim verileri üzerinde öğrenmesini ve ardından test verileri üzerinde performansını değerlendirmesini sağlar.

### Modelinin Oluşturulması ve Eğitilmesi:

RandomForestClassifier sınıfından bir model oluşturulur ve eğitim verileriyle eğitilir. Bu adımda, model hava verilerini kullanarak hava durumunu tahmin etmeyi öğrenir.

**Çapraz Doğrulama:** Modelin performansını doğrulamak için çapraz doğrulama yapılır. Bu adım, modelin genelleme yeteneğini değerlendirmeye yardımcı olur ve aşırı uyumu önler.

**Sonuçların Raporlanması:** Çapraz doğrulama sonuçları, modelin performansını belirlemek için kullanılır. Bu sonuçlar genellikle doğruluk (accuracy) değeri olarak raporlanır ve farklı modeller arasındaki karşılaştırmalar yapılabilir.

## 4.3 Gradient Boosting Classifier

**Çalışma Prensipleri:** Gradient Boosting Classifier, bir ensemble learning tekniği olan gradient boosting yöntemini kullanarak bir dizi zayıf öğreniciyi (genellikle karar ağaçları) bir araya getirerek bir sınıflandırma modeli oluşturur. Gradient boosting, bir önceki ağacın hatalarını düzeltmek için bir sonraki ağacı eğitmeye odaklanır.

### İşleyişi:

**Başlangıç Modeli:** İlk olarak, veri seti üzerinde bir başlangıç modeli oluşturulur. Bu başlangıç

modeli genellikle veri setinin sınıf dağılımı üzerinden tahminler yapar.

**Hata Tespiti:** Her bir örnek için gerçek sınıf ile model tarafından yapılan tahmin arasındaki hata hesaplanır. Bu hata, gerçek sınıf ile tahmin edilen sınıf arasındaki fark olarak ölçülür.

**Yeni Ağaç Oluşturma:** Hataları minimize etmek için yeni bir karar ağacı oluşturulur. Bu karar ağacı, hatalar üzerine odaklanarak eğitilir. Önceki ağaçların hatalarını düzeltmek için, yeni ağaç örneklerin özelliklerine dayalı olarak oluşturulur.

**Ağırlıklandırma:** Yeni oluşturulan ağaç, bir önceki ağaçla birleştirilir ve birlikte kullanılır. Ancak, her bir ağacın katkısı ağırlıklandırılarak, daha fazla dikkat edilmesi gereken ağaçlar vurgulanır.

**Adım Adım İterasyon:** Bu işlem, belirlenen bir iterasyon sayısına veya belirli bir hata eşikine ulaşılan kadar tekrarlanır. Her iterasyonda, bir sonraki ağaç önceki ağaçların hatalarını minimize etmek için eğitilir.

**Tahmin:** Son olarak, tüm ağaçların katkısı birleştirilir ve birlikte kullanılarak, bir giriş örneğinin sınıfını belirlemek için tahminler yapılır. Tahminler genellikle bir sınıf etiketi veya olasılık değeri olarak verilir.

## 4.4 K-Nearest Neighbors Classifier

K-Nearest Neighbors (KNN) Sınıflandırıcısı, basit ve etkili bir makine öğrenimi algoritmasıdır. KNN, sınıflandırma problemlerinde ve regresyon problemlerinde kullanılabilir, ancak genellikle sınıflandırma için tercih edilir.

KNN'nin çalışma mantığı şu adımlardan oluşur:

### 1. Eğitim Aşaması:

Eğitim veri seti, özellikler (bağımsız değişkenler) ve hedef değişkenler (bağımlı değişkenler) olarak ayrılır. KNN algoritması, eğitim veri setindeki örnekleri depolar, ancak önceden eğitim yapmaz.

### 2. Tahmin Aşaması:

Yeni bir veri noktası geldiğinde, algoritma bu veri noktasının etrafındaki en yakın  $k$  komşuyu bulur. En yakın  $k$  komşuların etiketlerini (sınıflarını) inceleyerek, verilen veri noktasının sınıfını belirler. Sınıf belirleme işlemi genellikle çoğunluk oyu ilkesine dayanır. Yani, en yakın komşuların sınıfları arasında hangi sınıf daha fazlaysa, yeni veri noktası o sınıfa atanır.

### 3. Model Parametreleri:

KNN algoritmasının en önemli parametresi, komşu sayısıdır ( $k$ ).  $k$ , genellikle tek sayı olarak seçilir, böylece oy kullanırken eşitlik durumunda çoğunluk sağlanır.

#### 4. Uzaklık Ölçümü:

KNN'de en yaygın kullanılan uzaklık metrikleri Öklid mesafesi ve Manhattan mesafesidir. Ancak, kullanım durumuna göre farklı uzaklık ölçüleri de kullanılabilir.

##### KNN Öklid Mesafesi: [7]

K-Nearest Neighbors (KNN) algoritması için birkaç önemli formül vardır. En yaygın kullanılanı, veri noktaları arasındaki mesafeyi hesaplamak için kullanılan Öklid mesafesi formülüdür. Bu formül, iki nokta arasındaki düz bir hattın uzunluğunu verir ve genellikle KNN algoritmasında komşuları belirlemek için kullanılır.

##### Öklid Mesafesi:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(t_{\min x} - t_{\min y})^2 + (t_{\max x} - t_{\max y})^2 + \sqrt{(p_x - p_y)^2 + (w_x - w_y)^2}}$$

$t_{\max}$  : Maksimum Sıcaklık

$t_{\min}$  : Minimum Sıcaklık

$p$  : Tahmin

$w$  : Rüzgar

$\mathbf{x}$  ve  $\mathbf{y}$  : İki farklı veri noktası

#### 4.5 Decision Tree

Bu algoritma, veri setini küçük ve hatta daha küçük parçalara bölerek geliştirilir. Bir karar düğümü bir veya birden fazla dallanma içerebilir. İlk düğüme kök düğüm (root node) denir. Bir karar ağacı hem kategorik hem de sayısal verilerden oluşabilir. [8]

Decision Tree algoritmasının temel bileşenleri ve nasıl çalıştığına dair bazı önemli noktalar:

##### Temel Bileşenler

- **Kök Düğüm (Root Node):** Karar ağacının başlangıç düğümüdür. Tüm veri seti bu düğüme gelir.
- **Dahili Düğümler (Internal Nodes):** Kök düğüm dışındaki düğümlerdir. Bunlar, bir özellik (feature) üzerinde bir karar kuralı içerir ve veri setini alt kümelerine böler.
- **Kenarlar (Edges):** Düğümleri birbirine bağlayan çizgilerdir. Her kenar, bir özellik değeri için bir şarta karşılık gelir.
- **Yaprak Düğümler (Leaf Nodes):** Karar ağacının en alt düzeyinde bulunan düğümlerdir. Her bir yaprak düğümü, bir sınıf etiketi veya bir regresyon değeri içerir.

##### Nasıl Çalışır ?

- **Veri Bölme (Data Splitting):** Karar ağacı, veri setini belirli özellikler (feature) üzerinde karar kuralları uygulayarak alt kümelerine böler. Bölmeler, homojen alt kümeler oluşturmak için yapılır.
- **Karar Kurallarının Oluşturulması (Rule Generation):** Her bir bölme işlemi sonrasında, veri seti bir sonraki bölme adımı için en iyi özellik seçilir. Bu seçim, özelliklerin bilgi kazancına veya Gini katkısına göre yapılır.
- **Ağaç Oluşturma (Tree Building):** Veri seti özellikler üzerinde tekrar tekrar bölündüğünde ve homojen alt kümeler elde edildiğinde, ağaç oluşturulur. Bu süreç, veri setinin tamamı homojen olana kadar devam eder veya belirli bir durdurma kriteri sağlanana kadar devam eder.
- **Tahmin Yapma (Prediction):** Bir veri noktası karar ağacından geçirilerek, yaprak düğümüne ulaşılır ve bu düğümdeki sınıf etiketi veya regresyon değeri tahmin olarak kullanılır.

#### 4.6 Support Vector Classifier (SVC)

Destek Vektör Makineleri (Support Vector Machine) genellikle sınıflandırma problemlerinde kullanılan gözetimli öğrenme yöntemlerinden biridir. Bir düzlem üzerine yerleştirilmiş noktaları ayırmak için bir doğru çizer. Bu doğrunun, iki sınıfının noktaları için de maksimum uzaklıkta olmasını amaçlar. [9]

##### İşlevi:

SVC, iki sınıflı (ikili) ve çok sınıflı (çoklu) sınıflandırma problemleri için kullanılabilir. Doğrusal ayrılabilir sınıflandırma problemleri için düzlemde (veya uzayda) bir karar sınırı belirler. Doğrusal olmayan sınıflandırma problemleri için veriyi yüksek boyutlu uzayda (hiperuzayda) dönüştürerek doğrusal ayrılabilir hale getirir.

SVC'nin temelini oluşturan Support Vector Machine (SVM) algoritmasının matematiksel formülasyonu bulunmaktadır. SVM'nin matematiksel formülasyonu, sınıflandırma problemlerini çözmek için margin maksimizasyonu prensibine dayanır. Veri noktaları için özellikler  $x_i$  ve sınıflandırma etiketleri  $y_i$  olsun. Ayrıştırıcıyı  $w$  ve kesim değerini  $b$  olarak ifade edersek, sınıflandırma kararını şu şekilde yapabiliriz:

$$f(x) = \text{sign}(w \cdot x - b)$$

Burada  $w \cdot x$  iç çarpımı temsil eder. Sınırlayıcıyı bulmak için genellikle bir optimizasyon problemi çözülür:

$$\min_{w,b} \frac{1}{2} \|w\|^2$$

şartıyla,

$$y_i(w \cdot x_i - b) \geq 1$$

Bu, sınıfların doğru şekilde ayrılması için bir marjın en üst düzeyde tutulmasını sağlar. Eğer veri noktaları doğrusal olarak ayrılabilir değilse, "çekirdek"

yöntemi kullanılarak, veri kümesi yüksek boyutlu uzayda dönüştürülerek bu ayırım yapılabilir hale getirilir.

SVC, SVM'nin sınıflandırma amacına odaklanan bir versiyonudur ve SVM'nin bu temel formülasyonunu kullanır.

## 5 FİGÜRLER

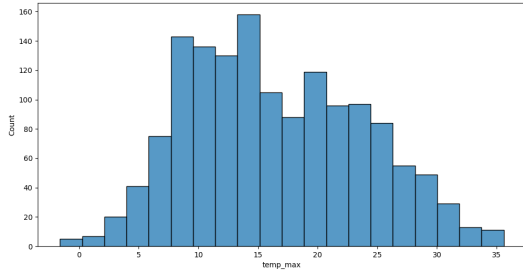


Figure 1: Max. Sıcaklık Değeri Grafiği

Bu grafik, x ekseninde sıcaklık değerlerini ve y ekseninde bu sıcaklık değerlerine sahip olan gündüzlerin sayısını (COUNT parametresi) gösteriyor.

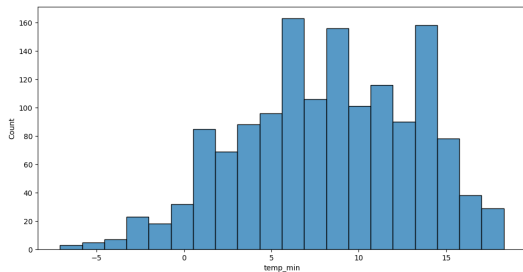


Figure 2: Min. Sıcaklık Değeri Grafiği

Bu grafik, x ekseninde sıcaklık değerlerini ve y ekseninde bu sıcaklık değerlerine sahip olan gecelerin sayısını (COUNT parametresi) gösteriyor.

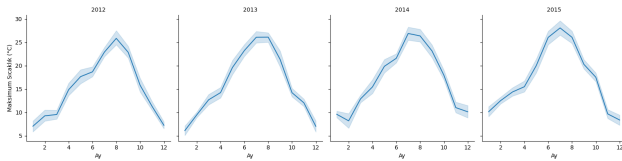


Figure 3: Max. Sıcaklıklar Grafikleri (2012-2015)

Yukarıdaki grafiklerde, 2012-2015 yılları arasındaki ayların ortalama maksimum sıcaklık değerlerini göstermektedir.

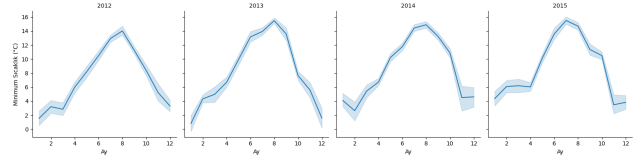


Figure 4: Min. Sıcaklıklar Grafikleri (2012-2015)

Yukarıdaki grafiklerde, 2012-2015 yılları arasındaki ayların ortalama minimum sıcaklık değerlerini göstermektedir.

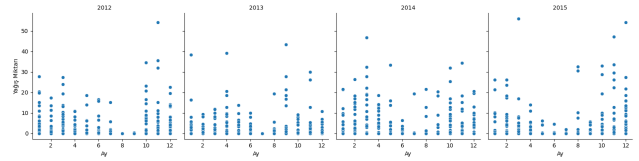


Figure 5: Yağış Miktarı Grafikleri (2012-2015)

Yukarıdaki grafiklerde, 2012-2015 yılları arasındaki ayların yağış miktarları verilmiştir.

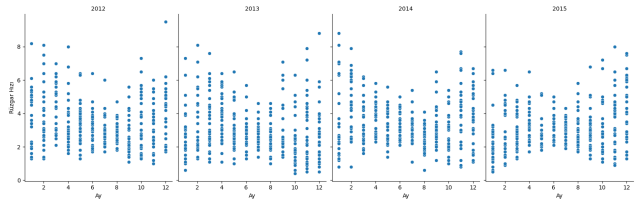


Figure 6: Rüzgar Hızı Grafikleri (2012-2015)

Yukarıdaki grafiklerde, 2012-2015 yılları arasındaki aylarda rüzgarların hızları verilmiştir.

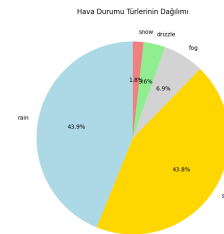


Figure 7: Hava Durumu Türlerinin Dağılımları

Verilen dairesel grafikte hava durumu türlerinin (sun, rain, fog, snow, drizzle) dağılımları verilmiştir. Yıl boyunca güneşli ve yağmurlu günlerin sayısı diğer günlere oranla daha fazladır. En az görülen hava durumu ise karlıdır.

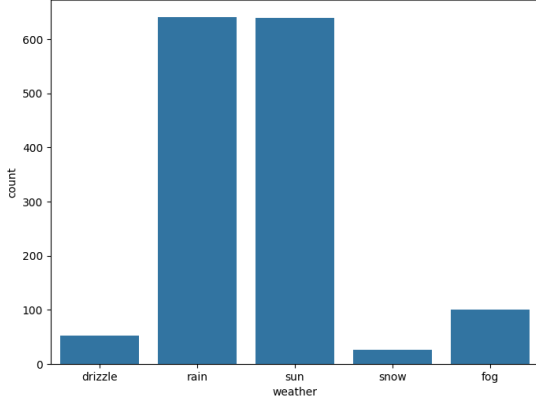


Figure 8: Hava Durumu Türlerinin Tekrarlama Sıklığı

Verilen sütun grafiğinde, hava durumlarının (drizzle, rain, sun, snow, fog) 3 yıl içerisinde kaç defa tekrarlandığını göstermektedir. (Örneğin; yağmurlu gün sayısı 600'dür.)

## 6 BULGULAR

### 6.1 Doğruluk Değerleri Tablosu

Model	CV Accuracy	F1 Score
GradientBoosting	0.802	0.802
RandomForest	0.788	0.789
DecisionTree	0.753	0.752
LogisticRegression	0.735	0.731
KNeighbors	0.716	0.717
SVC	0.700	0.708

Table 1: Model Performansı

Tablo, çeşitli makine öğrenimi modellerinin performansını iki ana metrik kullanarak karşılaştırmaktadır: Cross-Validation Accuracy (Çapraz Doğrulama Doğruluğu) ve F1 Score (F1 Skoru). Her iki metrik de modelin ne kadar iyi performans gösterdiğini ölçer, ancak farklı yönlerden değerlendirirler.

İşte her modelin performansını ve metriklerin ne anlama geldiğini açıklayan bir yorum:

#### Metriğin Anlamı

**Cross-Validation Accuracy (Çapraz Doğrulama Doğruluğu):** Bu metrik, modelin doğruluğunu ölçer. Yani modelin, veriyi ne kadar doğru sınıflandırabildiğini gösterir. Çapraz doğrulama, veri setinin farklı bölümlerinde eğitim ve test işlemleri yaparak, modelin genel performansını daha güvenilir bir şekilde değerlendirmeyi sağlar.

**F1 Score (F1 Skoru):** Bu metrik, modelin hassasiyet (precision) ve duyarlılık (recall) değerlerinin harmonik ortalamasıdır. Özellikle dengesiz veri setlerinde (yani bir sınıfın diğerine göre daha fazla veya az olduğu durumlarda) kullanışlıdır. Hem yanlış pozitifler hem de yanlış negatifler için ceza uygulayarak modelin genel performansını ölçer.

## MODELLERİN PERFORMANSI

### GradientBoostingClassifier:

Cross-Validation Accuracy: 0.802

F1 Score: 0.802

Yorum: Bu model, en yüksek doğruluğu ve F1 skorunu elde ederek diğer modellere göre en iyi performansı gösteriyor. Bu, modelin veriyi hem genel anlamda doğru sınıflandırdığını hem de sınıflar arasında dengeli bir performans sergilediğini gösterir.

### RandomForestClassifier:

Cross-Validation Accuracy: 0.788

F1 Score: 0.789

Yorum: Bu model, Gradient Boosting'e çok yakın bir performans sergiliyor. Doğruluk ve F1 skorunun yüksek olması, modelin güvenilir ve dengeli olduğunu gösterir.

### DecisionTreeClassifier:

Cross-Validation Accuracy: 0.753

F1 Score: 0.752

Yorum: Bu modelin performansı orta düzeyde. Karar ağaçları genellikle daha basit ve yorumlanabilir modellerdir, ancak doğruluk ve F1 skoru, daha karmaşık modellerin gerisinde kalıyor.

### LogisticRegression:

Cross-Validation Accuracy: 0.735

F1 Score: 0.731

Yorum: Bu model, biraz daha düşük performans gösteriyor. Logistic Regression, özellikle lineer olarak ayrılabilir veriler için iyi çalışır, ancak bu veri setinde diğer modellerin gerisinde kalmış.

### KNeighborsClassifier:

Cross-Validation Accuracy: 0.716

F1 Score: 0.717

Yorum: Bu modelin doğruluğu ve F1 skoru daha düşük. KNN, komşuluk tabanlı bir algoritma olduğu için büyük veri setlerinde veya yüksek boyutlu verilerde performansı düşebilir.

### SVC (Support Vector Classifier):

Cross-Validation Accuracy: 0.700

F1 Score: 0.708

Yorum: SVC, en düşük performansı gösteren model. Doğru parametre ayarlamaları yapılmadığında performansı düşebilir, ancak belirli veri türleri ve problem türleri için güçlü bir model olabilir.

## GENEL DEĞERLENDİRME

**En İyi Performans:** GradientBoostingClassifier ve RandomForestClassifier modelleri, yüksek doğruluk ve F1 skoru ile en iyi performansı gösteriyor. Özellikle Gradient Boosting, sınıflar arasında dengeli bir performans sağlıyor.

**Orta Seviye Performans:** DecisionTreeClassifier ve LogisticRegression, daha düşük ancak kabul edilebilir bir performans sergiliyor. Karar ağaçları, açıklanabilirliğin önemli olduğu durumlar için tercih edilebilir.

**Düşük Performans:** KNeighborsClassifier ve SVC, en düşük performansı gösteriyor. KNN'nin büyük ve yüksek boyutlu verilerde performansı düşerken, SVC'nin doğru ayar gerektirdiği görülüyor.

## Kaynakça

- [1] A. Hosny, C. Parmar, J. Quackenbush, L. H. Schwartz, and H. J. Aerts. Artificial intelligence in radiology. *Nature Reviews Cancer*, 18(8):500–510, 2018.
- [2] H. Greenspan, B. Van Ginneken, and R. M. Summers. Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5):1153–1159, 2016.
- [3] K. W. Johnson, J. T. Soto, B. S. Glicksberg, K. Shameer, R. Miotto, M. Ali, and J. T. Dudley. Artificial intelligence in cardiology. *Journal of the American College of Cardiology*, 71(23):2668–2679, 2018.
- [4] M. L. Littman. Reinforcement learning improves behaviour from evaluative feedback. *Nature*, 515(7521):445–451, 2015.
- [5] Ananth R. Weather prediction data set, 2021. Accessed: 2024-06-09.
- [6] OMÜ — Ondokuz Mayıs Üniversitesi. Lojistik regresyon analizi, 2024. Accessed: 2024-06-09.
- [7] Machine Learning Türkiye. Knn: K en yakın komşu, 2018. Accessed: 2024-06-09.
- [8] Erdiñ Uzun. Decision tree - karar ağacı (id3 algoritması) - classification - sınıflama, 2021. Accessed: 2024-06-09.
- [9] Deep Learning Türkiye. Nedir bu destek vektör makineleri? - makine Öğrenmesi serisi 2, 2019. Accessed: 2024-06-09.